A Case Study of Research through the App Store: Gaming the Android OS for Improving the Design of Smartphone Launchers

Matthias Böhmer, matthias.boehmer@dfki.de, DFKI GmbH, Saarbrücken, Germany*

Antonio Krüger, antonio.krueger@dfki.de, DFKI GmbH, Saarbrücken, Germany

Abstract

With a growing number of mobile applications available on application stores and the improved capabilities of smartphones, people download more applications to their devices. Researcher began to leverage this momentum for distributing applications for conducting studies on end-users' devices. This paper grounds the approach of research through the application store in the theory of quasi-experimental design. Further, with people having more applications installed, finding applications quickly when they need them can become a time consuming issue that impacts user experience. In this paper, we present our approach to improve future design of smartphone launcher menus. We present our approach of combining research through the app store with the idea of studying people's smartphones as the apparatus themselves. Therefore we designed a game that takes advantage of the user's smartphone itself as a game court. By measuring the time of a simple visual search task for an icon, we aim to deduce how well a user knows where he can find his applications, thus how well he can build a mental model of his smartphone launcher menu. We introduce our approach, present the game rappidly that serves as a vehicle of our research question, and discuss open challenges and future work.

Author Keywords: Mobile applications; game; operating system; launcher menus; smartphones; research through the app store.

INTRODUCTION

The number of applications (apps) available on mobile app stores is steadily increasing, and so is the number of application downloads from app stores. This also results in a growing number of apps people have installed on their devices, since many people keep applications installed without removing them after installation, and rather for instance move them to special places within their menus (Böhmer and Krüger, 2013). People put quite some effort into organizing their launcher menus. However, with an increasing number of icons in a launcher menu, finding an application for launching it can also become a time critical task. Although such time spans might be very short, searching too long for an app might negatively impact user experience. Minimizing this search time is the goal of ongoing research by presenting predictive launcher menus (e.g. Böhmer and Krüger, 2013; Parate et al., 2013; Shin et al., 2012; Zhang et al., 2012).

This paper has a twofold content: First, we present a new way of studying mobile phenomena by combining the idea of research through the application store and making the user's smartphone an integral part of the study apparatus. This enables to study interactions that are inherent to the smartphone and not only to the published application to the app store. We present a categorization of related works and explain how our approach goes beyond what has been done so far. Second, we present a case study of this approach stating a particular research question, which is: *How well do people know their current launcher menus?* Our goal is to quantify the time it takes to find a certain app on a smartphone, and to build a model for predicting this effort. This will help to improve menus of smartphone launchers, e.g., if a user would take exceedingly long to find and launch his next app (for prediction of next used applications see Parate et al., 2013; Shin et al., 2012), an adaptive menu could provide a shortcut to that app where he can easily find it when visually searching for that particular app (Zhang et al., 2012).

As such, this paper makes three contributions: (i) We present our approach of making the operating system (OS) an inherent part of a game to study questions inherent to the design of the smartphone; (ii) we present a game as a study in the large to address our specific research question on visual search times in launcher menus; (iii) we present preliminary findings on how fast people can find icons; and discuss how it can be used for informing future design of smartphones.

LEVERING APPLICATION STORES FOR RESEARCH

Collecting Data in the Wild

According to Rodgers, 2012, the idea of turning to the wild is to study *"phenomena in the context rather than in isolation"*, and is all about observing how people change, react to, or integrate novel technologies into their everyday lives. The approach of studying new technologies in the wild was used in different fields of HCI and ubiquitous computing to study the use of new or existing systems in situ. Rodgers remarks that isolating specific effects observed in an in-the-wild study is difficult since the participant rather than the researcher is in control of the study, and that effects may be caused by dependencies between various factors. According to this understanding, implementing a research study into a mobile application and deploying it an application store can be seen as a special case of conducting research studies in the wild.

Quasi-Experimental Design

Oulasvirta, 2012, urges rethinking experimental design when studying mobile and context-aware systems. His rationale is that assumptions about randomization and control, which can be made for experiments conducted in a controlled laboratory environment, are not necessarily valid in the wild. He proposes grounding the universal practicality of conducting studies in the scientific validity of quasi-experimentation design. Laying out the theory of experimental and quasi-experimental designs, Shadish et al., 2001, characterize an experiment to be a study where the investigator purposefully applies two or more treatments to parts of a sample to observe the treatments' effects. Common to different forms of experiments is the control of which treatment shall be applied to which units of the sample, though the form of control can differ. Differentiating the researchers' degree of control leads to four different kinds of experiments (Shadish et al., 2001):

- In *randomized experiments* the control over the experiment is typically applied by randomly assigning treatments to units of the sample. Observed differences in the effects of treatments between groups are likely to be caused by the treatments themselves.
- In *quasi-experiments* the experimenter gives up control of assigning treatments to sample units; most importantly, they are not randomly assigned. Instead, participants in these studies self-select their treatments.

- In *natural experiments* the treatment occurs naturally and comparable conditions are introduced afterwards. As such, neither the experimenter nor the study participants are in control of applying the treatment.
- Finally, in *passive observational studies* there is no treatment of subjects at all; strictly speaking, this is by definition a non-experimental design, rather than an experiment. The aim here is to observe relationships between variables.

Moving on this continuum from controlled randomized experiments to natural experiments raises questions regarding the validity of the study, reasons for this being loss of control over study participants and loss of randomization of units to study conditions. Oulasvirta additionally distinguishes randomized experiments into the two types of laboratory experiments and analogue experiments when applying this theory to studies in the field of HCI (Oulasvirta, 2012).

Research in the Large

Many researchers began to leverage the momentum of the growing mass of mobile application users who would install applications through the application store. The main motivation was to be able to gain large amounts of data for statistical analysis, run studies with a heterogeneous sample of participants, and observe behavior in naturally occurring user contexts (Henze and Pielot, 2013). Consequently, this new research approach began to establish itself as a new instrument in mobile HCI research that gained momentum as research in the large (coined in particular by the workshop series on Research in the Large; Poppinga et al., 2012).

Research through the Application Store

With an increased focus on application stores as the means for running studies we also refer to this instrument as *research through the application store*. In particular, we make this distinction since studies related to HCI can — obviously — also go large in terms of number of participants, geospatial spread and length of time period of observations without being distributed over an application store; e.g. when relying on web technologies (cf. Hilbert and Redmiles, 1998; Church et al., 2008) or being an inherent part of the operating system itself. Further, while Henze and Pielot, 2013, in particular praise application stores as a convenient means for the distribution of studies, we also leverage this possibility to get access to low-level APIs and to study inherent properties of smartphones, which is technically only possible when using native applications.

As described previously, the method of research through the application store can be grounded in the method of quasi-experimental design, and can be seen as a special case of collecting data in the wild. When conducting a study on an application store the experimenter naturally looses control over the selection of participants, since they are self-selected.

RELATED STUDIES DONE THROUGH THE APPLICATION STORE

Although the approach of research in the large is relatively young, it has been applied to different research questions in HCI, ubiquitous computing and related research. For a better overview on work related to this method, and additional classification of the method, we adopt categories from Henze's MobileHCI tutorial (Henze, 2011). We distinguish among five categories of works: (1) using application stores as proofs of concepts and for dissemination of work, (2) leveraging application stores as a research tool to study distinct research questions and learn about new aspects when up-scaling studies, (3) for study of dedicated research questions at scale, and

(4) for studying the ecosystems of smartphones and application usage itself. We will further add to this classification (5) a group of papers presenting lessons learned from using the approach previously presented. While we will present some related works as belonging in individual categories, some research actually falls into more than one category.

Proving Concepts and Disseminating Results

The motivation of deploying an application to app stores for some works is to prove the concept behind the research, or collect additional feedback on research contributions by making the work available to users. These types of works mainly rely on the review and comments function that application stores provide to collect feedback on their concepts (Henze and Pielot, 2013). Another motivation also might be to disseminate existing work, and make research results available for end-users, maybe aiming to acquire additional insights by deploying the system.

Wang, 2009, presents the design of *Ocarina*, which is a musical instrument for the *iPhone* that uses its touch display and additional sensors for creation of tones. By making their application available on the Apple AppStore, they were able to reach more than a million users, which allowed them to investigate users' social experiences of their application.

Buddharaju, 2010, test the concept of an application for measuring physical activity while walking, leveraging proxy measures for metabolic measurements. By deploying the application to the Apple AppStore and collecting data on the body mass index of users and daily patterns of physical activity, they were able to argue for the reasonability of their concept.

Zhai et al., 2009, present *ShapeWriter* as a *"transfer of user interface research to end-user practice"*, which is an implementation of research on keyboard interaction. ShapeWriter is an early example of how application stores can be leveraged to get user reviews, and Zhai et al. present the insights they got from analyzing 556 user comments.

Testing early versions of applications as proof of concept can precede other ways of leveraging an application store, before continuing a line of work, to get first feedback from users.

Using Application Stores as Research Instruments

Application stores can also be leveraged as a tool to the inform design of a system following a user-centered approach, as the following papers suggest. As such, works of this category focus on improving the design of a system rather than investigating dedicated research questions.

McMillan et al., 2010, describe how they collected a maximum number of participants combining qualitative and quantitative feedback for redesign of a game called Yoshi. Their approach was to inform the design of the application itself based on a large amount of feedback collected from end users through in-game feedback measures and by contacting some players through social networks and interviewing them over VoIP or telephone.

Karpischek et al., 2012, deployed an application for sharing product reviews based on barcode scans. First user comments were analyzed to improve the design of the system. The authors describe their deployment on the application store as a research tool to learn about customer-product interactions.

Henze and Pielot, 2013, discuss the research tool of studies through the application store as a means for increasing external validity, simply because large samples of worldwide participants, who will use the application in their natural contexts, can be reached.

Implementing Dedicated Research Questions

Another way to leverage application stores for research studies is to conduct studies that alternatively also could have been conducted within controlled lab studies. As such, this category's goal of using an application store is to study a distinct research question, with special focus on upscaling the size of a sample, reaching a wider range of participants, or bringing the apparatus into the natural context of the participants.

A good example for studies whose research questions are implemented within applications are Henze's studies of off-screen visualizations on small screens. In Henze and Boll, 2010, the study compared stretched arrows, scaled arrows and Halos with a task of finding 10 randomly distributed points on a map. In Henze et al., 2010, the focus is more on testing the method. To test such "experiments in the wild [they used] the well-defined off-screen problem", which they already had used previously, but this time with the study task embedded into a game.

Budde and Michahelles, 2010, study the question of whether a social gaming approach can be used to build a crowd-sourced database of product information. During a rather short study period of 17 days, the authors were able to provide evidence that it is possible to motivate users to start submitting information, but the authors could not show that a full database could be created.

Henze et al., 2011, studied the touch performance of smartphones with regard to precision. With a task where users had to tap on elements of different size and position, they replicated a known offset in touch accuracy. Most importantly, they found that this offset is systematically skewed. By releasing an updated version of the game, they showed that the resulting touch error could be compensated for, and touch performance could be improved.

Sharazi et al., 2011, investigate whether non-verbal iconic user interfaces are reasonable for real-time opinion sharing in the case of television programs. They released an application for judging football games, and studied its use during a football world championship. Using data from 925 users they revealed that the interface serves its purpose and can even create a sense of connectedness among users in different locations.

Informing an Understanding of Smartphone and Application Usage

Researchers also apply the instrument of research through the application store to learn about more general aspects of smartphones and application usage. As such, in this fourth category we present works that conduct research through the application store because their research questions are inherently bound to this ecosystem. Some of the works presented previously on recommender systems for mobile applications could also be put into this category.

Watzdorf and Michahelles, 2010, investigate the accuracy of positioning to understand how accurate such data is, based on data collected through a commercial iPhone application. They find that different positioning technologies (GPS, WiFi-based, Cell-ID) provide different degrees of accuracy. Based on their findings they suggest which positioning technology should be used for which requirements.

In Böhmer et al., 2011, we have presented AppSensor: a virtual sensor for tracing mobile application usage on Android devices. By deploying this sensor to the Android application store we were able to collect data from more than 4,100 users over a period of more than 4 months. We found that the average session with an application lasts less than one minute, even though users spend almost an hour a day using their phones. Further, we showed that news applications are most popular in the morning and games are at night, but communication applications dominate through most of the day. We also analyzed chains of application usage and find that despite the variety of available apps, communication apps are almost always the first used upon a device's waking from sleep.

Do et al.,2011, investigate mobile application usage trails of 77 users over periods of 9 months. Essentially, the contextualized data set they collected by means of a background logging service contained application usage logs, location data, and Bluetooth data. Among other results, they find that application usage correlates with users' semantic locations; e.g. at holiday locations people are likely to use their camera and map applications.

Ferreira et al., 2011, study smartphone-charging behavior through a study in the large. Based on data from more than 4000 people they found that people charge their phones in long periods over night, and short bursts during the day, while in the latter case people use USB charging (connected to a computer via USB) more often than a AC power outlet. On average, people leave their devices plugged in for 4 hours and 39 minutes after the battery is full.

Girardello and Michahelles, 2010, investigate installation behavior of smartphone users and present AppAware: a recommender system that is based on people's overall application installations, uninstallations and updates. The system recommends applications based on their popularity, i.e. how many times an application was installed but not removed.

Informing the Research Approach

Lessons learned constitute the main contributions of a few papers conducted using the approach of research in the large. As such, they inform the approach itself and provide guidelines for other researchers using this approach.

Henze et al., 2011, report on the experience of five studies they have conducted in the large. One valuable insight is that findings cannot be generalized without knowing much about the sample, e.g. when they people reside only in one country. Secondly, when users have to opt-out from the study instead of opting-in, obviously more data can be collected, though this has to be aligned with legal and ethical aspects.

Kranz et al., 2013, study the adoption of near-field communication (NFC) technology (they released a game that motivates people to scan NFC tags they have avail- able) and investigate people's behavior with respect to updating applications. The main lessons are that short development cycles can support fast iterations on user feedback, visual appeal does attract users, marketing and maintenance is required to turn downloads into active use, multiple applications for the same research question could increase the sample size, and studies in the large are conditioned by user-side constraints (e.g. whether NFC is available).

Dey et al., 2012, present lessons learned from a study of people's battery charging habits observed over a 4-week study of more than 4000 users. They describe how they improved the application based on user feedback. The paper discusses ideas for running controlled studies by randomizing conditions based on device identifiers, and argues that maintaining a deployed system would be well supported because update cycles can be short. A downside that was experienced was that engineering efforts tend to be high when aiming for good reviews, which will impact the visibility of research applications among the large number of applications.

Tossel et al., 2012, explain nine constraints that need to be taken into account when using the method of smartphone logging for studies in the wild. These are related to the variables that are needed, whether data is potentially sensitive and requires privacy, the degree of obtrusiveness and whether the user will be interrupted for data collection, whether an interface is required for logging, whether participants' tasks will be natural or constructed artificial tasks, the type of technology used, who the participants are, where the study will take place and what the setting will be, and finally how long the study will be. The authors compare three studies they have conducted, and conclude that logging can be more accurate than self-reporting.

Last but not least, Coulton and Bamford, 2011, report on experiences with applications released to the *WidSet* platform, with two deployed applications with a total of more than 1.4 million users. Data collection was begun in October 2007, and — to the best of our knowledge — this constitutes the first and largest conducted study to be reported. Their lessons are that while value-added functionality can amplify popularity and usage, it may also impact usage behavior in a way that impacts the study. To the best of our knowledge, this is the only work reporting on the risk that the evolution of the chosen application store may strongly impact the research plan, as promotional actions might boost application installs; finally, the *WidSet* application store was taken offline, which put an end to the study.

Discussion

We cannot claim that in these five categories we present an exhaustive list of all works relating to the method of conducting research through the application store, but rather we wanted to exemplify the ways the method of conducting research through the application store can be leveraged.

We are extending the method of research through the app store that when aiming on informing an understanding of smartphone and application usage in general. Figure 2 shows three different approaches for conducting research through deployed applications. So far, studies mainly leveraged two possibilities: First, researchers began to implement study tasks within the application itself. Participants would contribute to the study by using the application (for instance McMillan et al., 2010; Henze et al. 2010; Henze et al., 2011; Kranz et al., 2013). Next, researchers began to leverage background processing and the APIs provided by the operating systems to collect data that allows describing the device and application usage. Thereby participants would continuously contribute to the study while the application would run in the background (e.g., Böhmer et al., 2011; Ferreira et al., 2011; Girardello and Michahelles, 2010). In this paper, we present an approach that actually incorporates the system's user interface instead of observing user behavior passively by means of logging. Therefore the participant would contribute to the study by switching from our application to the UI of his smartphone, while we are able to observe the transmissions the user makes.

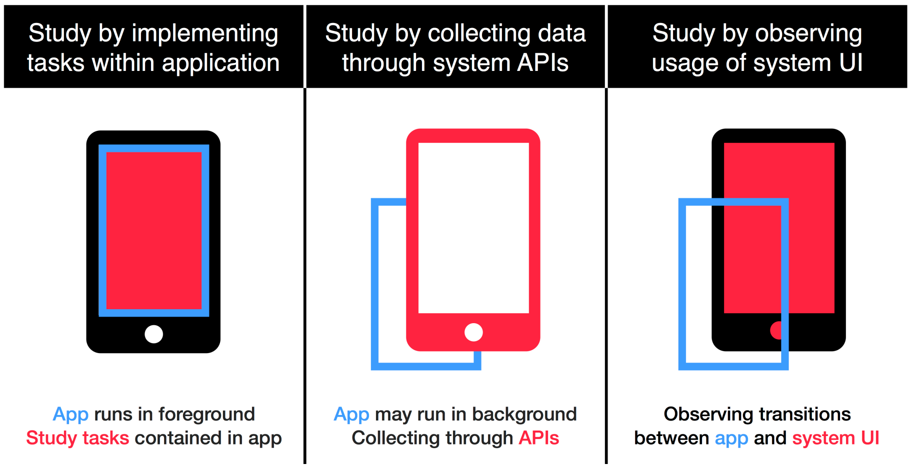

Figure 1. Three approaches for conducting research through the app store.

CASE STUDY ON THE SPEED OF APP LAUNCHING

In this section we present a case study that sets out to investigate how fast people can find applications on their smartphones to launch them. This study goes beyond the related work presented previously because we cannot encapsulate this study within one single application, but rather have to make the participant's original application launcher an integral part of the study.

Gaming the Android OS

The design and release of games has proven to be a vital research tool for running research studies through mobile application stores. Games are naturally good candidates for high degree of user interest and engagement, which is required to successfully collect enough data for meaningful results. Pioneering work by McMillan et al., 2010, leveraged a game to study the capability of research in the large as a method itself. Henze, 2012, was able to study several questions by releasing a simple game. However, so far such game-based studies investigate questions that were encapsulated and controlled within the released application itself.

In contrast, in the approach presented in this paper, we designed a game that becomes an inherent part of the users smartphone itself and makes it an essential part of the game play. As a result of this approach, the UI of a user's smartphone itself becomes the apparatus of our study.

Work Related to Case Study

Besides the work we presented previously on the method of conducting research through the application store, there is also a number of other works that relate to the research question of our case study.

Ziefle and Bay, 2004, investigated people's abilities to build mental models of their hierarchical mobile phone menus. They found that younger people have a better mental model of their mobile phones' menus. Further, they also found that awareness of the menu's structure increases navigation performance. Ziefle et al., 2007, particularly investigate young and old users navigating menus on PDAs with or without hyperlinks. From a 20-person study measuring performance when navigating websites, the paper reports that older users were less efficient when hyperlinks were available. Building on mental models of phone menus, Gustafson et al., 2011, build an imaginary phone that is operated by mimicking the interaction with a smartphone's interface on the palm of a hand. They found that their study participants knew the positions of 64% of home-screen icons by heart, and could even more accurately recall the positions of applications used daily at a success rate of 75%.

Böhmer and Krüger, 2013, studied how people organize their launcher menus. Based on a collection of screenshots and a qualitative study of people's concepts for arranging icons, they found that people apply five different strategies for arranging icons within their launcher menus: most people either arrange icons based on the usage frequency of the applications, or based on the relatedness of the applications. Smaller fractions of users arrange icons for reasons of usability, for reasons of aesthetics or they apply external concepts like the alphabet.

Hang et al., 2013, studied patterns of people launching applications on their Android smartphones. They conducted a study by instrumenting their participants' smartphone launchers with an instrumented version that enabled them to log how people launch their apps (. They also compared navigation times and found that people can launch apps from their dock fastest, followed by home-screen menus. In comparison launching apps form folders and app drawers were the slowest.

Study Setup

We have built a simple game to study how well fast people can find icons in their launcher menus, called rappidly. The objective of the game is to start apps as fast as possible. Therefore users can start new laps, and in each lap he has to find a new random app that he has installed, as Figure 2 shows. When a user decides to start a new lap he will see a countdown (Figure 2a-c), and finally the icon and name of an application with a started stop watch (Figure 2d). From this screen the user has to search for the shown app (here: *Angry Birds*) in own launcher menu and launch it from there (Figure 2e). As soon as he has clicked the app icon he will return to our game and see how long it took him to find the app icon (here: 9.666 seconds; Figure 2f). From here he can decide to start a new lap or post his lap stopping time to Facebook.

Our approach of making the original smartphone an inherent part of the gameplay and of the study design as such becomes clear at the transitions from Figure 2d) to e) when the user leaves our game application to search for the icon of the requested application within his menu, and at the transition from Figure 2e) to f) when the user clicks on the original icon of the target application within his launcher menu, but instead of launching the original application he returns to the game that we implemented.

We enriched the gameplay with elements of social networking to make the game itself more competitive for gamers, and to collect demographics on our players (when they connect to Facebook).

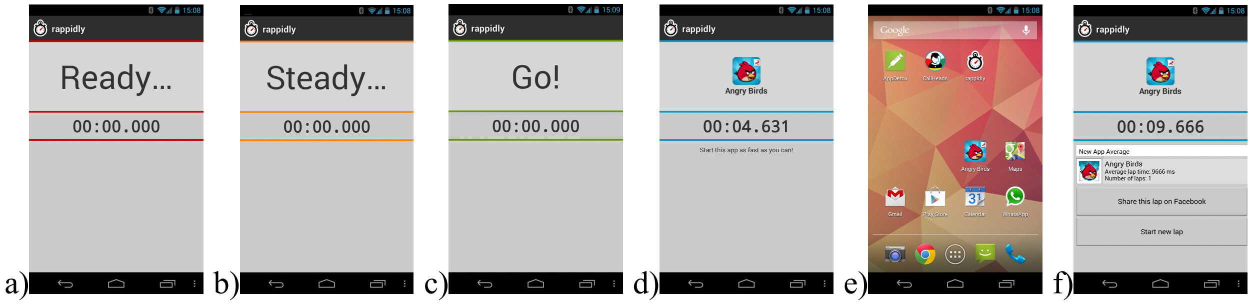

Figure 2. Gameplay and the screens a user will see when starting a new lap in our game: (a-c) count-down for launching the app, (d) instructions which app to start, (e) searching for the app in smartphone launcher menu for launching it, (f) final scoring screen when app was launched.

Study Findings

Within the game we implemented means for measuring the user's performance when playing the game. Essentially we kept track of a user id, a lap's app, and how long it took him to find an application until he was able to launch it. This data was collected and send to our servers when a user shared it publicly on Facebook (otherwise not). We released the first version in June 2012, and since then it was installed more than 590 times. So far, 41 users have contributed data from 304 laps played with 164 different apps. One user played 7.41 laps mean (SD 19.90, min 1, max 128). On average, in the laps they played people took 5.66 seconds to find an app (SD 4.46). On the lower end (min 1.20, below 2 sec) in particular we can find laps on note taking apps (e.g. Evernote), which one usually needs fast access to when noting down a quick thought. We found the maximum search time for somebody searching for Facebook (40.65 sec).

DISCUSSION AND CHALLENGES

Gaming the OS to Inform Design

Our approach of making the operating system itself the game court of the game we designed allows us to study aspects, which are not inherent in our application, but are customized and created by the very end-user himself outside of our direct control through the app. Therefore, this study can be seen as a quasi-experimental design. We think that this approach can be extended to study other aspects of smartphone UI design leveraging the method of research through the app store. For example, by adopting a game on memorizing sequences of items (e.g., Wikipedia, 2013) to smartphones one could test for how well people can memorize the stack of recently used applications, and an understanding of how people memorize recent apps could also help to improve application launching. Speaking more general, the stimulus for a study task needs to be given and controlled by the released app, while the task is conducted on the users smartphone out of direct control, and the task needs to be observable either directly or through a proxy measure.

Information on Navigation Paths and Menu Structure

In our current implementation of the game we cannot retrieve any information on the path a user takes for navigating to an app icon after leaving our application. Such tracing would only be possible by applying more intrusive techniques (e.g. logging raw touch events), or getting a handle on the launcher menu itself. The latter would also allow querying how a user has arranged his icons (Böhmer and Krüger, 2013). However, so far we were not able to implement such functionality and by deploying a new launcher app with the game itself we would destroy the menu structures and mental models that users created, thus bias search performance. The approach presented by Hang et al. 2013, where they reconstructed the user's launcher menu after replacing the original launcher with an instrumented one, would not scale to a dimension where one could conduct the study through an application store.

Data Cleaning and Ground Truth

As a common shortcoming of studies conducted through the app store in a rather uncontrolled setting, we cannot know about some circumstances of the user when playing the game. Obviously, we cannot know if the user is playing the game with full attention, what might result in lower performance if not the case. Further, we do not know if using *rappidly* itself has an impact on how a user organizes his launcher menu; e.g., we cannot know if *rappidly* is such engaging for a user that he optimizes his icon arrangement for best performance in the game instead of common reasons (Böhmer and Krüger, 2013). For instance, it remains unclear why somebody took more than 40 sec to launch Facebook in our sample.

Why in the Large?

We implemented our research question into a mobile application and released it on the application store, because we wanted to study how different properties of devices (like screen size, resolution) and launcher menus (e.g. swipe menu vs. scroll menu) would affect participant's performance of searching for icons. The work presented in this paper falls in the line of our research on understanding and supporting people when customizing and using app launcher menus (Böhmer and Krüger, 2013; Parate et al., 2013). However, putting an application out to a mobile application store cannot be seen as a guarantee for reaching the number and breadth of participants one would expect. As discussed before, the researcher is out of control of this aspect. For rappidly we have reached a sample size which is sufficiently large enough to make first conclusions, but currently too small for providing evidence for any subtle effects between device models. However, the study of rappidly provided evidence that our approach of making the original user's smartphone part of the game is reasonable for studying them.

CONCLUSION AND FUTURE WORK

In this paper we presented a new approach of studying smartphone usage by making the user interface of the smartphone an integral part of the study apparatus. We presented a categorized overview of related studies done through the app store and grounded the method in the theory of quasi-experimental design. As a case study we presented the Android game *rappidly*: a game designed to study how fast people can launch mobile apps. We described preliminary results for informing the design of future smartphone launcher menus. Further, we discussed our approach of gaming the OS, which allows implementing research questions on the general design of smartphone UIs with the advantages of conducting research through the app store. Future work is on enlarging the user base of rappidly. Besides collecting data on how fast people can launch apps we will trace general app usage using our AppSensor (Böhmer et al., 2011). This will allow testing hypothesis like: the more often people launch apps, the faster they can find them; or: the more apps people have installed, the longer it takes them to find single apps.

References

Böhmer, M., and Krüger, A. (2013). A study on icon arrangement by smartphone users. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI ’13, pages 2137–2146, New York, NY, USA. ACM.

Böhmer, M., Hecht, B., Schöning, J., Krüger, A., and Bauer, G. (2011). Falling asleep with Angry Birds, Facebook and Kindle: a large scale study on mobile application usage. In Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, Mobile- HCI ’11, pages 47–56, New York, NY, USA. ACM.

Budde, A., and Michahelles, F. (2010). Product empire – serious play with barcodes. In Proceedings of Internet of Things, IOT 2010, pp. 1–7. IEEE.

Buddharaju, P., Fujiki, Y., Pavlidis, I., and Akleman, E. (2010). A novel way to conduct human studies and do some good. In CHI ’10 Extended Abstracts on Human Factors in Computing Systems, CHI EA ’10, pp. 4699–4702, New York, NY, USA, 2010. ACM.

Church, K., Smyth, B., Bradley, K., and Cotter, P. (2008). A large scale study of European mobile search behaviour. In Proceedings of the 10th international conference on Human computer interaction with mobile devices and services, MobileHCI ’08, pp. 13–22, New York, NY, USA. ACM.

Coulton, P., and Bamford, W. (2011). Experimenting through mobile ’apps’ and ’app stores’. Int. J. Mob. Hum. Comput. Interact., 3(4):55–70, 2011.

Dey, A. K., Ferreira, D., and Kostakos, V. (2012). Lessons learned from Large-Scale user studies: Using android market as a source of data. Int. J. Mob. Hum. Comput. Interact., 4(3):28–43.

Do, T. M. T., Blom, J., and Perez, D. G. (2011). Smartphone usage in the wild: a large-scale analysis of applications and context. In Proceedings of the 13th international conference on multimodal interfaces, ICMI ’11, pages 353– 360, New York, NY, USA, 2011. ACM.

Ferreira, D., Dey, A. K., and Kostakos, V. (2011). Understanding human-smartphone concerns: a study of battery life. In Proceedings of the 9th international conference on Pervasive computing, Pervasive’11, pages 19–33, Berlin, Heidelberg, 2011. Springer.

Gustafson, S., Holz, C., and Baudisch, P. (2011). Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In Proc. of UIST, pp. 283–292, New York, NY, USA, 2011. ACM.

Hang, A., De Luca, A., Hartmann, J., Hussmann, H. (2013). Oh App, Where Art Thou? On App Launching Habits of Smartphone Users. In Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI 2013). Munich, Germany, August 27-30 2013.

Girardello, A. and Michahelles, F. (2010). AppAware: which mobile applications are hot? In Proceedings of the 12th international conference on Human computer interaction with mobile devices and services, MobileHCI ’10, pages 431–434, New York, NY, USA. ACM.

Henze, N. (2011). How to do MobileHCI Research in the Large? Tutorial at MobileHCI 2011, http://de.slideshare.net/NielsHenze/how-to-do-mobilehci-research-in-the-large, last accessed on 16.08.2013.

Henze, N. (2012). Hit it!: an apparatus for upscaling mobile HCI studies. In: Proc. of CHI ’12 Extended Abstracts.

Henze, N. and Pielot, M. (2013). App stores: external validity for mobile HCI. ACM Interactions, 20(2):33–38, 2013.

Henze, N., and Boll, S. (2010). Push the study to the app store: evaluating off-screen visualizations for maps in the android market. In Proceedings of the 12th international conference on Human computer interaction with mobile devices and services, MobileHCI ’10, pp. 373–374, New York, NY, USA. ACM.

Henze, N., Pielot, M., Poppinga, B., Schinke, T., and Boll, S. (2011). My app is an experiment: Experience from user studies in mobile app stores. International Journal of Mobile Human Computer Interaction, 3(4):71–91.

Henze, N., Poppinga, B., and Boll, S. (2010). Experiments in the wild: public evaluation of off-screen visualizations in the android market. In Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries, NordiCHI, pp. 675–678, New York, NY, USA, ACM.

Henze, N., Rukzio, E., and S. Boll. (2011). 100,000,000 taps: analysis and improvement of touch performance in the large. In Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI ’11, pp. 133–142, New York, NY, USA. ACM.

Hilbert, D. M. and Redmiles, D. F. (1998). An approach to large-scale collection of application usage data over the Internet. In Proceedings of the 20th international conference on Software engineering, ICSE ’98, pp. 136–145, Washington, DC, USA, 1998. IEEE Computer Society.

Karpischek, S., Michahelles, F., and Fleisch, E. (2012). my2cents: enabling research on consumer-product interaction. Personal Ubiquitous Comput., 16(6):613– 622, 2012.

Kranz, M., Murmann, L., and Michahelles, F. (2013). Research in the large: Challenges for large-scale mobile application research – a case study about nfc adoption using gamification via an app store. International Journal of Mobile Human Computer Interaction, 5(1):45–61.

McMillan, D., Morrison, A., Brown, O., Hall, M., and Chalmers, M. (2010). Further into the wild: Running worldwide trials of mobile systems. In P. Floréen, A. Krüger, and M. Spasojevic, editors, Pervasive Computing, volume 6030 of Lecture Notes in Computer Science, chapter 13, pp. 210–227–227. Springer.

Oulasvirta, A. (2012). Rethinking experimental designs for field evaluations. Pervasive Computing, IEEE, 11(4):60–67, 2012.

Parate, A., Böhmer, M., Chu, D., Ganesan, D., and Marlin, B. M. (2013). Practical prediction, prefetch, and prelaunch for faster access to applications on mobile phones. In: Proc. of UbiComp 2013.

Poppinga, B., Cramer, H., Böhmer, M., Morrison, A., Bentley, F., Henze, N., Rost, M., and Michahelles, F. (2012). Research in the large 3.0: app stores, wide distribution, and big data in MobileHCI research. In Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services companion, MobileHCI ’12, pp. 241–244, New York, NY, USA, 2012. ACM

Rogers, Y. (2012). HCI Theory: Classical, Modern, and Contemporary (Synthesis Lectures on Human-Centered Informatics). Morgan & Claypool Publishers, 1 edition.

Shadish, W. R., Cook, T. D., and Campbell, D. T. (2001). Experimental and Quasi-Experimental Designs for Generalized Causal Inference. Cengage Learning, 2 edition.

Shin, C., Hong, J.-H., and Dey, A. K. (2012). Understanding and prediction of mobile application usage for smart phones. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp ’12, pages 173–182, New York, NY, USA. ACM.

Shirazi, A. S., Rohs, M., Schleicher, R., Kratz, S., Müller, A., and Schmidt, A. (2011). Real-time nonverbal opinion sharing through mobile phones during sports events. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI ’11, pp. 307–310, New York, NY, USA, 2011. ACM.

Tossell, C. C., Kortum, P., Shepard, C. W., Rahmati, A., and Zhong, L. (2012). Getting real: a naturalistic methodology for using smartphones to collect mediated communications. Advances in Human-Computer Interaction. http://www.hindawi.com/journals/ahci/2012/815972/, last accessed 10.07.2013.

von Watzdorf, S., and Michahelles, F. (2010). Accuracy of positioning data on smartphones. In Proceedings of the 3rd International Workshop on Location and the Web, LocWeb '10, New York, NY, USA. ACM.

Wang, G. (2009). Designing smule's ocarina: The iPhone's magic flute. In Proceedings of the International Conference on New Interfaces for Musical Expression, pp. 303—307. http://www.nime.org/archive/?mode=ylist&y=2009, last accessed 08.07.2013.

Wikipedia. (2013), Touch Me (arcade game), http://en.wikipedia.org/wiki/Touch_Me_(arcade_game), last accessd on 16.08.2013

Zhai, S., Kristensson, P. O., Gong, P., Greiner, M., Peng, S. A., Liu, L. M., and Dunnigan, A. (2009). Shapewriter on the iphone: from the laboratory to the real world. In CHI '09 Extended Abstracts on Human Factors in Computing Systems, CHI EA '09, pp. 2667–2670, New York, NY, USA, 2009. ACM.

Zhang, C., Ding, X., Chen, G., Huang, K., Ma, X., and Yan, B. (2012). Nihao: A predictive smartphone application launcher. In Uhler, D., Mehta, K., and Wong, J., editors, Mobile Computing, Applications, and Services, volume 110 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pages 294–313. Springer.

Ziefle, M., and Bay, S. (2004). Mental models of a cellular phone menu. comparing older and younger novice users. In S. Brewster and M. Dunlop, editors, Mobile Human-Computer Interaction - MobileHCI 2004, volume 3160 of Lecture Notes in Computer Science, chapter 3, pp. 25–37. Springer, Berlin, Heidelberg, 2004.

Ziefle, M., Schroeder, U., Strenk, J., and Michel, T. (2007). How younger and older adults master the usage of hyperlinks in small screen devices. In Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '07, pp. 307–316, New York, NY, USA. ACM.