

Launcher50+: An Android Launcher for Use by Older Adults

Craig Leamy¹, Bilal Ahmad³, Sarah Beecham²^a, Ita Richardson^{1,2,4}^b and Katie Crowley^{1,2}^c

¹Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland

²Lero – The Science Foundation Ireland Research Centre for Software, Ireland

³Learnovate Centre, Trinity College Dublin, Dublin, Ireland

⁴Ageing Research Centre, Health Research Institute, University of Limerick, Limerick, Ireland

Keywords: Usability, Accessibility, Mobile, Launcher, Older Adult, Software Applications, Apps.

Abstract: While software is becoming increasingly ubiquitous, needed, and available, applications developed for use by older adults do not always take their specific requirements into account. Our thesis is that: *implementing Usability and Accessibility smartphone application requirements which cater for Older Adults' specific needs will improve their ability to engage with software.* Our prior research developed 44 Recommendations for the Development of Smartphone Applications for the Ageing Population (ReDEAP). We assessed 5 existing launchers against relevant ReDEAP recommendations, finding that they had 48-64% compliance. To test the feasibility of ReDEAP, we implemented a subset of recommendations, developing a new launcher app, Launcher50+. The recommendations supported the implementation of a simple user interface, addressing a key concern raised in the wider ReDEAP study. In this paper, we also evaluate other smartphone launcher applications, assessing the level of ReDEAP recommendation compliance. This identified several weaknesses, suggesting that catering for the needs of older adults can be improved.

1 INTRODUCTION


The increase in the proportion of older adults (OAs) (people over 50 years of age) creates new challenges and opportunities. Smartphone applications (apps) have the potential to alleviate problems for OAs, for example, social isolation and healthcare. Research (Ahmad et al., 2021) by three of this paper's authors (BA, SB, IR), developed Usability and Accessibility guidelines for software development – ReDEAP (Recommendations for the Development of Smartphone Applications for the Ageing Population). Although recent studies suggest that personal technology has the potential to cope with some of the challenges related to the ageing population such as social isolation (Barbosa Neves et al., 2019) and physical and psychological health (Matthew-Maich et al., 2016), we identified that most apps for OAs do not seriously consider these aspects. The rate of adoption of personal technologies for OAs appears low despite the potential benefits they intend to


provide (Lee and Coughlan, 2015). Examples of such technologies include public displays (Lindley, 2012), websites (Sudore et al., 2014) and apps (Bull et al., 2017). Since different technologies require a different set of design requirements, this research focused only on one form of personal technology, apps, because they appear to be frequently used by OAs (Berenguer et al., 2016). Also, they can help OAs by providing them access to digital services given the portability, low-cost and controlled nature of many mobile platforms (Bull et al., 2018).


ReDEAP provides a structured and evaluated set of recommendations in design pattern format (Ahmad et al., 2021). In this paper, we assess existing launcher apps against ReDEAP, and then validate some ReDEAP recommendations by developing a mobile launcher, Launcher50+.

1.1 ReDEAP Recommendations

ReDEAP was developed through uncovering the needs of OAs and transforming these into a set of

^a <https://orcid.org/0000-0003-1584-5447>

^b <https://orcid.org/0000-0002-5493-2837>

^c <https://orcid.org/0000-0003-3596-4363>

recommendations to support usability and accessibility for OAs (OA-Recommendation). They are in design pattern format (Alexander et al., 1977) which makes them available and actionable for developers (Ahmad et al., 2021). The key research question for that study was: “What do developers of smartphone applications for older adults need to do to make their applications usable and accessible for their target population?” Following a literature review on app development for OAs, surveys, interviews and observation were conducted with, in total, 235 OAs who had mixed levels of technical experience. A proof-of-concept application was developed, and further data collected based on observation of OAs’ interaction with their smartphones and online forums. Thematic analysis, descriptive statistics and inferential statistics were applied to yield an augmented set of recommendations. The derived themes and recommendations were evaluated by conducting inter-rater reliability tests and were subsequently transformed into design pattern format. These were augmented by studying the expectations of technically proficient (tech-savvy) OAs. Thus, ReDEAP consists of 44 empirically derived and evaluated set of recommendations. ReDEAP addresses a problem - namely, that, when designing apps for this section of the population, industry needs to take notice of the specific usability and accessibility needs of OAs.

1.2 Project Summary

The research presented in this paper arises from a capstone final year project undertaken by author CL under the supervision of IR. Its’ goal was to implement ReDEAP recommendations, demonstrating their practicality for software engineers. Thus, the developed software should comply with the needs of OAs. To test the feasibility of ReDEAP, a mobile launcher was developed – Launcher50+. Also called a home-screen replacement, this is an app that temporarily modifies the software design and features of a phone's operating system, controlling the home screen, app drawer, and lock screen of a smartphone. Through customising the home screen of a device, it provides users with a graphical user interface through which they can organize and manage applications and other items on their phones. Launchers are available in app stores. Additionally, we evaluated other launcher apps to assess the level of ReDEAP recommendation compliance.

Launcher50+ should:

- Be customizable, meaning that the user could, for example, change the app theme, font size, icon size;
- Be easily usable by non-tech savvy OAs;
- Make smartphones easier to use.

2 RESEARCH METHODS

2.1 OA-Recommendations

Initially, each OA-recommendation in (Ahmad et al., 2021) was evaluated as to how they could be incorporated into Launcher50+. They were divided into 3 categories (see Table 1) – those that would be implemented, those that are directly applicable to a launcher (25 recommendations, 19 which were implemented in Launcher 50+ - see Appendix 1), those which could apply to a launcher depending on the features offered by the launcher (8 recommendations), and those that were not at all applicable to a mobile app launcher (11 recommendations).

2.2 Review of Existing Applications

We identified that, in App stores, there are many simplified Android Launchers available, including some launchers that are marketed for older users. The “Top 5” simplified launchers, as determined by the Google Play Store, were reviewed. They are: Big Launcher (Fig 1), Simple Launcher, Elder Launcher, BaldPhone and Simple Mode.

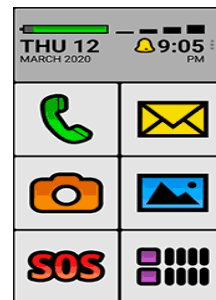


Figure 1: Big Launcher (Big Launcher, 2022).

We assessed the “Top 5” launchers by comparing them to the 33 ReDEAP OA-recommendations which could apply to a launcher, as shown in Table 1.

To gain further insight into the selected launchers we examined reviews from the Google Play Store. A Python script was written to scrape every review

which had been written about each of these applications. This made use of an existing library called “*google_play_scraper*”. After scraping all the reviews, the script was formatted and stored in a spreadsheet.

Table 1: OA-Recommendation categorization.

Category	Description
Recommendations to be implemented (25)	These are recommendations that are directly applicable to a launcher. They specify guidelines for things you would expect to see in a launcher. For example, “Use clear text instead of pictorial stimuli to relay information” (Ahmad et al., 2021) would be applied to app launch icons.
“Wishlist” features (8)	These are recommendations that could apply to a launcher depending on the features offered by the launcher. For example, “Allow the older adults to choose a preferred theme in the application” (Ahmad et al., 2021). Choosing a theme is not a necessary feature but could be added given enough time.
Recommendations that are not applicable (11)	These are recommendations that are not at all applicable to a launcher. They specify guidelines that would be impossible to implement. For example, “Incorporate a display without glare or reflection from the touch screen” (Ahmad et al., 2021). It is impossible for a launcher to control the type of screen a user has on their smartphone.

We excluded unhelpful reviews, as some reviews did not have any real content or yield any usable information, e.g. “Nice App”. Content Analysis was then used to analyze these qualitative comments. This involved reading through the reviews and marking the review with codes. For example, if a review complained about the app having a lot of errors, that review would get code E. The sum of the number of E’s indicated how frequently errors were complained about. Typically, the negative reviews were a better

indicator of what features required improvement. Results from this stage are shown in Table 2.

2.3 Questionnaire

A questionnaire, answered by 10 OAs, 9 of whom were aged 50-60, and one aged 60-70, was developed to gain further insight into user experiences and difficulties faced with the default Android mobile phone launcher. We used snowballing to identify participants in this questionnaire. They came from a variety of different backgrounds, with different lifestyles, and living in urban and rural areas of Ireland. Ethical approval was secured to administer the questionnaire.

Table 2: Content Analysis Results.

	Applications				
	BL	SL	ELr	BP	SM
Errors	10	10	3	5	1
Feature Requests	5	13	4	8	11
Usability	4	11	3	2	2
Dignity	3	0	1	2	0
Total Reviews	150	150	21	61	68
% of Reviews	14%	22%	52%	27%	20%

BL: Big Launcher, SL: Simple Launcher, EL: Elder Launcher, BP: Bald Phone, SM: Simple- Mode

Table 2 outlines the results of the content analysis results under the following headings:

- **Errors:** This refers to any review which mentioned an error e.g. the app crashing;
- **Feature Requests:** This refers to any review where the user is asking for something new to be added to the app;
- **Usability:** This refers to any review where users are complaining about the app being difficult to use or unintuitive;
- **Dignity:** This refers to any review where users mention being embarrassed to use the launcher or being offended by terms such as “elder”;
- **Total Reviews:** This is the number of evaluated reviews. Some launchers did not have enough reviews and hence had lower amounts of evaluated reviews;
- **% of Reviews:** This is the percentage of total reviews that mentioned one of the 4 labels.

Design Pattern (sample of 9 shown)	BIG Launcher	Simple Launcher	Elder Launcher	BaldPhone	Simple Mode
Allow cancellation of a selection made by the user in the application					
Incorporate fast swiping in the application					
Concentrate information mainly in centre of the application interface					
Use simple, clear and consistent terminology and navigation in the application					
Use layman's language in the application for error notifications					
Use precise and easy to understand terms and conditions in the application					
Incorporate ALT tags for the images					
The application should not break during execution of functionality					
Show the right keyboard at the right time. e.g. when there is a need to enter a phone number, show only numeric keys					
Total found in each product (out of 25)	16	17	17	21	17
%(out of 25) found in each product	48%	51%	51%	64%	51%

Figure 2: Excerpt from Comparison spreadsheet.

3 ANALYSIS

3.1 Existing Applications

None of the Top 5 launchers adhered totally to ReDEAP recommendations with some having complex user interface components or ambiguous information. For example, Fig 1 is the default home screen from the top launcher (Big Launcher). There are no text labels for the icons, which are themselves not standard, and some menu options are ambiguous. For example, the functionality of the bottom right icon was unclear until it was pressed.

Analysing these launchers against Ahmed's (2021) recommendations (Fig. 2) shows from 48% to 64% adherence. This is a rather large difference, as each missed OA-recommendation contributes to a decrease in either usability or accessibility. To illustrate this, apart from the sample shown in Fig. 2, these three OA-recommendations were not evident in any of the 5 launchers reviewed:

- Use clear text instead of pictorial stimuli to relay information;
- Give specific and clear instructions and make help and documentation available in the application;
- Use small high-speed adjustments for user vibration feedback to provide easily detected sensation.

Omitting these OA-recommendations would have a large impact on the usability of an application.

This analysis provided a good understanding of what users expect from a launcher like this. From this, and specific feature requests, a shortlist of potential features was developed to include in the new

launcher, Launcher50+:

1. Ability to remove default icons/apps on the home screen
2. Ability to change icons
3. Ability to zoom in on specific sections
4. Have an optional SOS emergency button
5. The SOS emergency button should be regional
6. Allow quick contacts on the home screen
7. Have the option to display weather in Celsius or FahrenheitChoose between a 12hr or 24hr time format
8. Have a pill reminder to remind users to take any medications regularly
9. Change the colours used in the app
10. Change the icon sizes
11. Allow editing to be locked (To prevent accidentally moving/deleting apps).

Launcher50+ implements 19 of Ahmed's (2020) 44 recommendations. This was based on the amount of time available to complete the student project.

3.2 Questionnaire Results

The results of the questionnaire showed that the age of participants was in the younger band of the "adults over 50" age range. This uneven sample meant that we could not draw any general conclusions for all adults over 50 years old. Furthermore, as the sample was so small, we cannot say that the results were conclusive. However, they do give an indication of user issues.

We found that some of the usability issues outlined by Ahmed (2020) do not seem applicable to this younger age range. For example, no respondent marked reading text or scrolling as difficult. Despite

this, many recommendations were applicable, even to this age range. These included:

- 40% of respondents noted having issues with associating an app icon with the app
- 80% of respondents wanted the ability to customize their phone's home screen
- 50% of respondents liked when their phone gave haptic feedback.

In addition, the final free-text question allowed users to type their feedback and some responses aligned with ReDEAP recommendations and the findings from analysis of existing launchers. One user noted they *“can struggle to find apps that are not on the home screen sometimes”*. Another said, *“My most commonly used apps are all scattered across multiple pages of apps. I wish I could put them all together instead of always scrolling looking for them”*.

For future studies, it would be beneficial to survey more people in an older age range. While hints of the problems outlined in other launcher reviews and ReDEAP recommendations are present in this age range, there is not a widespread difficulty with using smartphones in this age range.

3.3 Dignity

Analysis of existing apps highlighted an issue that had not previously been identified when developing ReDEAP - dignity. Results showed that many people were embarrassed to use these simplified launchers. They did not want people to see their phones as the simplicity felt degrading. This led to people leaving negative reviews and opting to revert to the default launcher. However, these reviews only occurred in the very over-simplified launchers. The two launchers which had a more modern style had no reviews with these types of complaints.

A further literature search on the topic was conducted and it was found that similar problems have been reported with assistive technologies in general. People avoid asking for help as they do not want to reveal their lack of knowledge (Kuerbis, 2017). For some, utilizing such devices is an embarrassing admission of dependence (Kang, 2010). The proposed solution to this problem is “invisibility”. That is, the assistive technologies must be unobtrusive or invisible to other people.

3.4 Development Platform

Development of Launcher50+ was restricted to the Android operating system as Apple(iOS) does not allow for custom launchers, so the user would need to

jailbreak the phone to install the mobile launcher. Jailbreaking is essentially hacking the phone, which ruled out iOS.

Two options were available for the Android platform: to create a custom firmware or to develop a launcher application. The custom firmware would give the developer full control of the operating system and allow for much more flexibility. It would also make the launcher more reliable and less prone to error. However, installing this would mean installing a new operating system on the device. As the capstone project was aiming to make the phone easier to use, this complex setup step would be a huge barrier to the target audience of the app.

The second option was to create a launcher app. This is a standard Android application that listens for the event broadcasted when the user presses the home button. In this case, the app can interrupt and bring the user to the launcher screen instead of to their regular home screen. This option comes with a major advantage - it is installed in the same way as any other app. The user can simply download it from the Google Play Store and install it as normal. Thus, it was decided to develop Launcher50+.

An Android Application is made up of four components:

- **Activities:** This is the visual part of the app. Which is usually made up of different views. In Launcher50+, this is where the user's home screen is displayed.
- **Services:** These are tasks which run in the background, regardless of if the app is on-screen or not. Typically, the user is not aware of these background services.
- **Broadcast Receivers:** This receives and reacts to events that are broadcasted by other apps. In Launcher50+ the broadcast receiver is listening for the home button being pressed.
- **Content Providers:** This is essentially an interface between the app and any data stored. This information can be shared with other apps if allowed. For example, the “Contacts” app allows any app with proper permissions to read the user's contacts.

4 PRODUCT

4.1 Core Functionalities

The first core functionality that was implemented was listening for the event that is broadcasted when the user presses the home button, at which point



Figure 3: First Working Version of Launcher50+.

Launcher50+ starts. Another core functionality was ensuring that the software could find and display all installed applications. With these two functionalities, the launcher was already usable as a replacement launcher. However, as Fig. 3 shows, it was still far from complete. The next step was designing the layout of Launcher50+. A wireframe is a plan of what components will be displayed on each page, providing a visual understanding of a page early in a project (Experience UX, 2022).

4.2 Home Screen

The *Home Screen* (Fig. 4) is the starting point of the launcher. It displays the time, navigation buttons, and shortcuts to the user's chosen favourite apps. While this screen may look simple at first, there are several subtle details to note here.

A deliberate section of white space was left at the bottom of the screen. This was done to avoid having the “All Apps” button at the very bottom of the display, because newer smartphones, with edge-to-edge displays, can cause strain reaching to the bottom of the display. This was based on feedback from the content analysis and findings from Ahmed (2020) where he describes adults’ fingers as “dry and husky”.

Another design decision was in the positioning of the “Change Preferences” button. A common complaint observed in the analysis of other launcher reviews was that users would accidentally remove apps from their home screen. To avoid this, we positioned the button in a location where it would be difficult to press unintentionally. In addition, to follow the OA-recommendation, the button would need to be plain text instead of the regular ‘cog’ icon usually used to represent “change preferences”. The

decision to place it as the second button from the top meant that a user would have to deliberately reach for it, avoiding accidental selection. It is also far enough from the top of the display to not be pressed while swiping down the notification tray. As an additional precaution, we added a “Are you sure you want to proceed?” pop up (Fig. 4). We also opted to make the buttons on the home screen slightly transparent. If the user selects their own choice of a background image, it would be difficult to see behind large, coloured buttons. With this transparency, background images are visible without impacting the large button usability.

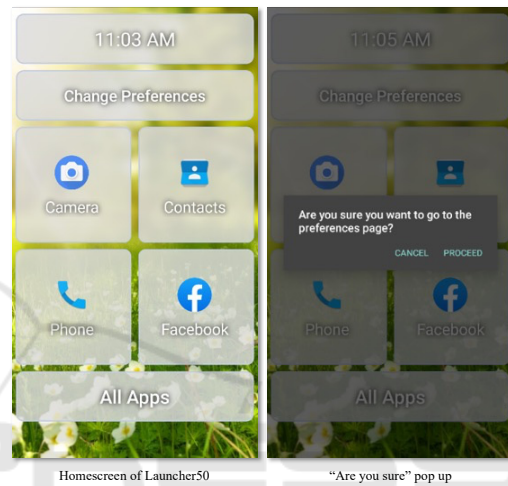


Figure 4: Launcher50+ Homescreen (left) and “Are you sure?” pop-up (right).

4.3 Font Screen

The *Font Screen* (Fig. 5) allows the user to select the launcher font. Viewing a preview of each font option, the user clicks to select it, and presses “Apply” to save changes. ReDEAP recommends four fonts: Serif or Sans Serif, Helvetica, Arial, or Times New Roman (Ahmad et al., 2021). Launcher50+ uses Arial as the default font with an option for Helvetica. It would be possible to add additional fonts in future versions.

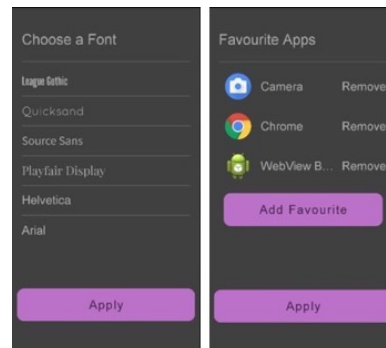


Figure 5: Font Screen and Favourites Screen.

4.4 Favourites Screen

The *Favourites Screen* (Fig 5) allows users to choose their favourite apps. The options to remove existing favourites and add new favourites are both clear and easy to read. If an app name is too long to fit here, it is automatically truncated.

4.5 All Apps Screen

The *All Apps Screen* (Fig. 6) displays all installed apps in alphabetical order. Alphabetical order was chosen for consistency instead of having the list dynamically change order. Each app is pulled from the phone's package manager. It is then processed to find the app name, icon, and launch package. The icons are then resized to be a uniform size.

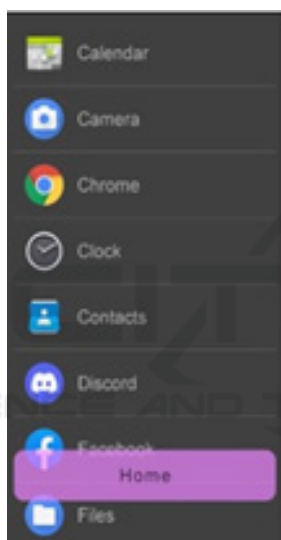


Figure 6: All Apps Screen.

4.6 Lock Screen

This is a feature that was not developed in the launcher. Implementing this would have involved overwriting the default lock screen to apply ReDEAP OA-recommendations. However, this results in security issues as no app can be as secure as the operating system level locking mechanism.

4.7 Testing

Manual testing, primarily carried out during development, consisted of listing every feature of the launcher and manually clicking through them to ensure that they worked as expected. This was done as features were implemented. While this is not a

foolproof method of testing, it was quick and easy and worked well for development.

Once a feature was working, an automated test was written using Android Instrumentation tests and the Espresso testing library. This allowed for automatic click-through testing within the launcher. These automated tests could be re-run every time a new feature was added to ensure that no existing features were broken in the process of adding new features. The final version of the launcher has 8 automated test suites giving coverage of every aspect.

5 CONCLUSION

Older adults have distinguishing characteristics which need to be considered when developing apps, as identified in our previous research. In this paper, 19 OA-recommendations, taken from ReDEAP, are implemented and tested when developing Launcher50+. When looking at feedback from users of other popular launchers, a new recommendation which was identified is to ensure that people's dignity is maintained. Designers of apps for OAs need to balance the need for simplicity with that of creating an interface that could be considered demeaning.

We demonstrated how ReDEAP's OA-recommendations can be used to support the development of OA software. Our future work includes evaluating Launcher50+. We also intend developing further apps to demonstrate the implementation of the remaining 25 recommendations. Through the development of Launcher50+, we have demonstrated that ReDEAP is a usable set of recommendations which can be used successfully by software engineers

ACKNOWLEDGEMENTS

This research was supported in part by a grant from Science Foundation Ireland to Lero – the SFI Research Centre for Software under Grant numbers 13/RC/2094 & 13/RC/2094_P2. The capstone project is a core requirement for the BSc in Computer Systems in the Department of CSIS at UL.

REFERENCES

- Ahmad, B., 2020. Recommendations for developing smartphone applications for an ageing population, PhD Thesis: University of Limerick

Ahmad, B., S. Beecham and I. Richardson (2021), Towards Design Patterns for Developing Smartphone Applications for an Ageing Population, 2021_TR02_Design_Patterns_ReDEAP, www.lero.ie

Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdalh-King, I. and Angel, S., 1977, A Pattern Language: Towns, Buildings, Construction, Oxford University Press, New York, U.S.A. ISBN-13: 978-0195019193

Barbosa Neves, B., Franz, R., Judges, R., Beermann, C. and Baecker, R. (2019), 'Can digital technology enhance social connectedness among older adults? a feasibility study', *Journal of Applied Gerontology* 38(1), 49–72. 3

Berenguer, A., Goncalves, J., Hosio, S., Ferreira, D., Anagnostopoulos, T. and Kostakos, V. (2016), 'Are smartphones ubiquitous?: An in-depth survey of smartphone adoption by seniors', *IEEE Consumer Electronics Magazine* 6(1), 104–110. 3, 33, 175, 178

Big Launcher, 2022. Big Launcher. [Online] Available at: <http://biglauncher.com/> [Accessed 10 01 2022].

Bull, C. N., Simm, W. A., Knowles, B., Bates, O. E. G., Davies, N. A. J., Banerjee, A., Introna, L. D. and Hayes, N. (2017), 'Mobile age: Open data mobile apps to support independent living'. 3

Bull, C., Harding, M., Mikusz, M., Knowles, B., Davies, N. and Hayes, N. (2018), 'Designing mobile applications for older adults', *HotMobile* . 3, 4

Experience UX, 2022. Experience UX. [Online] Available: <https://www.experienceux.co.uk/faqs/what-is-wireframing/> [Accessed 25 April 2022].

Kang, H. G., 2010. In Situ Monitoring of Health in Older Adults: Technologies and Issues, s.l.: s.n.

Kuerbis, A., 2017. Older adults and mobile technology: Factors that enhance and inhibit utilization in the context of behavioral health., s.l.: s.n.

Lee, C. and Coughlin, J. F. (2015), 'Perspective: Older adults' adoption of technology: an integrated approach to identifying determinants and barriers', *Journal of Product Innovation Management* 32(5), 747–759. 3, 14, 27, 178

Lindley, S. E. (2012), 'Shades of lightweight: supporting cross-generational communication through home messaging', *Universal Access in the Information Society* 11(1), 31– 43. 3

Matthew-Maich, N., Harris, L., Ploeg, J., Markle-Reid, M., Valaitis, R., Ibrahim, S., Gafni, A. and Isaacs, S. (2016), 'Designing, implementing, and evaluating mobile health technologies for managing chronic conditions in older adults: a scoping review', *JMIR mHealth and uHealth* 4(2), e29. 3

Sudore, R. L., Knight, S. J., McMahan, R. D., Feuz, M., Farrell, D., Miao, Y. and Barnes, D. E. (2014), 'A novel website to prepare diverse older adults for decision making and advance care planning: a pilot study', *Journal of pain and symptom management* 47(4), 674–686. 3, 15.

APPENDIX

OA-Recommendations Implemented in Launcher50+

OA-Recommendation	How it is implemented
Use clear text instead of pictorial stimuli to relay information.	Most buttons in the application are text-only. When icons are used for app logos, they are always accompanied by large text.
Make accessible without need for passwords e.g., fingerprints.	The app is not password protected.
Allow sufficient white space to ensure a balanced user interface design.	The amount of information displayed on the screen at any one time is always low.
Use font type: serif or sans serif, Helvetica, Arial or Times New Roman in the application.	Arial font is used by default, with the option of Helvetica.
Allow the older adults to choose a preferred theme.	The application's theme can be customized by the user.
Avoid animations and marquees (e.g. text moving top to bottom)	There are no animations or marquees.
Disable inactive user interface objects.	The only object which can be inactive is the home screen favourite shortcuts. Empty buttons are not displayed..
Use consistent and explicit step-by-step navigation and user interface elements in the application.	The user navigates through the user interface by clicking on buttons with the name of the screen they are trying to visit.
Avoid scrolling in the application or only allow vertical scrolling for such scenarios.	Vertical scrolling is only used where necessary - on the full apps list. It is not used between the home screen and all apps screen..
Provide and make sure that the back button behaves predictably	Using the back button always returns to the previous screen. The default back button sometimes had to be overridden.
Prevent an error rather than have a recovery mechanism for it.	Launcher50+ passed 100% of test cases, and author 1 has been using it personally without any errors.
Do not have links to dodgy websites in the application, that may entice older adults to pay for certain things.	Launcher50+ does not contain any links.

OA-Recommendation	How it is implemented
Avoid the advertisements in the application.	Launcher50+ does not contain any advertisements.
Use small high-speed adjustments for user vibration feedback to provide an easily detectable sensation above the threshold.	Every time a button is pressed a small high-speed vibration is played.
Allow cancellation of a selection made by the user in the application.	The favourites, font, and theme pages can all be cancelled instead of applying changes made.
Incorporate fast swiping in the application.	Fast swiping is present on the full app list.
Concentrate information mainly in centre of application interface.	The user’s favourite apps are in the centre of the screen.
Use simple, clear and consistent terminology and navigation in the application.	Language is consistent throughout the app, eg: “Apply” is always used to apply settings. Simple terminology is used, e.g: “Preferences” instead of “Settings”.
The application shouldn’t break during the execution of functionality.	Launcher50+ passed 100% of test cases, and author 1 has been using it personally without any errors.

