

Chapter 2

Background and Related Work

This chapter provides the background and related work for this thesis. The notion of notifications as it relates to the field of mobile Human-Computer Interaction (mHCI) is introduced, and the applied research methodology for the studies presented in Chaps. 4–6 is outlined. Finally, a literature review on related work is provided.

2.1 Mobile Human-Computer Interaction

Human-Computer Interaction (HCI) “is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them” [73]. This dissertation focuses on the domain of smartphones and mobile applications, i.e., a subset of such interactive computing systems. When referring to this limited definition, the term mobile Human-Computer Interaction (mHCI) is used.

2.1.1 *Limitation of Attention*

Before the age of smartphones and mobile application stores, notifications on mobile devices were mostly limited to incoming phone calls, messages, and system warnings. With the provision of notification services to third-party mobile application developers, the number of applications utilizing this functionality rapidly increased. Today, notifications are a core feature of many applications, pushing information to users and competing for users’ attention.

Davenport and Beck [41] defined the concept of attention as “focused mental engagement on a particular item of information. Items come into our awareness, we attend to a particular item, and then we decide whether to act”. This definition already hints at the limited availability of attention as a resource, which coined the

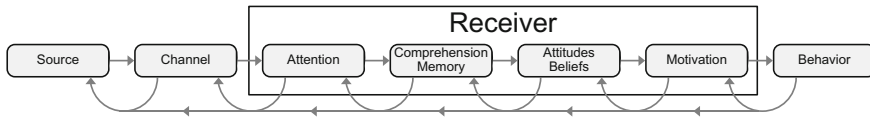


Fig. 2.1 Communication-Human Information Processing (C-HIP) model (adapted from Conzola and Wogalter [35], used with permission)

term *attention economy* [41] as an approach to the management of information in the field of economics.

Reports highlight that “media has always been at war for [...] attention — and has always come up with new ways to win it”.¹ Further, “notifications [on mobile phones] represent the future access and discovery point for mobile services — [...] notifications will be the starting point [...] for all of the interactions on your phone”.²

Information overload, as a consequence, is already a prominent example of problems associated with the notion of attention as a scarce commodity. Therefore, an investigation of notification on smartphones seems appropriate.

2.1.2 Information Processing

In order to structure the stages involved as information flows from a sender to a receiver, concepts of Wogalter, DeJoy, and Laughery’s communication-human information processing (C-HIP) model [169] and Cranor’s human-in-the-loop framework [38] are adopted. These concepts will be addressed in Chaps. 4–6, when studying factors that influence the acceptance of notifications on smartphones.

Communication-Human Information Processing Model

Wogalter, DeJoy, and Laughery [169] introduced the C-HIP model in the field of warning research. It structures the stages involved as information flows from a source to a receiver, eventually producing behavior. The model is based on prior work in the field of communication theory by Lasswell [93] and Shannon [143], and adopts the conceptual stages of *source*, *channel*, and *receiver*. Among other potential elements, Wogalter, DeJoy, and Laughery did not include the concept of noise (cf. [143]), e.g., background noise for auditory warnings, which leads to a simplified and idealized model (cf. Fig. 2.1).

In the frame of this thesis, i.e., with respect to mobile applications, the stages of the C-HIP model may be described as follows:

¹<http://www.wired.com/2014/12/new-media-2/> (last accessed: 2016-03-06).

²<http://techcrunch.com/2015/04/21/notifications-are-the-next-platform/> (last accessed: 2016-03-06).

SOURCE The originator of the information, e.g., a mobile application publisher. Characteristics of the source, e.g., perceived credibility, influence the effectiveness of the given information [169].

CHANNEL The way a message is transmitted from the source to the receiver. It may involve different modalities, and thus include more than one sensory modality (e.g., visual, auditory, and/or tactile).

The receiver, as depicted in Fig. 2.1, represents a superordinate category that incorporates a number of information processing stages (cf. [169]):

ATTENTION The receiver's first operation. Capturing and maintaining attention is influenced by factors such as characteristic of the message itself or environmental variables (e.g., physical location, ambient noise conditions, or stress level). Maintenance of attention is required upon capture, as this stage allows for information extraction. This in turn is facilitated by factors including legibility and brevity.

COMPREHENSION AND MEMORY This stage describes the factors that facilitate understanding of the presented information, including whether a message and pictorial symbols can be understood by the receiver.

ATTITUDES AND BELIEFS Beliefs refer to the receiver's knowledge of a topic. Attitudes are similar to beliefs, but have a greater emotional impact.

MOTIVATION The motivation to carry out the intended behavior is influenced by factors such as cost of compliance, explicit consequences, and anticipated severity.

Finally, if motivated sufficiently, individuals carry out the desired behavior the information was directed at, for example opening a notification from a mobile application.

This process, although described as linear, includes feedback loops to earlier stages. Conzola and Wogalter [35] give an example of an influence of the comprehension stage on attention stage: "when a warning stimulus becomes habituated over time from repeated exposures, attention is less likely to be allocated to the warning on subsequent occasions". Wogalter, DeJoy, and Laughery [169] acknowledged limitations of the C-HIP model, as it requires the processing of each stage. However, it is possible that stages may be bypassed, i.e., not all stages may be necessary for behavior to occur. For example, the mere stimulus might serve as a cue that elicits the desired result for someone with prior knowledge.

Human-in-the-Loop Security Framework

The C-HIP model has been introduced in 1999. With emerging research in related fields, the model has been adapted to fit particular contexts. Cranor's human-in-the-loop framework [38], depicted in Fig. 2.2, is based on the C-HIP model. Cranor further added components that are typical in a computer security context, i.e., *capabil-*

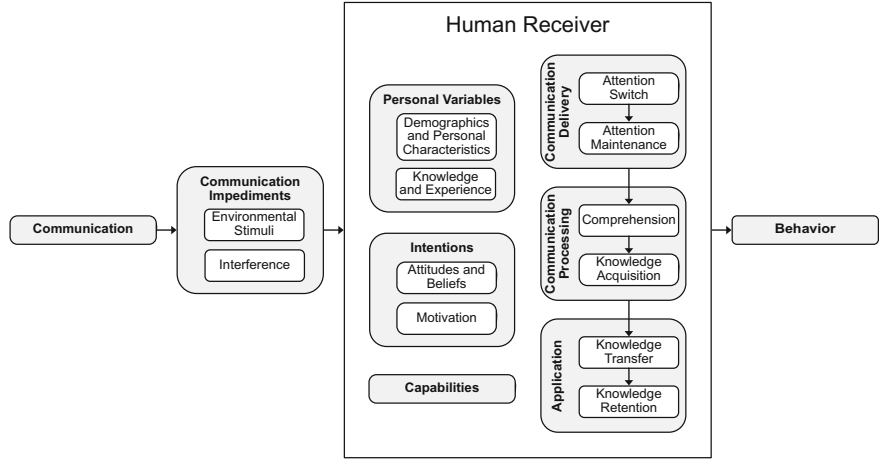


Fig. 2.2 The human-in-the-loop security framework (adapted from Cranor [38], used with permission)

ities and *interference* (communication impediments). Further, the model is modified to apply more generally to the five steps of computer security communications, as distinguished in [38]: warnings, notices, status indicators, training, and policies. In contrast to the C-HIP model, the human-in-the-loop framework explicitly includes the concept of noise, referred to as communication impediments.

The psychological processes involved in paying attention to information messages, grasping their meaning, and deciding to comply with them haven’t changed substantially, even in the digital realm [22]. Thus, stages of the C-HIP model and the human-in-the-loop framework will be addressed in the studies presented in Chaps. 5 and 6, for example *attitudes and beliefs* when studying acceptance of notification permission requests in Chap. 5, *personal variables* when investigating ties to stress due to information overload in Chap. 6, or *motivation* when examining framing and motivators in Sect. 6.3.

2.2 Technical Background

This section provides the background on technical aspects that need to be considered when dealing with notifications on smartphones.

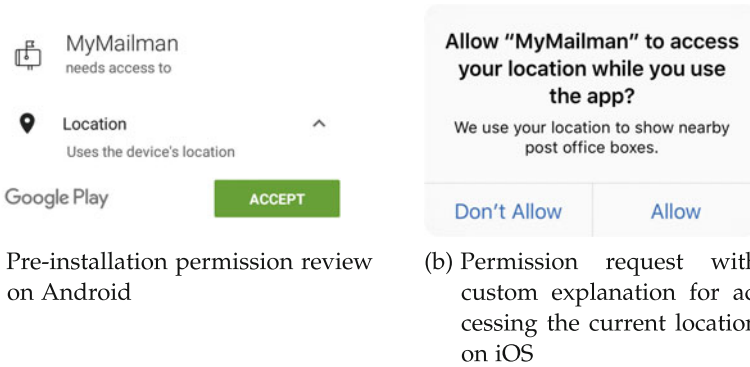


Fig. 2.3 Permission requests for accessing location on Android and iOS. Requested permissions are shown before installation on Android, whereas iOS presents permission requests at the time an app wants to access it for the first time

2.2.1 *Permission Models on Smartphones: Accessing Protected Resources*

Applications on Android and iOS run in a so-called application sandbox,³ i.e., by default, they can only access a limited set of system resources. Access to protected resources, e.g., camera functions, location data, or notifications, is handled by the operating system. Android and iOS distinguish themselves in terms of how access to these resources is requested. Android requests user approval at the time the user wants to install an application (cf. Fig. 2.3a). As a consequence, the app will only be installed if all requested permissions are granted, using a single confirmation dialog. On the contrary, iOS asks the user for permission to a protected resource at the time an app wants to access it for the first time (cf. Fig. 2.3b), using one request for each permission.

Permissions, once granted or denied, remain valid until the user either changes them manually using the system settings panel, or uninstalls the application. Since the release of iOS6⁴ in 2012, Apple allows developers to specify why an application needs access to privacy related resource, i.e., access to Bluetooth peripherals, calendar data, contacts, location, reminders, and the photo library. In the succeeding release, iOS7 in 2013, requests for accessing the microphone and camera were added.

³Application security. Android: <https://source.android.com/security/overview/app-security.html>, iOS: https://www.apple.com/business/docs/iOS_Security_Guide.pdf.

⁴Apple Developer Documentation: <https://developer.apple.com/library/mac/documentation/General/Reference/InfoPlistKeyReference/Articles/CocoaKeys.html>.

2.2.2 Mechanisms for Notifications on Smartphones

There are diverse types of notifications on smartphones, targeting different use cases. In the frame of this dissertation, a distinction is drawn between *system notifications*, i.e., those that are not visible to the user, and so-called *user notifications*. This work focuses on user notifications that enable an app to make users aware of events such as an incoming message, an impending calendar event or breaking news. For this reason, unless otherwise stated, the term *notification* will refer to *user notification* throughout this thesis.

User notifications may be further categorized into *local* and *remote* notifications, where local notifications are scheduled and sent by an application itself (e.g., an impending calendar event) and do not require an active internet connection, and remote notifications arrive from outside the device (e.g., a message). Whether local or remote in origin, user notifications look the same when presented by the operating system.

Technical Procedure

Remote notifications, also known as push notifications, typically involve three entities: (i) the system, i.e., the operating system and an app, (ii) a third party server, e.g., the back-end of an app publisher, and (iii) a push notification provider. In case of the two major smartphone operating system providers, Apple⁵ and Google,⁶ both also provide a push notification service (i.e., Apple Push Notification service (APNs) and Google Cloud Messaging (GCM)).

Before actually receiving remote notifications, an app has to register for these with the push notification provider, as illustrated on a high-level in Fig. 2.4 (steps 1–5). First, the app tells the operating system to register for receiving notifications (step 1), which in turn contacts the push notification provider (step 2). Upon success, the push notification provider returns a device token (step 3), which is then forwarded to the app (step 4). Finally, the app sends this token, in addition to an identifier associated to the device or user, to a third party server (step 5), which does user management and scheduling of notifications. In production environments, this third party server may usually consist of multiple servers, but is depicted as one in this figure for the sake of simplicity.

After registering for remote notifications, the third party server may send notifications to the device by contacting the push notification provider with the device token and a message (step a). The push notification provider then sends the message to the associated device (step b), which presents a notification to the user. When the user taps the notification, the message is passed to the associated app (step c).

⁵<https://www.apple.com/ios>.

⁶<https://www.android.com>.

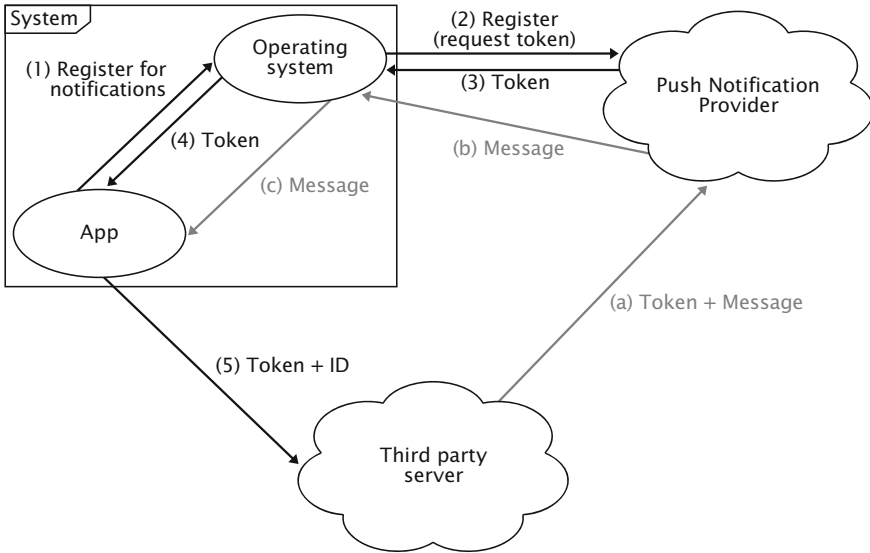


Fig. 2.4 High-level illustration of the process for registering for notifications (steps 1–5), and delivery of a notification from a sender to a device (steps a–c)

Notification Handling

Whereas the depiction in Fig. 2.4 focuses on technical components, Fig. 2.5 shows the interaction between the system and a user for notification management. The first part of this sequence diagram — requesting permission to send notifications — is present only on devices running Apple’s iOS (cf. Fig. 2.6). This process is initiated by step 1 in Fig. 2.4 (an app wants to register for notifications) and, given permission by the user, followed by step 2 (registration with a push notification provider). Once an app is registered for receiving notifications, the system can present notifications to the user, who may engage by interacting with the notification. The user may also change settings for notifications on the system level (through the system settings app), or settings provided by an application itself.

Notification Indicators

Notifications can display an alert message (cf. Fig. 2.8b) or, in case of Apple’s iOS, additionally badge the application’s icon (Fig. 2.8a). They can also play a (custom) sound or cause a vibration. Additionally, some vendors of Android phones offer the possibility of lighting a light-emitting diode (LED) as an indicator of a notification.

The state of the device determines the style a notification is presented (cf. Fig. 2.7): in case the device is not in active use and the display is locked, notifications are listed on the lock screen, as shown in Fig. 2.8b (iOS). The user may swipe (iOS) or tap (Android) a notification to launch the associated app directly from the lock screen, or may unlock the device and utilize the notification center/drawer where notifications

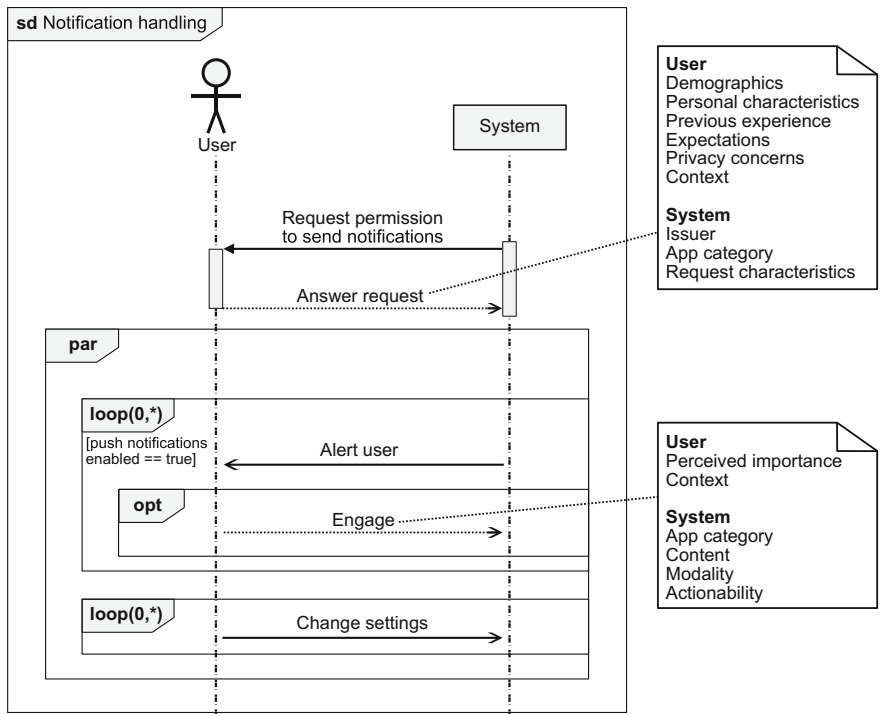
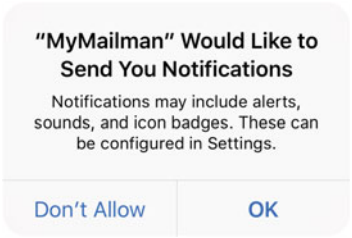


Fig. 2.5 Sequence diagram showing the interaction between system and user for permission requests, notifications and setting changes. Notes cover possible influencing factors as derived from the C-HIP model and the human-in-the-loop framework that will be addressed in Chaps. 5 and 6

Fig. 2.6 Default push notification permission request on Apple’s iOS



are stored for later review. Upon launching an app by means of a notification, information associated to the notification is passed to the app, which may then present further details or load remote content. It is noteworthy that an app does not receive information on pending notifications, if it is launched by tapping the application icon instead of the notification.

If, however, the application receiving a notification is already in active use, the notification payload is passed directly to the application, which is responsible for presenting new content or refreshing the user interface (UI).

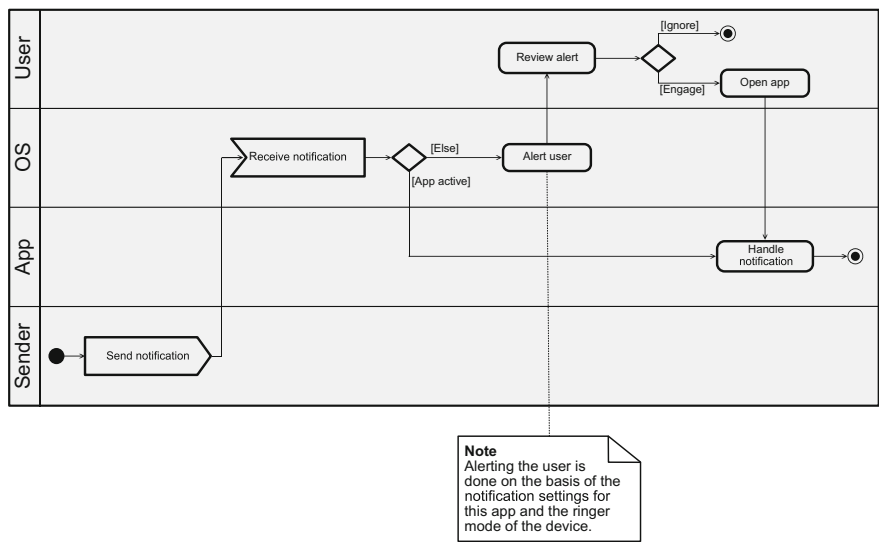


Fig. 2.7 Activity diagram for sending and receiving notifications

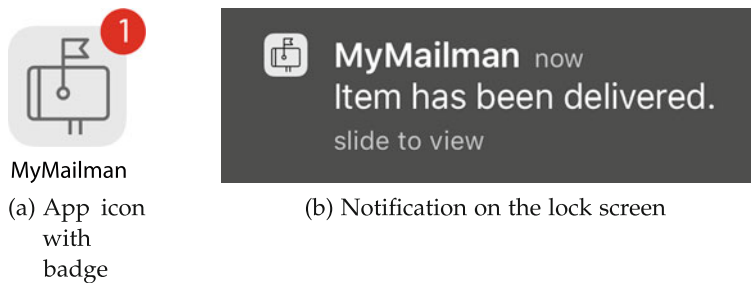


Fig. 2.8 Indicators of a new notification (Apple iOS)

Factors influencing (a) the acceptance of notification permission requests, (b) receptivity to notifications, and (c) changes in notification settings will be investigated in this thesis and discussed in Chaps. 5 and 6.

2.3 Research Methodology

Very early systems in the domain of ubiquitous computing (ubiquomp) (e.g., [119, 159]) were largely confined to controlled laboratory settings. Since then, there has been a general trend towards studies that take place in contexts more representative of the technologies’ eventual intended use. *GUIDE* [30], an early example developed by Cheverest et al., is a context-aware electronic tourist guide relying on

mobile computing technologies. Further into Weiser’s vision of ubiquitous computing, Joffe [80] and Bell et al. [13] investigated how mobile games interweave with everyday life.

Akin to mobile phones, which are carried everywhere, all the time [82], smartphones are “an accepted and integrated part of the lives of most people in Western countries” [130]. Being able to access data from smartphones with their sophisticated sensing capabilities in near real-time, Raento, Oulasvirta, and Eagle [130] further discuss that this “enables research approaches that have been either impossible or prohibitively expensive”.

The methodology of this work is inspired by the idea of utilizing applications on smartphones and leveraging them for research purposes. As pointed out by Abowd and Mynatt [1], rigorous evaluation of such systems “requires real use, and this, in turn, requires a deployment into an authentic setting”. Thus, most studies conducted in the frame of this dissertation follow a *deployment-based research* approach.

2.3.1 Leveraging Mobile Application Stores

The introduction of mobile application stores greatly simplified the distribution of mobile apps. Apple’s App Store opened in 2008 with 500 applications,⁷ followed by the Android Market in the same year with 50 applications.⁸ The number of available apps is increasing steadily and as of 2015, over 1.5 million apps are available in each of the two leading app stores.⁹ Unsurprisingly, the number of apps downloaded from application stores is increasing, too, passing 100 billion for iOS apps.¹⁰

The increasing popularity of smartphones and the proliferation of these distribution channels greatly diminished the burden of reaching users. This in turn provides researchers with the opportunity to receive a greater amount of quantitative data, from different geographic locations, as well as from a potentially more heterogeneous sample without the need to manually install software on users’ devices or the provision of a software repository.

Zhai et al. [172], working on text entry for mobile devices with touch screen-based UIs, were among the first to utilize a mobile application store for research purposes. McMillan et al. [107] described how they switched from using a public software repository to a mobile application store to inform the design of the application itself based on feedback from end-users. They further pointed out the benefit of ‘recruiting’ participants via application stores in terms of realistic conditions, as app installation

⁷<http://www.apple.com/pr/library/2008/07/10iPhone-3G-on-Sale-Tomorrow.html> (last accessed: 2016-02-09).

⁸<http://android-developers.blogspot.de/2008/10/android-market-now-available-for-users.html> (last accessed: 2016-02-09).

⁹<http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/> (last accessed: 2016-02-09).

¹⁰<http://www.statista.com/statistics/263794/number-of-downloads-from-the-apple-app-store/> (last accessed: 2016-02-09).

and use happens on a voluntary basis. As participants presumably use the system under investigation (i.e., a mobile application) on their own device rather than on one provided by the research team, a potential bias (e.g., being unfamiliar with the device) may not be assumed. Henze, Poppinga, and Boll [71] evaluated off-screen visualization techniques using a smartphone game published in an app store. They concluded that this approach may complement or even replace laboratory studies.

While previous research focused on evaluation of single applications, often concerned with usability aspects, Böhmer et al. [20] reported on general mobile application usage, using a logging framework (cf. Sect. 2.3.3) that tracks instances from all apps installed on the device.

In the frame of this dissertation, the development of a series of mobile applications is covered (see Chap. 3). A subset of these applications was deployed to mobile application stores and thus available to the public. This approach may be seen as a special case of *deployment-based research* with the purpose of *collecting data in the wild*, and allowing for *research in the large*. Feedback from users, either reviews in application stores, or answers to in-app questionnaires, was integrated into the iterative design process of the respective applications.

Experimental Design

Studying mobile application usage and evaluating such applications in the field is challenging. The paradigm of laboratory studies assumes perfect control of events and perfect randomization of subjects to conditions, whereas experiments in the field — by definition — involve events that are random, uncontrollable, and inadvertently biased by the intervention itself [120]. Oulasvirta [120] therefore proposed to rethink the experimental design of evaluations in a way that don't assume perfect control as laboratory studies do. *Quasi-experimentation* may serve as a principled alternative to laboratory-based testing, where the ability to identify and address disturbance variables determines the validity of the experiment.

Shadis, Cook, and Campbell [142] characterized an experiment as a study in which an intervention is deliberately introduced to observe its effects. Different subtypes of experiments result from a varying level of control [142]:

- Participants of a *randomized experiment* are assigned to a condition by chance, e.g., the toss of a coin or a table of random numbers.
- In a *quasi-experiment*, assignment of participants to conditions is not based on a random process. Instead, the assignment is by means of self-selection.
- The variation of a factor in a *natural experiment* occurs naturally, i.e., neither the participant, nor the experimenter is in control. Comparisons are made in retrospect.
- An *observational study* examines size and direction of a relationship among variables. As there is no treatment at all, it is also known as non-experimental study.

As for quasi-experimentation, the validity of the study needs to be carefully considered when conducting a natural experiment and thus moving further on the continuum of the level of control. This work employs randomized experiments (e.g.,

when studying notification permission request characteristics, see Sect. 5.4), quasi-experiments (e.g., when investigating receptivity based modality, actionability, and perceived importance, see Sect. 6.2), and observational studies (e.g., when studying receptivity to notifications, see Chap. 4).

Reflections on the Research Methodology

The prospect of potentially reaching a very large number of voluntary participants from around the world using the approach of publishing an app on a mobile application store is tempting. However, it constitutes challenges — different from those in a laboratory based experiment — that need to be taken into account when deciding upon the study design.

McMillan et al. [107] discussed that data from a large sample may be helpful with respect to statistical analysis, while potentially constituting an inhibitor to qualitative findings at the same time. They further suggested that the process of ‘recruitment’ may lead to more realistic conditions compared to traditional trials, while this advantage has to be weighed against issues as reduced knowledge of local context and culture.

Prototypes tested in laboratory experiments are typically not developed to the stage of market maturity. Mostly, this is not an issue, as they are introduced as research prototypes, which may crash once in a while. However, releasing an application to a wide audience via established distribution channels raises expectations [37], as it is presented as a finished product. High expectations on apps released in an app store were also found by Schleicher et al. [139], stating that “users [...] will give low ratings if an app is not polished or crashes during usage”. Thus, mobile applications, which are intended to be used for research purposes and released to the public, need to be developed under the objective and requirements of commercial ones.

Developing a system that handles hundreds or thousands of users simultaneously also poses a technical challenge. Whereas interaction logs may be stored on the device and manually transferred at a later point in time in a laboratory study, servers providing content and accepting log data are needed in a real world scenario. As pointed out earlier, the system as a whole needs to be functionally robust, which involves code review and, in case of a detection of a bug in a production version, rapid releases of new versions. Bentley, Basapur, and Hamilton [15] added legal issues for consideration: many mobile applications include open-source components that are licensed under different terms (e.g., Apache,¹¹ GPL,¹² or MIT¹³). These licenses may impose obligations, e.g., modifications to the source code that are distributed to the public must be made available in source code form (e.g., GPL).

Barely releasing an application in a mobile application store won’t result in many downloads, as it is not easily discovered. Thus, promotion of the app needs to be considered. However, Cramer et al. [37] brought a possible influence on research results into question. In general, drawing general conclusions from experiments with

¹¹ Apache license: <https://www.apache.org/licenses>.

¹² GNU General Public License: <https://www.gnu.org/licenses>.

¹³ MIT License: <https://opensource.org/licenses/MIT>.

mobile application requires caution in view of the fact that users of a specific app may not represent the general smartphone population, but people interested in the app. In the same line, although roughly two billion people use a smartphone,¹⁴ smartphone users may not represent the world population.¹⁵

In addition to the costs of promotion and operation of servers, time for developing and deploying such a production-ready system should not be underestimated [15, 37, 139]. Bentley, Basapur, and Hamilton [15] estimated that the preparation for large deployment was close to five times the cost in terms of design/development time.

Finally, ethical concerns arise when doing research with publicly available software, e.g., applications in a mobile application store. Henderson and Abdesslem [69] pointed out differences regarding privacy, regulation, ethics and culture, which may differ between countries. This issue will be discussed in greater detail in Sect. 2.3.4.

The approach of leveraging mobile application stores for research purposes has gained momentum in recent years and is employed by researchers of varying fields of study. Work related to this dissertation will be outlined in the following section.

2.3.2 *Methodically Related Studies*

Research on the design, evaluation and use of mobile applications gained momentum with the introduction of mobile application stores. Prominently represented in the fields of mHCI and ubicomp, social sciences and other related fields employ mobile applications for research purposes, too. Focusing on research in the large (e.g., through mobile application stores), Henze [70] proposed four categories to distinguish work in this field. In order to give an overview of methodically related work, this classification is adapted, without the restriction to large scale studies: (1) mobile applications as a proof of concept, (2) observing general aspects of smartphone and mobile application use, (3) implementing dedicated research questions with app-specific features, and (4) studying operating system-specific aspects, and mobile application stores as a research tool. These categories are not exclusive, and so studies may fall into more than one category. Further, this collection gives an overview on representative work in this field, but is not meant to be exhaustive. Work that is strongly related to the overall topic of this dissertation will be introduced in Sect. 2.4.

Proof of Concept

Predominantly early work in this field used mobile application and application stores to prove concepts, and to inform the design of an application by means of reviewing feedback submitted through the app stores' review functionalities.

¹⁴<http://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/> (last accessed: 2016-02-16).

¹⁵<http://www.nielsen.com/us/en/insights/news/2014/mobile-millennials-over-85-percent-of-generation-y-owns-smartphones.html> (last accessed: 2016-02-16).

Working on (virtual) keyboard text input methods for mobile devices, Zhai and Kristensson [171] proposed a method that augments stylus keyboarding with short-hand gesturing. Later on, they implemented the method for smartphones and published *ShapeWriter WritingPad* [172] on Apple's App Store. As a "direct transfer of user interface research to end-user practice", they collected feedback and released further revised versions.

Wang [157] presented *Smule's Ocarina*, a mobile musical artifact, which resembles an ancient flute-like instrument. Published as a mobile application for the iOS platform, they reached more than a million users and formed mobile phone orchestras around the world (e.g., the Stanford Mobile Phone Orchestra¹⁶ and the Michigan Mobile Phone Ensemble¹⁷).

General Aspects of Smartphone and Mobile Application Use

Researchers also study general aspects of smartphone and mobile application use in order to understand how such devices are used and embedded into every day life.

Böhmer et al. [20] developed a logging framework (*AppSensor*, cf. Sect. 2.3.3) that was integrated into a mobile app recommender system. Apart from basic descriptive statistics (e.g., average session length, number of installed apps), they also discussed contextual findings (app use over the course of the day). In a similar manner, Wagner, Rice, and Beresford [156] developed a standalone application that runs on the Android platform and captures a time-series of more than 200 different events. Over the course of two years, they were able to gather 53 billion data points from 12,500 users.

Studies on general smartphone use are not limited to those that include a deployed system at a large scale. Böhmer and Krüger [18] conducted a screenshot-based study in order to understand arrangement of app icons on smartphone screens. They found distinct concepts for arranging icons, based on functional relatedness.

Oulasvirta et al. [121] conducted a series of studies and identified a *checking habit*, which is characterized as a brief, repetitive inspection of dynamic content quickly accessible on the device. They further argued that this is a particular characteristic of smartphone use, which is not, at least in this dimension, observed for laptops.

Runyan et al. [136] used a mobile application, *iHabit*, that employed the experience sampling method (ESM)¹⁸ to study how first semester undergraduates spent their time. Among other findings, the authors reported that app use prompted greater self-awareness concerning time-management, which led to changes in behavioral patterns (spending more time studying early in the semester) for some participants.

Implementing Dedicated Research Questions with App-Specific Features

Besides observing general aspects of smartphone use, researchers also develop applications that are designed to investigate dedicated research questions.

¹⁶Stanford Mobile Phone Orchestra: <http://mopho.stanford.edu>.

¹⁷Michigan Mobile Phone Ensemble: <http://mopho.eecs.umich.edu>.

¹⁸The experience sampling method is used to gather systematic self-reports from individuals at random occasions in order to create an archival file of daily experiences (cf. [92]).

Henze, Poppinga, and Boll [71] explored different techniques for off-screen visualization. They found that the performance of visualization techniques depended on the number of objects visible on screen.

Bentley and Tollmar [16] used data gathered from their mobile application for several design iterations. Once each app update was online, they were able to verify effects of design changes.

Henze, Rukzio, and Boll [72] published a mobile game to investigate touch performance. They argued that touch positions were systematically skewed and proposed a compensation function that shifts the users' touches to reduce the error rate. Although investigated with a single application, the implications concerned the underlying operating system.

Gallego, Woerndl, and Huecas [59] implemented a context-aware restaurant recommender and evaluated two mobile user interfaces for delivering proactive recommendations. Comparing a widget-based to a notification-based version, they found that users found notifications more annoying and favored the widget-based UI instead.

Application Stores and Operating System-Specific Aspects

Mobile application stores and smartphone operating systems present a frame for research on use of mobile applications. As such, these are of interest for researchers, too.

The Android OS has a permission-based security system that handles access to sensitive information (address book, location) for third party applications. The risk evaluation is left to the user at the time of installation. Kraus, Wechsung, and Möller [88] investigated the effect of additional (statistical) information on communicating such risks. They showed that users were more likely to choose an app with fewer requested permissions, if additional information was available.

Similar to Android, Apple's iOS also handles access to sensitive information. The main difference is that it requests user permission at the time an app wants to access a resource, and that the developer may add justifying explanations for some requests (address book, calendar, location, and photos). Tan et al. [146] showed that requests that include explanations were significantly more likely to be approved.

Finally, utilizing mobile application stores to reach a large and potentially diverse group of users raises ethical issues. Related work on ethical considerations is further discussed in Sect. 2.3.4.

2.3.3 Methodically Related Frameworks

Chapter 3 presents the technical framework employed to study mobile application usage and notification handling. Studying user behavior on smartphones is not a new approach as such, and related systems are available to support research in this direction. Various frameworks for different purposes have been developed in the scientific

and commercial domain, which allow for collection of various sensor information on users' smartphones. With the extension of new sensors and capabilities, the number of available frameworks is increasing.

Raento et al. [131] released an early framework for the collection of contextual information for mobile phones in 2005: *ContextPhone*. It consists of modules providing access to device usage and sensor readings, and runs on smartphones using the Symbian operating system. Salovaara et al. [137] used the system to investigate unavailability for calls on mobile phones.

MyExperience by Froehlich et al. [58] combines objective and subjective data collection. In addition to passive logging of sensor readings and device usage, it collects self-reports from users applying the experience sampling method (ESM). Self-reports may be triggered either by sensor readings, or pre-defined rules. *MyExperience*, running on devices using Windows Mobile, also provides synchronization of collected data points with a remote server.

The *Funf Open Sensing Framework* [3] was originally developed by MIT and provides an extensible approach for collecting data from users' smartphones. Funf Journal,¹⁹ built using the Funf framework, is an Android application that may be used by researchers or "self-trackers" in a *Quantified Self* manner.

The *AWARE* framework by Ferreira, Kostakos, and Dey [53] is similar to *Funf* in the sense that it provides a plugin-based solution. Likewise, it offers personal data recordings for individuals.

Böhmer et al. [20] presented *AppSensor*, an Android application acting as a virtual sensor for measuring mobile app usage. AppSensor collects information on app installations, updates, and removals, as well as usage sessions of apps.

Wagner, Rice, and Beresford [156] developed an Android application named *Device Analyzer*, which captures a time-series log of more than 200 different events, device settings, telephony, or CPU and memory information for running processes. Over the course of several years, they collected a dataset containing over 100 billion records, which they made available for researchers.

Platforms and frameworks related to this work are typically designed to be exhaustive in gathering data from all available sources. Still, they do not provide means for collecting information on notification handling, nor do they allow for capturing events within a single application. The framework that is presented in this work explicitly focuses on capturing these data points.

2.3.4 Ethics

Conducting research with publicly available software, mobile applications, raises technical and non-technical issues. While technical ones were discussed in Sect. 2.3.1, this section discusses non-technical issues, with respect to ethics, privacy, and regulation.

¹⁹Funf Journal: <http://funf.org/journal.html> (last accessed: 2016-02-03).

Henderson and Abdesslem [69] highlight issues that arise when conducting world-wide ('planet-scale') experiments. They point out differences in data protection legislation between countries that need to be considered when designing experiments that are conducted across countries. In order to allow for such studies, they propose a regulation-aware architecture, which handles access to data sources (sensors) based on country codes, as provided by a mobile phone.

The American Psychological Association [4] requires informed consent of the individuals taking part in a study. However, McMillan et al. [107] argue that users of mobile applications, although accepting the terms and conditions for using the product — which explain that it is part of an academic trial — may not be aware of being part of a study. This is identified as a drawback of such studies, as the procedure in a traditional face-to-face handover may be repeated or re-worded as necessary, whereas the user's understanding in a remote setting is assumed on the basis of a checked checkbox. Good et al. [62] also found that users often do not understand the content of an end-user license agreement (EULA). If it is explained to them, in terms of a short summary, many users would uninstall an application that they had otherwise used. Pielot, Henze, and Boll [126] found that subtle differences in presentation style of consent dialogs can affect the opt-in rate. Cramer, Rost, and Bentley [36] note that it is the researchers' responsibility to ensure that all participants of a study, whether in a laboratory setting, or in the large, are fully informed, even if this results in fewer users of the application or service.

In order to provide users with more control over what is being logged, Morrison et al. [114] presented control switches for data logging to users of their application. They found that most users, who made use of these controls, turned off logging completely. Morrison et al. [115] further propose a compromise to protect users whose consent could not be assured to be 'informed': limited interaction in form of non-compulsory survey questions, and examination of log data in an aggregated form. Building on the results of [114], Morrison, McMillan, and Chalmers [113] suggest to interrupt use of an application with a visual representation of collected data, opposed to the common practice of providing a description of logged data points at first launch.

The American Psychological Association [4] further calls for a debriefing as a means to obtain appropriate information about the nature, results, and conclusions of the research. However, Kraut et al. [89] note that debriefing in online research may be difficult, as participants may leave the setting (a website) before receiving debriefing. This also applies for mobile applications [28, 115], where a user may simply uninstall the application, or the trial itself has no defined end date.

Finally, McMillan, Morrison, and Chalmers [106] propose a set of guidelines for large scale mHCI studies. They propose to give users the opportunity to review and purge the data they have generated as an alternative to the traditional debriefing.

In the frame of this dissertation, data from human participants was gathered in laboratory and field studies. Protection of this data according to legally effective privacy and data protection guidelines was mandatory. It was guaranteed that no sensitive ethical or legal issues were touched in the studies. The participants were informed about the recorded data and the purpose of the studies prior to every experiment and were able to abort at any time without giving reasons. This was done in the form of

paper-based consent forms in laboratory studies and visual pop-ups that appeared when launching the application for the first time in field studies. Where possible, the gathered data was anonymous, so that conclusions on the identity of participants were impossible. Otherwise, data was collected in a pseudonymized way and the links from pseudonyms to participants were disconnected after the study.

2.4 Related Work

The research questions addressed in this dissertation were outlined in Sect. 1.2. This section provides an overview of work related to these questions, which can be found in four areas: first, a literature overview on general use of mobile phones and smartphones is given in Sect. 2.4.1, followed by a discussion on studies on warnings and permission requests in Sect. 2.4.2. Systems allowing for adaptation of settings and related studies are presented in Sect. 2.4.3, finalized with an overview on work on notifications and interruptions in stationary and mobile contexts in Sect. 2.4.4.

2.4.1 *General Usage of Mobile Phones and Smartphones*

Harmon and Mazmanian [67] analyzed the cultural discourse about smartphones on the basis of advertisements and news articles. They differentiated stories presented in the media in one calling for increased technological integration, and, on the other hand, one urging individuals to ‘dis-integrate’ the smartphone from everyday life. While the former promised a transformation into ‘multi-task masters’, the latter presented the prospect of becoming ‘authentic humans’, as opposed to ‘distracted addicts’. The authors reported that “people are struggling to locate themselves in the overly simplistic stories that encounter about smartphones” and thus called on researchers to resist the simplification of findings, a clear-cut distinction between good and bad.

Adoption of Smartphones

Rahmati et al. [134] explored the influence of socioeconomic differences on smartphone use. They equipped 34 individuals with an Apple iPhone, which was instrumented with a logging software, for a yearlong study. The authors report that participants with a low socioeconomic status spent more money on applications, installed a larger number of applications, and spent more time using the device compared to others.

Rahmati and Zhong [133] studied the adoption of smartphones with 14 teenagers with little or no prior experience with smartphones over a time period of four months. Smartphones (*HTC Wizard*) that were provided to participants of this study were instrumented with logging software that traced battery level, charging status, available WLAN access points, and display status. Due to privacy concerns with underage

participants, the authors refrained from logging actual application usage and focused on reports in focus groups instead. The authors reported that participants' initial excitement about the device led to increased usage, which stabilized after roughly six weeks from the beginning of the study. Furthermore, they found that boredom affected the usage of the device: applications that were popular at the beginning of the study saw a decline in usage towards the end of the study.

Kujala and Miron-Shatz [90] examined emotions and experience episodes during real-life mobile phone use over a period of five months. The authors investigated how emotions and experiences related to usability, user experience (UX), and behavioral intentions. Twenty-two participants evaluated device use after five days after receiving a new phone, after two and a half months, and finally after five months of use. Results suggested that positive emotions were mostly related to good UX, whereas negative ones to low usability. Interestingly, users overestimated their positive emotions in early stages of use, focusing on UX; the importance of usability increased over time.

Personality Traits and Smartphone Use

Phillips, Butt, and Blaszczyński [123] investigated possible associations of personality traits and interest in mobile game applications, as these were previously reported for gambling on the internet. In a study with 112 participants, they found that people who scored low on agreeableness were more likely to use their mobile phone to play games. Chittaranjan, Blom, and Gatica-Perez [31] derived behavioral characteristics of 83 users who participated in the *Lausanne data collection campaign* (cf. [87]). They investigated the relationship between these characteristics and self-reported personality traits. Results of their study showed meaningful relationships between traits and application usage, call and SMS logs. Based on these results, they developed an automated system for classifying users based on personality traits. In a study with 312 participants, Lane and Manner [91] examined the effect of personality traits on smartphone ownership and use. Their results suggest that while personality was a fairly weak predictor of smartphone ownership and use overall, extraverted individuals were more likely to own a smartphone. Extraverts also reported a high importance of texting functionalities, while individuals with a high score on the agreeableness scale favored making phone calls.

Mobile Information Needs

Church and Smyth [32, 33] conducted a four-week diary study of mobile information needs. They found that mobile needs significantly differ from general Web needs and motivate this finding with the *on-the-move* nature of mobile users. They further highlight the importance of context, in particular location and time, on information needs, and conclude that context-sensitive systems may offer an improved mobile search experience to users. In an in-situ study with 18 Android users over two weeks, Carrascal and Church [27] investigated mobile app and mobile search interactions. They found that users who were engaged in search activity interacted more with other mobile applications and for longer duration within those search sessions. Those

sessions were also characterized by a different set of used applications compared to non search sessions, i.e., applications related to communication, browsing, retail and entertainment were more intensively used.

Brown, McGregor, and Laurier [25] investigated mobile application use by means of a video-based approach. Fifteen participants were equipped with wearable cameras, and a software captured the screen of their smartphone during a “typical day”. They reported that device use was threaded through other activities, passing a museum triggered a search for details on the device. In [105], McMillan, McGregor, and Brown extended the method of data collection to a combination of encapsulated screen recording, ambient audio recording, data logging, wearable cameras, and distributed remote uploads. They argued that, while the analysis of video and audio data can take considerable time, this method allows for an understanding of the context of use and of the use itself.

Do, Blom, and Gatica-Perez [45] presented a study on contextualized smartphone usage with 77 participants over a period of nine months. Similar to [31], they conducted their study on data from the Lausanne data collection campaign. Their findings suggest that application usage depends on the contextual factors place and social context, users may have a preference for the camera application when on holidays. Based on another excerpt of the Lausanne data set, Do and Gatica-Perez [46] proposed a probabilistic framework for mining usage patterns of mobile applications. Based on these patterns, the framework could also be used for user retrieval.

Habits

Oulasvirta et al. [121] conducted a series of studies investigating the habit-forming nature of smartphones. In particular, they were interested in the characteristics and the role of habits in HCI. They identified a *checking habit*: “brief, repetitive inspection of dynamic content quickly accessible on the device” that may increase smartphone usage overall.

King et al. [86] studied *nomophobia* (no-mobile-phone-phobia), the discomfort or anxiety caused by the non-availability of a virtual communication device (a mobile phone). Results of a case study suggested that dependency on virtual communication environments may be an indicator for a root cause disorder, a social phobia disorder.

On the other hand, Przybylski and Weinstein [129] investigated how the presence of mobile phones in social settings influences face-to-face interactions. The authors reported results of two studies, and concluded that the mere presence of such devices can interfere with human relationships.

Dey et al. [43] presented a field-study with 28 participants over a period of four weeks that investigated whether smartphones may serve as a proxy for the user’s context and availability. Similar to previous research on mobile phones (not smartphones), they found that the device, if switched on, is within arm’s reach for only about half of the time. However, the device was found to be in the same room as the user for about 90% of the time.

Negative Health Effects

Lee et al. [96] examined how a smartphone user's character affects perceived stress levels. In their study with 325 participants they found that compulsive smartphone usage and technostress, i.e., stress experienced by end users of information and communication technologies (ICTs), were positively correlated to psychological traits.

Thomée, Härenstam, and Hagberg [150] studied possible negative health effects of mobile phone exposure. From their questionnaire-based study with 4156 participants they found that high frequency of mobile phone use was a risk factor for mental health, it was associated with sleep disturbances and symptoms of depression. Lee et al. [95] investigated usage patterns that were related to smartphone overuse. In a study with 95 college students they compared risk to non-risk groups and found different diurnal usage patterns and longer use per day for the risk group. Further, they report that participants received more than 400 notifications per day on average, mostly from messaging applications. Individuals of the risk group were more susceptible to such notifications.

Metrics for Smartphone and Mobile Application Use

Based on data collected with the *AppSensor* framework (cf. Chap. 2.3.3), Böhmer et al. [20] reported results of a study with 4100 users in the time frame of August 2010 to January 2011. They found that the average session length with an application was less than a minute, whereas the total time spent with the device was almost an hour a day. A comparable amount of time spent with smartphones per day was also reported in [118]. Remarkably, this number increased to 3 h and 40 min in 2015 [84]. Böhmer et al. further presented contextual findings, news applications were found to be popular in the morning, whereas games were popular at night; applications related to communication showed a more steady use throughout the day.

Falaki et al. [49] reported on diversity in smartphone usage. Usage data collected from 255 participants showed differences in the number of times individuals interacted with their device, from 10 to 200 times a day. Similarly, the mean interaction length varied between 10 and 250 seconds for different users. However, the authors reported that user demographics alone cannot reliably predict how a user will use the phone.

2.4.2 Warnings and Permission Requests

The communication-human information processing (C-HIP) model introduced by Wogalter, DeJoy, and Laughery in 1999 (cf. Sect. 2.1.2) did not explicitly target warnings in modern technological devices, such as personal computers or smartphones. Therefore, Laughery and Wogalter [94] presented approaches for applying technology to warnings, and called for accurate, appropriate warning information in a timely fashion with properties better than those of traditional static warnings.

This section provides an overview of work in this field, partially based on the C-HIP model or on the human-in-the-loop framework.

Stationary Context

Egelman, Cranor, and Hong [47] presented a study on the effectiveness of web browser warnings. In a laboratory study with 60 participants, they examined possible causes of failure to comprehend such warnings on the basis of the C-HIP model. Comparing active and passive browser warnings, they found that participants were highly susceptible to phishing attacks with passive warnings, with habituation given as one reason for ignoring warning indicators. The authors argued that warning indicators need to be designed such that they interrupt the user's primary task in order to be effective.

Bravo-Lillo et al. [23] investigated the effect of such attractors that inhibited the user from processing the primary task. In a series of studies using the Amazon Mechanical Turk (MTurk)²⁰ system, they found that participants exposed to such warnings were less likely to bypass them, but instead more likely to make an informed decision. The authors acknowledged the cost in terms of delayed workflow, but added that this may decrease over time due to habituation. Similar results were reported by Anderson et al. [5], who used functional magnetic resonance imaging to show that visually similar security warnings can lead to users ignoring them. In a follow-up study, Bravo-Lillo et al. [24] replicated the previous study ([23]) and added low-habituation conditions in order to measure how habituation affects attention. Their results indicated that not all attractors were resilient to habituation, whereas others imposed great usability burdens. Finally, they called for an integration of habituation tests when evaluating real-world security dialogs.

Bravo-Lillo et al. [22] further conducted a study in order to gain insights into how novice and advanced users with respect to computer security make sense of warnings. They interviewed 30 participants and reported that the two groups differ in how they evaluate security warnings, novices revealed several misconceptions about security. For example, novices reported that interaction with banking websites, although carrying a warning message, "ought to be safe simply because banks have good security". Further, they pointed out a trade-off between the amount of information presented in warnings, and the added likelihood that new information facilitates an informed decision, as participants often did not thoroughly read warning messages.

Kim and Wogalter [85] investigated habituation, dishabituation, and recovery effects in visual warnings. In a laboratory study with 71 participants, a repeated stimuli exposure resulted in a decrease in perceived alertness, suggesting habituation. Changing the format of the warning caused dishabituation, whereas a return to the initial warning format provided evidence for a recovery of habituation. The authors conclude that, besides the positive effects of standardization for warning signs, flexibility in the design of warnings could be beneficial.

Harbach et al. [65] studied the content of security warnings with respect to linguistic properties in a series of online studies and phone interviews. They found

²⁰ Amazon Mechanical Turk: <https://www.mturk.com>.

that warning message texts were often too complicated, although understandable if participants spent enough time. Linguistic properties accounted for half of the variance in ratings of warning texts, therefore the authors concluded that such messages should be simple, preferably using non-technical terms, and without complicated grammatical constructs.

Mobile Context

Felt, Greenwood, and Wagner [51] reviewed browser extensions and mobile applications for the Android OS based on their requested permissions. They argued that systems requiring up-front permission declarations were advantageous compared to traditional permission systems in that they, on the basis of the reviewed applications, do not request the maximum set of permissions. Still, they found that “users were frequently presented with requests for dangerous permissions”. Based on the C-HIP model, Felt et al. [50] evaluated whether Android users pay attention to, understand, and act on permission information during installation of mobile applications. They conducted two studies, an internet survey with 308 Android users, and a laboratory study with 25 Android users. They reported that participants of their studies paid low attention to permission information and displayed low comprehension rates, only 17% of participants of the laboratory study paid attention to permissions during installation, and merely 3% of respondents to the Internet survey did comprehend permission information. The authors concluded that Android permissions fail to inform the majority of users, although they were not found to be wholly ineffective.

In order to tackle the issue of failing permission information, Harbach et al. [66] proposed to leverage personal information stored on the user’s device to communicate risks. For this purpose, they designed various permission visualizations that included personal information, a photo of the photo library was displayed when an app requested permission for accessing the storage of the device. In a laboratory study with 36 participants, and an online screenshot-study (using MTurk) with 332 participants they found that their custom permission visualizations led to more privacy-conscious choices when deciding which apps to install. Further, the authors reported that participants of the study paid more attention with modified permission dialogs.

Kraus, Wechsung, and Möller [88] took a similar approach in that they augmented the default permission visualization for Android applications with additional information: they displayed statistical information that compared the number of requested permissions to the one of similar apps. They found that a graphical representation of this added information led users to more privacy-savvy decisions when installing mobile applications.

Tan et al. [146] investigated the adoption and effect of developer-specified permission requests for privacy-related data (address book or location) on the iOS platform. As outlined in Sect. 2.2.1, developers of iOS applications can specify the purpose for requesting permission to access to a protected resource. In an analysis of 4400 apps they found that only 19% of these provided a custom explanation. The authors conducted an online screenshot study with 772 smartphone users, and found

that users granted permission more often, if a custom purpose was given, regardless of the actual content.

Last but not least, Urban Airship [154], providing a commercial mobile engagement platform, analyzed 2946 iOS applications, which used their service for delivering notifications. They reported that high performing apps across all industries have opt-in rates for notifications above 50% with applications from the travel, business and charities categories topping out at above 70%.

2.4.3 *Adaptation of Settings on Smartphones*

Zhou et al. [173] presented *TISSA*, a system that implements a privacy mode on the Android platform. Motivated by reports of leaked personal information from Android applications, the authors added a layer of protection to the existing Android permissions, which may provide default access to trusted applications, anonymized data (contact information), bogus information (providing a random device identifier), or no data at all. Adaptation of the level of access is done via a settings panel. Zhou et al. evaluated their approach with respect to effectiveness and performance, but did not conduct a user study. Ben-Asher et al. [14] presented results of a survey with 465 participants on users' security needs, awareness, and concerns in the context of smartphones. The authors argued for a graded security model that would allow for a variety of authentication methods (PIN entry, or fingerprint scans) for different content on the device, as participants of their study reported a varying level of sensitivity for different content. Related frameworks, which modify the Android operating system to offer more fine-grained controls over permissions, have been presented by Beresford et al. [17] and Nauman, Khan, and Zhang [117]. While *MockDroid* [17] reported accessed resources as empty or unavailable to third-party applications, *Apex* [117] allowed users to conditionally grant access to resources, based on time of the day.

Jeon et al. [79] presented a different approach, relying on modifications of third-party applications instead of modifying the system level. Their framework added more fine grained permissions to Android applications, access to the address book may be limited to certain fields. The authors showed that their system may be employed for existing applications and suggested that it “allows app users to more directly control the security of the apps they run”. However, they did not validate this in a user study.

Backes et al. [10] presented another system, *AppGuard*, which is a standalone Android application that modifies applications installed on the device and offers users a settings panel with revocable permissions. Apart from a performance evaluation, they report that AppGuard has been installed on more than 5,00,000 devices.

Previously reported systems either modified the operating system itself [17, 117, 173], or made changes to third-party applications [10, 79]. While the former requires the user to install a custom operating system, the latter prevents modified applications to receive updates through the mobile application store. Liu et al. [98] proposed

a crowd-sourced approach to recommend app-based privacy settings. *PriWe* is an Android application that scans installed third-party applications and their requested permissions. Users of the application may indicate whether they approve or dislike access to specific resources of this application. This user-generated data is then used to provide recommendations to other users of *PriWe*. The authors conducted two studies (one using MTurk with 382 participants, and one with 78 participants in the field) and concluded that their system was capable of providing proper recommendations for privacy settings.

2.4.4 Notifications and Interruptions

Interruptions have been investigated long before the emergence of personal computers, for example by Zeigarnik [170] in 1927. Yet, modern computing systems extend the diversity and frequency of interruptions.

McFarlane [102] defines *human interruption* as “the process of coordinating abrupt change in people’s activities”. In the context of Human-Computer Interaction (HCI), he argued that “interrupting the user is not a *bad* thing. Instead, interruption is just another kind of HCI”. Tolmie et al. [151] added that interruption and the disruption of action were an inevitable fact of life, borne of a multitude of contingencies.

Notifications and Interruptions in a Stationary Context

In 1989, Gillie and Broadbent [61] conducted a series of experiments in the context of a computer-based adventure game. Participants’ main task, playing the game, was interrupted with another task (mental arithmetics). The authors showed that current memory load and length of the interruption were not determining factors for perceived disruption. Rather the complexity (amount of processing required) and similarity to the main task were found to be main influencing factors for whether an interruption was disruptive or not. Others have replicated and extended this result, Czerwinski, Chrisman, and Schumacher [40] and Bailey, Konstan, and Carlis [11].

In a study with 36 participants, McFarlane [103] investigated approaches to decide when to interrupt people based on the *Method of Coordination* [102], which differentiates the type of coordinating user-interruptions into (1) immediate (requiring immediate response), (2) negotiated (the user chooses when to attend), (3) mediated (an intelligent agent determines the best time to interrupt), and (4) scheduled (interruptions occur at predefined time intervals). McFarlane found that none of these four methods was the single best approach to interrupt users in tasks across all performance measures. It is noteworthy that users completed the interrupting task promptly if forced to handle the interruption immediately. However, overall performance decreased with this approach.

Cutrell, Czerwinski, and Horvitz [39] further investigated the cost of interruptions in the context of notifications from instant messaging (IM) applications. From a study with 16 participants they reported a generally disruptive effect of IM notifications

on search tasks. The authors also provided evidence that interruptions differ with respect to disruption based on the current stage of a task the user is involved in, as suggested by Miyata and Norman [111].

Arroyo, Selker, and Stouffs [7] explored the use of different sensory cues as a means to interrupt users: visual (light), auditory (sound), haptic (heat and vibration), and olfactory (smell). In their experiment with twelve participants they did not find significant differences in disruption for different modalities. However, they argued that individual differences, users' backgrounds, determine the effectiveness of these modalities. In a follow-up study with 23 participants in the context of playing a game on a desktop computer, Arroyo and Selker [6] compared visual (light) to haptic (heat) cues to interrupt users in their main task (playing the game). For this study the authors reported a significant difference for modalities: the thermal modality was found to entail a larger decrease in performance on the main task and a higher level of disruption compared to the visual modality.

Adamczyk and Bailey [2] presented a study on the effects of interruptions at different moments within task execution. Sixteen participants in this study executed a series of computing tasks and were interrupted at different moments. The authors reported that the presumed best time to interrupt users, i.e., after completing a sub-task within the main task, produced less annoyance, frustration, and time pressure. Further, this interruption strategy required less mental effort. The authors argued that attention managers, trying to identify opportune moments for interruption, need to be equipped with task models or learn them over time in order to be effective.

Horvitz, Apacible, and Subramani [76] reviewed experiments on bounded deferral [77], a method aimed at reducing disruption of incoming messages and alerts by deferring them until the user transitions to a non-busy state, or a prior set maximum time is exceeded. The authors suggested that trading off delays in information delivery against a reduction in disruption during busy situations, office hours, is a promising approach for "quieting the noisy chatter of incoming alerts, while allowing people to stay aware of important information".

Avrahami, Fussell, and Hudson [9] investigated context, message characteristics, and demographic characteristics as factors influencing the time to react to interpersonal communication in IM applications. Nineteen participants used an IM chat application that was instrumented with a logging software over the course of four weeks. Results showed that not only demographic characteristics influenced reaction time to a message, students responded faster than participants working in a start-up, but also message characteristics, questions were responded to faster than non-questions. Similar to previously reported studies, the authors found that users who were engaged in a task that involves frequent switching between applications responded faster to incoming communication.

Vastenburg, Keyson, and Ridder [155] studied acceptability of alerting and informational messages in the context of a Smart Home. In a study with ten participants, a computer was set up in participants' homes to trigger a set of simulated alarms at a predefined schedule. It is noteworthy that these alarms, "coffee is ready", were not related to real occurring events. The authors reported that "message urgency was

found to be a better predictor of acceptability than the degree of user engagement during ongoing activities”.

Iqbal and Horvitz [78] reported results of a study on desktop e-mail notifications in the workplace. They recorded desktop interactions of 20 participants for two weeks, with disabled e-mail notifications in the second week. In line with previous research, the authors reported the disruptive nature of notifications. However, they also highlighted the added value of raising awareness. Interestingly, turning off notifications caused self interruption — to explicitly monitor e-mail arrival — for some users.

Notifications and Interruptions in a Mobile Context

Previously presented work focused on interruptions and notifications in a stationary context, typically associated with a stable context of use (work or home). This setting provides natural ways to opt-out, for example by not starting the messenger or by walking away from the personal computer (PC) [127]. Conversely, mobile devices such as smartphones are typically in close proximity to the user [43].

Kern and Schiele [83] proposed to utilize body-worn sensors to estimate interruptibility of a user of a wearable computer. Instead of modeling the complete context of the user, the authors’ model included the user’s current activity, social engagement, and social expectation of group behavior. Sensors for determining these factors were an accelerometer for detecting activity, a microphone for the reasoning about the social context, and a wireless local area network (WLAN) access point to determine location. The authors reported that this setup was sufficiently precise in detecting personal interruptibility, i.e., in 94.6% of the time in a pilot study with the authors as participants.

Previous work on strategies for notification delivery in a stationary context suggested that task switches may serve as opportune moments for notifying the user [2, 9, 76]. Ho and Intille [74] used this approach in a mobile context to compare receptivity to notifications delivered at activity transitions to those delivered at random times. The authors pointed out that changes in physical activity may be correlated with self-initiated task interruptions, and may thus serve as opportune moments for notification delivery. Twenty-five participants of their study were equipped with two accelerometers to record activity switches and a PDA to receive notifications for one day. Results of their study showed that messages delivered at activity transitions were better received than those at random times. However, some participants also reported that “there were periods during the day when they had nothing to do and were surfing the Internet”, describing these moments as ones they were especially receptive to any interruption. Fischer, Greenhalgh, and Benford [55] used an Android application to monitor user interaction with the device. In a study with 20 participants, they compared Short Message Service (SMS)-style notifications sent at (pseudo-)random times to those sent after the user had finished an episode of mobile phone activity. Results of their two-week study showed that participants attended to and dealt with the notification significantly faster upon finishing an episode of mobile phone activity, finishing a voice call.

Fischer et al. [56] investigated the effects of content and time of delivery on receptivity to SMS interruptions. The authors set up a system that sent out SMS

messages to eleven participants of their study with content from various categories over a period of two weeks, with weekends omitted. Distinguishing between good and bad content and timing, they reported that “content trumps the time of delivery” in the context of SMS. The authors further suggested that most of their participants were well-trained in ignoring the interruption caused by incoming messages, thus not leading to a significant effect of time of delivery. However, Fischer [54] also hypothesized that “an automated prediction of opportune moments for interruption is a highly uncertain endeavour that may never be 100% accurate”.

Rosenthal, Dey, and Veloso [135] presented an Android application that learned users’ preferences for receiving audible notifications (calls, SMS, and calendar alerts) in order to automatically turn on or off the phone volume in a later step. The authors reported that participant’s volume preferences changed throughout the study with ninety-four participants, which represents a challenge for systems learning users’ routines and habits.

Leiva et al. [97] reported on an observational study on the costs of mobile application interruptions by incoming phone calls. Data was collected with the AppSensor framework (cf. Sect. 2.3.3) from 3600 users. Results suggested that application interruptions, although rarely happening, were costly in terms of delayed task completion. The authors further reported that application-switching did not happen as often as it is presumed, a finding that needs to be considered when designing systems that defer notifications to opportune moments ([55, 74]).

Pielot, Church, and Oliveira [125] reported results of a one-week in-situ study on mobile phone notifications involving 15 participants. Participants’ smartphones were instrumented with an application that logged incoming notifications and participants provided daily feedback regarding perceptions and experiences with notifications for the previous day. Results showed that participants received on average 63.5 notifications per day, mostly from messaging and e-mail applications. Still, these notifications were typically attended to within minutes, and participants reported social pressure as a main reason. The authors further highlighted the two-edged character of notifications on mobile phones: an increasing number of notifications was associated with an increase in negative emotions, but at the same time reinforced the feeling of connectedness among participants.

Pielot et al. [127] further investigated users’ interaction with notifications of messaging applications. Some popular messaging applications, WhatsApp,²¹ display the time and date a user was last seen. Pielot et al. argued that this information raised social pressure, a friend might expect a fast response if one has been using the application a few minutes earlier, but was a bad predictor of how fast a user might respond. They proposed a machine-learning approach for predicting and informing whether a person will see a notification within the next few minutes or not. It was evaluated in a two-week study with 24 Android users and found to be correctly predicting a user’s attentiveness in 70.6% of cases. Moreover, the authors reported that users found this method less privacy invading and at the same time reducing social pressure compared to the current mechanism.

²¹ WhatsApp: <https://whatsapp.com>.

Mashhadi, Mathur, and Kawsar [101] explored how users deal with notifications on their smartphones in a field study with ten participants for fifteen days. Results suggested that participants were more likely to attend to a notification if they were actively engaged with the device. While they did not find a significant effect of modality, i.e., sound, light, or vibration, participants reported to associate sound and vibration with important notifications. Thus, the authors call for more fine-grained control over notification management, “allowing them [users] to specify what is important to them”. They further looked into the impact of time of the day a notification was delivered on time to attend to the notification. The authors argued that while different times of the day led to a varying number of attended notifications, time alone may not be a good predictor for how long it will take a user to attend to them.

Shirazi et al. [144] conducted a large-scale study by deploying a system, *Desktop Notification*, that collected notifications from users’ Android smartphones and forwarded these to a browser plugin of a stationary desktop computer, which additionally allowed users to provide feedback on a notification. The authors analyzed close to 200 million notification from more than 40000 users. The authors reported that, if a user attended to a notification, it happened within the first 30 seconds after arrival. They further reported differences in time it took users to attend to a notification dependent on the apps’ categories, with the system category (Google Play Store app updates) yielding the lowest reaction time, followed by messengers. However, notifications from the system category were found to be rated with least importance among all categories. Still, reaction time was found to be negatively correlated with perceived importance overall, i.e., more important notifications were attended to faster. The authors concluded that notifications were considered important if they were about events or provided information about the user’s context or contacts, including notifications from messenger applications.

Previous work on detecting opportune moments for delivering notifications took attention as a scarce resource as a starting point. Pielot, Baltrunas, and Oliver [124] argued that attention is not always scarce, for example when people are bored. They proposed to detect boredom to trigger the delivery of notifications. Results of a study with 16 participants for two weeks showed that participants were more likely to attend to a notification (in this case a recommendation to read an article on a website) if their model inferred the user to be bored.

Mehrotra et al. [110] used the ESM to investigate the acceptance to notifications on smartphones. Thirty-five users voluntarily installed an Android application from the Google Play Store that logged incoming notifications and randomly prompted users to provide a rating for a notification, resulting in a dataset of more than 70,000 notifications after three weeks. The authors reported that acceptance of a notification depended on the content and originator of the notification. Physical activity was found to be a predictor of time needed to attend to a notification. Based on their results they built a machine learning model that was capable of predicting whether or not a user attended to a notification within ten minutes after arrival with an accuracy of 70%. Mehrotra et al. further suggested to prompt users at opportune moments

for providing feedback in order to better align ESM with users' lifestyle and gather meaningful responses to questions [109].

Turner, Allen, and Whitaker [152] argued that the binary view on attendance to notifications, i.e., whether the user did or did not open the notifying application, may miss partial interactions. The authors decomposed the decision process to include single steps, turning on the screen as a reaction to an audible or tactile notification. From a study with 94 participants over six month they found that partial responses occurred in 14.6% of interruptions when the device was not in use. Thus, they concluded that considering partial responses increases the number of cases where some degree of interruptibility is shown.

2.5 Summary

This chapter introduced the foundations of this thesis, outlined the research approach, and presented works related to the four fields of this thesis.

First, the notion of mobile Human-Computer Interaction was introduced, and concepts of human attention as a scarce resource, as well as human information processing were discussed in Sect. 2.1. The scarcity of human attention justifies the necessity for research in this domain, as notifications on smartphones compete for users' attention, which may eventually result in stress due to information overload.

Access to notifications is handled in different ways, depending on the operating system. Differences worthy of note were discussed in Sect. 2.2, as well as the path a notification takes from a sender to its recipient.

Next, the research approach taken in this dissertation was explained in Sect. 2.3. Application stores as an instrument allowing for deployment-based research in a potentially large scale were presented, further establishing the possibility for research in the wild. Based on that, studies and frameworks utilizing this approach were presented.

Section 2.4 presented work related to the four fields of general usage of mobile phones, handling of permission requests, adaption of settings, and dealing with notifications and interruptions. Links to this work are as follows:

- *General Usage of Mobile Phones and Smartphones*: The literature suggests a diversity of smartphone ownership and use [49], advising against simplification of findings [67]. Smartphones were found to be habit-forming [121], with compulsive usage leading to negative health effects [96, 150]. Based on that, Chap. 4 reports on a global analysis of user behavior with mobile applications and notifications using an experience sampling approach as a basis for upcoming studies. It contributes an up-to-date picture of how users currently deal with notifications on their smartphones and means they use in order to limit notifications' intrusiveness.
- *Warnings and Permission Requests*: Research in this area has shown that warning messages and requests often fail, either due to habituation [23, 47, 85], or because the given message is hard to understand [65]. Consequently, researchers presented

approaches tackling these issues, by adding additional information [88, 146], or personalizing these messages [66]. Presented work in this chapter related to the field of privacy and security to a great extent, but did not target the field of notifications on smartphones. As such, Chap. 5 contributes to the literature by considering permission requests in the area of notifications on smartphones.

- *Adaptation of Settings*: Research on adaptation of settings on smartphones is related to privacy controls to a great extent. Approaches were presented that left the responsibility of adjusting settings to the system by modifying parts of it, the system level [17, 117, 173] and user-installed applications [10, 79], or recommended users settings based on crowdsourcing [98]. In the frame of this dissertation, settings with respect to notifications are in focus. In this area, more fine-grained controls over notification settings were demanded [101]. Although technically feasible, the approach of modifying system components does not seem to be suitable for the research approach taken in this work, i.e., utilizing mobile application stores as an instrument. Thus, approaches of communicating means to adapt settings are investigated in Chap. 5.
- *Notifications and Interruptions*: Interruptions are seen as an inevitable part of life [151] and see a long tradition of research on various characteristics. Whereas the number of studies on notifications and interruptions in a stationary context saw a decline over the past years, research in the mobile context gained momentum with the advent of smartphones and mobile application stores. Researchers mostly agree that notifications may be disturbing [54, 101, 104, 125, 151], but also highlight benefits, such as increased awareness [76, 78], or the chance to bridge periods of boredom [124]. As such, detecting opportune moments for interruption were in the focus of many studies [55, 111, 124]. Chapter 6 presents studies on receptivity to notifications, extending the literature by investigating various characteristics of notifications on smartphones.

The following chapters present work that is based on the foundation outlined in this chapter.

User Acceptance of Mobile Notifications

Westermann, T.

2017, XVI, 144 p. 39 illus., 20 illus. in color., Hardcover

ISBN: 978-981-10-3850-1