Universität Stuttgart

Empirical Assessment and Improvement of Ubiquitous Notifications

Von der Fakultät für Informatik, Elektrotechnik und Informationstechnik
und dem Stuttgart Research Centre for Simulation Technology
der Universität Stuttgart zur Erlangung der Würde eines Doktors der
Naturwissenschaften (Dr. rer. nat.) genehmigte Abhandlung

Vorgelegt von
Dominik Christian Weber
aus Backnang

Hauptberichter:	Prof. Dr. Niels Henze
Mitberichter:	Prof. Dr. Christian Becker
Mitberichter:	Prof. Dr. Stefan Schneegass

Tag der mündlichen Prüfung: 27.10.2023

Institut für Visualisierung und Interaktive Systeme
der Universität Stuttgart

2023

Zusammenfassung

Intelligente Geräte sind allgegenwärtig geworden. Geräte wie Smartphones, Smartwatches, Tablets, Laptops und Smart-TVs begleiten uns den ganzen Tag. Dank Fortschritten bei der Rechenleistung und Drahtlostechnologien sind diese Geräte immer eingeschaltet und immer verbunden. Während einige Geräte nur situativ genutzt werden, sind andere Geräte wie Smartphones immer beim Nutzer. Dies hat die Art und Weise, wie mit diesen Geräten interagiert wird, grundlegend verändert. Anstatt manuell nach Neuigkeiten und neuen Nachrichten zu suchen, können uns diese Geräte rund um die Uhr proaktiv durch Benachrichtigungen über neue Ereignisse informieren. Von neuen Nachrichten über Erinnerungen bis hin zu Systemaktualisierungen - Benachrichtigungen sind grundsätzlich persönlich und decken ein breites Spektrum an Themen ab. Während Benachrichtigungen von den Nutzern geschätzt werden und ihnen das Gefühl geben, verbunden zu sein, können sie auch zu Unterbrechungen und Ablenkungen führen. Da immer mehr Dienste auf immer mehr Geräten auf Benachrichtigungen zurückgreifen, werden die potenziellen negativen Auswirkungen von Benachrichtigungen verstärkt. So kann beispielsweise eine einzige E-Mail einen Benutzer auf mehreren Geräten mit verschiedenen Modalitäten benachrichtigen. Um negative Auswirkungen zu verringern, ist ein Verständnis der verschiedenen Kategorien von Benachrichti-

gungen, der verschiedenen Geräte und der Bedürfnisse der Benutzer erforderlich. Das Benachrichtigungsmanagement ist ein Balanceakt zwischen dem Stillen des Informationsbedürfnisses der Benutzer und der Wahrung ihrer Aufmerksamkeit.

Diese Arbeit beschäftigt sich mit der empirischen Bewertung und Verbesserung von allgegenwärtigen Benachrichtigungen. Es werden mehrere Nutzerstudien präsentiert, von Online-Umfragen, Laborstudien, In-situ-Studien bis hin zu großangelegten In-the-Wild-Studien. Zunächst wird auf die Bewertung und das Management mobiler Benachrichtigungen auf Smartphones eingegangen und anschließend auf die Herausforderungen der Durchführung von kontrollierten In-situ- und In-the-wild-Studien unter Wahrung der Privatsphäre der Nutzer. Es wird ein Benachrichtigungsdatensatz präsentiert, verschiedene Nutzertypen vorgeschlagen und neue Ansätze vorgestellt, mit denen die Nutzer über ihre Benachrichtigungen reflektieren und sie verwalten können. Anschließend wird der Anwendungsbereich auf andere Gerätetypen wie Smartwatches, Tablets und Laptops erweitert, um ein ganzheitliches Verständnis dafür zu schaffen, wie sich diese Geräte in Bezug auf die Erwartungen der Nutzer an den Empfang von Benachrichtigungen unterscheiden, indem Aktivitätsprotokollierungen auf mehreren Geräten mit Erfahrungsstichproben kombiniert werden. Schließlich wird der Anwendungsbereich erneut erweitert, um große und allgegenwärtige Displays einzubeziehen. Abschließend wird ein Open-Source-Protokollierungsframework für mobile Geräte vorgestellt, damit andere Entwickler und Forscher auf dieser Arbeit aufbauen können.

Der Beitrag dieser Arbeit besteht aus drei Teilen. Erstens werden in dieser Arbeit mehrere Ansätze zur Erforschung allgegenwärtiger Benachrichtigungen vorgestellt, von kontrollierten Laborstudien bis hin zu groß angelegten Studien in freier Wildbahn. Zweitens bietet die Arbeit Einblicke in die Benachrichtigungspräferenzen und -interaktionen der Nutzer auf verschiedenen Gerätetypen. Drittens wird ein technischer Beitrag geleistet, der ein Open-Source-Framework zur Protokollierung von Benachrichtigungen und einen Datensatz für Benachrichtigungen umfasst. Diese Beiträge bilden eine Grundlage für die zukünftige Forschung zu allgegenwärtigen Benachrichtigungen.

Abstract

Smart devices have become ubiquitous. Devices like smartphones, smartwatches, tablets, laptops, and smart TVs accompany us throughout the day. Advancements in computational efficiency and wireless technologies allow these devices to be always on and always connected. While some devices are used situationally, other devices like smartphones are always with the user. This inherently changed how we interact with these devices. Instead of manually looking for news and new messages, these devices can proactively inform us about new events through notifications around the clock. From new messages, reminders, to system updates, notifications are fundamentally personal and cover a wide range of topics. While notifications are valued by users and make them feel connected, they can also cause interruptions and distractions. With more and more services making use of notifications on more and more devices, potential adverse effects are amplified. For instance, a single email might alert a user on multiple devices using multiple modalities. To reduce adverse effects, an understanding of different categories of notifications, different devices, and user needs is required. Notification management is a balancing act between satisfying users' information needs and respecting their attention.

This thesis investigates the empirical assessment and improvements of ubiquitous notifications. We present multiple user studies, from online surveys, lab studies, in-situ studies to large-scale in-the-wild studies. We first focus on the

assessment and management of mobile notifications on smartphones, tackling the challenges of conducting in-situ controlled and in-the-wild user studies while preserving the users' privacy. We present a notification data set, propose user types, and introduce new approaches for users to reflect on and manage their notifications. We then expand the scope to include other device types such as smartwatches, tablets, and laptops to create a holistic understanding of how these devices differ regarding user expectations for receiving notifications by combining activity logging on multiple devices with experience sampling. Afterward, we expand the scope again to include large and pervasive displays. Finally, we present an open-source logging framework for mobile devices to enable other developers and researchers to build on top of this work.

The contribution of this thesis is threefold. First, this thesis introduces multiple approaches to conducting research on ubiquitous notifications, from controlled lab studies to large-scale in-the-wild studies. Second, the thesis provides insights into users' notification preferences and interactions on different types of devices. Third, a technical contribution, including an open-source notification logging framework and notification data set. These contributions are a foundation for future research on ubiquitous notifications.

Acknowledgements

This thesis would not have been possible without the support of some truly awesome people. First I want to thank my supervisor **Niels Henze**, who guided me from being a student assistant, through my diploma thesis, to this PhD thesis. Thank you for your trust and for giving me the freedom to follow my interests. A little less freedom might have sped things up but here we are. Second I want to thank the PhD committee, consisting of **Christian Becker**, **Stefan Schneegass**, and **Ralf Küsters** for the interesting discussions and smooth process. Special thanks to **Albrecht Schmidt** for creating a truly unique environment at the University of Stuttgart where everyone could feel empowered. Thank you **Alireza Sahami Shirazi** for setting me on this path.

Alexandra Voit, thank you for the close collaboration and support throughout these years and after. **Sven Mayer**, thank you for the motivation and pushes. **Rufat Rzayev**, thank you for sharing many hotel rooms even though some had bugs. **Valentin Schwind**, thank you for the nightmares about uncanny cats. **Huy Viet Le**, thank you for enduring me as an office mate. **Passant El.Agroudy** and **Sarah Faltaous**, thank you for the shared suffering and support. **Yomna Abdelrahman**, thank you for your giant heart and delicious food. Special thanks to **Jakob Karolus**, **Markus Funk**, **Pascal Knierim**, and **Thomas Kosch** for the dedicated *WhatsApp* group and "When do you submit?" pings after every single conference. It helped.

Shoutout to the *hciLab* group at the University of Stuttgart: **Bastian Pfleging**, **Céline Coutrix**, **Florian Alt**, **Francisco Kiss**, **Hyunyoung Kim**, **Katrin Wolf**, **Lars Lischke**, **Lewis Chuang**, **Mariam Hassib**, **Matthias Hoppe**, **Mauro Avila**, **Miriam Greis**, **Nitesh Goyal**, **Norman Pohl**, **Patrick Bader**, **Paweł W. Woźniak**, **Romina Poguntke**, **Thomas Kubitza**, **Tilman Dingler**, and **Tonja Machulla**. It was great having all of you around. Every one of you helped to create a truly special place. Extended greetings to the *VIS(US)* group and friends: **Dominik Herr**, **Edwin Püttmann**, **Kuno Kurzhals**, and **Tanja Blascheck**. At *SimTech*, thank you, **Andrea Barth**, for co-advising my milestone presentation. **Reiner Dietz**, thank you for granting me root access. Thanks to **Anja Mebus**, **Murielle Naud-Barthelmeß**, and **Eugenia Komnik** for your administrative support.

Thanks to all the students I worked with. Espcially **David Hägele**, **Frank Bastian**, **Gisela Kollotzek**, **Lucas van der Vekens**, **Marcus Hepting**, **Marvin Tiedtke**, **Philipp Kratzer**, and **Rodrigo Ventura Fierro** whose contributions ended up in this thesis. And **Jonas Auda** who ended up finishing his PhD before me. Great work!

If I missed anyone, I would like to sincerely apologize. I am writing these words several years after concluding the actual work of the thesis, so some things already started to slip from my mind. As a catch-all, I want to thank everyone I worked with at the University of Stuttgart and in research projects. Thanks to everyone who inspired me and provided feedback at conferences, doctoral colloquia, and workshops.

Finally, I want to thank my parents **Margot** and **Bernhard Weber** for their endless love and support throughout this and everything else in life.

Thank you.

Table of Contents

1

Introduction

In the 1991 article "The Computer for the 21st Century," Mark Weiser outlined the vision of ubiquitous computers [186]. He predicted that future computers would be integrated seamlessly into the world and "vanish into the background." According to Weiser, these ubiquitous computers would come in different sizes: *Tabs* ("inch-scale"), *pads* ("foot-scale"), and *boards* ("yard-scale"). Although Weiser mentioned several challenges in terms of software and hardware requirements, including wireless connectivity, he highlighted the benefits of ubiquitous computers:

> "Most important, ubiquitous computers will help overcome the problem of information overload. There is more information available at our fingertips during a walk in the woods than in any computer system, yet people find a walk among trees relaxing and computers frustrating. Machines that fit the human environment, instead of forcing humans to enter theirs, will make using a computer as refreshing as taking a walk in the woods." (Mark Weiser, 1991 [186])

Thirty years later, computers have indeed become ubiquitous. Mobile phones gained rapid adoption and evolved into always-connected smartphones that accompany users throughout the day. Smartwatches emerged as companion devices to smartphones and standalone wrist-worn computers. Tablet-computers complement desktop PCs and laptops. Televisions in the living room have become smart as well, allowing access to media on demand. Outside the home, public displays provide users with information. These smart devices are often connected using high-speed and highly-efficient wireless networks.

Not only are these devices providing users with information at their fingertips. Smart devices can provide users with information proactively. Using visual, tactile, and auditory cues, these devices can use notifications to gain the user's attention [66]. Notifications can be issued for all kinds of events, from communication-related, to breaking news, weather updates, to system alerts. However, prior work found that while users value such notifications, they can also cause interruptions and distractions [47, 67, 137]. This can lead to adverse effects, such as increased stress [193], inattention [78], and reduced task performance [30]. In an environment with ubiquitous computing, devices these adverse effects might become amplified. For instance, in an environment with a smartphone, smartwatch, tablet, and laptop, a single email might cause all devices to notify the user.

As ubiquitous computing environments expand, we need to research how to balance the users' information needs while respecting their attention.

1.1 Research Questions

This thesis investigates the empirical assessment and improvement of ubiquitous notifications. As this is a broad topic, we first need to define the scope of the thesis. Smartphones themselves have become ubiquitous with a high market penetration. Combined with the fact that smartphones are typically with the user throughout the day, we focus on mobile notifications in the first part of this thesis. We then expand the scope beyond mobile notifications to include different kinds of personal devices, such as smartwatches, tablet computers, and desktop PCs/ laptops. Finally, we expand the scope further to include large and pervasive displays, such as smart TVs and public displays.

The research questions (RQs) are listed in Table 1.1. The first research question (RQ1) is about how mobile notifications materialize on smartphones. While prior work already investigated how many notifications users receive on a daily basis, which kinds of notifications are valued by users, and how fast users attend notifications, what is missing is an assessment of how many pending notifications users see throughout the day and whether there are different approaches in attending those notifications. Assessing notifications in user studies is challenging from a privacy perspective since notifications are inherently personal. Our second research question (RQ2) is, therefore, how can we assess notifications, including the content, while respecting the privacy of participants. Apart from purely assessing mobile notifications, another aspect is assisting users by improving mobile notification management. The third research question (RQ3) is about novel approaches to improve the state-of-the-art of notification management.

Looking beyond mobile notifications on smartphones, an open research question is the differences between different kinds of devices when it comes to whether users would like to receive notifications on those devices. Our fourth research question (RQ4) is about how various types of personal devices differ in multi-device environments with regard to receiving notifications. Further, we differentiate between notifications on personal devices and on devices that are often shared, such as smart TVs. The fifth research question (RQ5) is about considerations for showing notifications on such devices. Finally, going even further beyond an open research question is how public displays can be used to display personal notifications. Our sixth and final research question (RQ6) is about what to consider when showing highly personal notifications on public displays.

Research Question	No.	Chapter
Assessing Notifications on Mobile Devices		
How do notifications materialize on smartphones, and how are users managing them?	RQ1	Chapter 3
How can we assess notifications in detail while respecting users' privacy?	RQ2	Chapter 3
Improving the Management of Mobile Notifications		
How can we support users with managing mobile notifications?	RQ3	Chapter 4
Beyond Mobile Notifications		
How do various types of personal devices differ in multi-device environments with regards to displaying notifications?	RQ4	Chapter 5
What are the considerations when displaying notifications on smart TVs?	RQ5	Chapter 6
What are the considerations when displaying notifications on public displays?	RQ6	Chapter 6

Table 1.1: The research questions that are being addressed in this thesis.

1.2 Challenges and Research Contributions

A major challenge is that notifications are highly personal and timely. When researching actual notifications that users receive on a daily basis, lab studies are often not suitable. Users receive notifications throughout the day and these often include sensitive communication-related content. For notification research, we often need to trade the internal validity of user studies for external validity by conducting studies in-situ. However, this poses new challenges, such as heterogeneous device environments, little control, and no supervision. We also have to consider if and how the user studies affect the study results. For instance, prompts for questionnaires are typically implemented as notifications as well. A major challenge of this work is how we can create unobtrusive research probes to answer our research questions. Table 1.2 provides an overview of the research probes and contributions in this thesis.

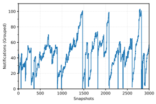
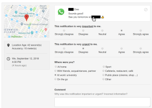
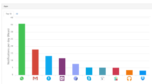
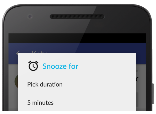
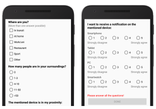
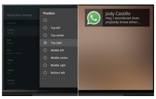
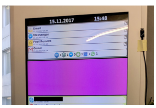
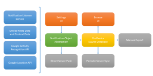
Picture	Description	Chapter
Assessing Notifications on Mobile Devices		
	The *Notification Drawer* data set contains over 8.8 million notification drawer snapshots from almost 4,000 Android devices.	Chapter 3
	*Annotif* is a system that allows users to annotate notifications and share them with researchers in a privacy-respecting manner.	Chapter 3
Improving the Management of Mobile Notifications		
	The *Notification Dashboard* allows users to reflect on the number of notifications they receive on a daily basis.	Chapter 4
	*NHistory* is an Android app that allows users to "snooze" notifications for a duration or to a specific point-in-time.	Chapter 4
Beyond Mobile Notifications		
	The *dedicated ESM device* allows triggering questionnaires at random times throughout the day to avoid affecting other devices.	Chapter 5
	The *TV lab study app* allows video playback while overlying previously recorded user-customizable notifications.	Chapter 6
	*PD Notify* is a system that allows users to mirror their smartphone notifications on nearby public displays.	Chapter 6
Notification Logging Framework		
	*Notification Log* is a notification logging framework for mobile notifications implemented as an open-source Android app.	Chapter 7

Table 1.2: An overview of the research probes and data sets in this thesis.

1.3 Methodology and Evaluation

Throughout this thesis, we use several methods for data collection. In two cases, published apps in app stores to conduct large-scale in-the-wild studies with hundreds and thousands of users. Further, we conducted smaller scale in-situ studies by asking participants to install apps and following up with semi-structured interviews or questionnaires. To complement this approach, we also conducted focus groups, an online survey, and a lab study. From these user studies, we collected empirical quantitative and qualitative data. Using this data, we derived insights to answer our research questions on ubiquitous notifications.

1.4 Research Context

The research for this thesis was conducted between February 2015 and June 2019 at the Institute for Visualization and Interactive Systems at the University of Stuttgart.

1.4.1 Graduate School

The graduate school of the SimTech Cluster of Excellence at the University of Stuttgart provided a framework of checkpoints, seminars, and events that supported the research for this thesis and fostered the interdisciplinary exchange with other researchers. Intermediate steps were regularly presented, including a milestone presentation examined by Prof. Dr. Niels Henze and Prof. Dr. Andrea Bart from the Institute of Applied Analysis and Numerical Simulation.

1.4.2 Publications

This thesis is based on prior scientific publications [166, 174, 177, 180–185]. The work for these publications was conducted in collaboration with Niels Henze, Albrecht Schmidt, Alexandra Voit, David Hägele, Florian Alt, Frank Bastian, Gisela Kollotzek, Huy Viet Le, Jonas Auda, Lucas van der Vekens, Marcus Hepting, Marvin Tiedtke, Philipp Kratzer, Rodrigo Ventura Fierro, Stefan Schneegass, and Sven Mayer. The work led to further publications that are not part of this thesis [172, 176, 178].

In the following, we present the individual contributions for the parts of this thesis that are based on prior scientific publications:

- The work described in Section 3.1 was initiated and driven by the author. He developed and released the system, consisting of an Android app and the server component. He cleaned and analyzed the data set, which he also anonymized and open-sourced. Alexandra Voit and Niels Henze supported the creation of the resulting paper [180], which was published in the proceedings of the conference *Mensch und Computer 2019*.
- Section 3.2 is based on the bachelor thesis project "Annotation and Analysis of Notification Content," by Gisela Kollotzek in 2018. The author initiated the project and was the primary supervisor of the thesis. Alexandra Voit was the second supervisor, and Niels Henze the examiner. Gisela Kollotzek conducted the case study and developed the first version of the system. The author revised the system and analyzed the collected data for the paper [182], which was published in the proceedings of the conference *MUM 2019*. The paper was driven by the author and supported by Alexandra Voit, Niels Henze, and Gisela Kollotzek.
- The system described in Section 4.1 was initiated and developed by the author. The study was conducted and analyzed by the author with the support of Alexandra Voit and Huy Viet Le. Additionally, Niels Henze supported the creation of the resulting workshop paper [185], which was presented in the second iteration of the *Smarttention, Please!* workshop [131] and published in the adjunct proceedings of the conference *MobileHCI 2016*.
- The work described in Section 4.2 is based on the master thesis project "Investigation of Delay Opportunities of Mobile Notifications," by Jonas Auda in 2016. The project was initiated by the author, and he was the primary supervisor for the thesis. Alexandra Voit was the second supervisor, and Niels Henze the examiner. Jonas Auda implemented the system and conducted the studies described in the section. For the resulting paper [177], the author analyzed the collected data. The author and Alexandra Voit analyzed the semi-structured interviews. The author drove the paper with the help of all the mentioned parties, as well as Stefan Schneegass. The

paper was published in the proceedings of the conference *MobileHCI 2018*. Jonas Auda later joined the group led by Stefan Schneegass as a PhD student and drove a follow-up paper [10], which was published in the adjunct proceedings of the conference *CHI 2018*.

- Section 5.1 is based on the bachelor thesis project “Smart Distribution of Notifications Across Multiple Devices,” by Philipp Kratzer in 2015. The author initiated the project and was the supervisor. Niels Henze was the examiner of the thesis. Philipp Kratzer developed the apps described in the section and conducted the study with the help of the author. For the resulting paper [184], the author analyzed the collected data. Alexandra Voit supported the paper writing process. The paper was published in the proceedings of the conference *UbiComp 2016*.
- Section 5.2 is based on the student project “Comparison of the Development of Notification Systems”, by Frank Bastian, David Hägele, and Marvin Tiedtke. The project was initiated by the author (primary supervisor) and Alexandra Voit (second supervisor), and supported by Huy Viet Le (third supervisor). Niels Henze was the examiner of the project. The qualitative user study was conducted by the students and later analyzed by the author and Alexandra Voit. The resulting workshop paper [166] was presented at the third iteration of the *UbiTtention* workshop [173] and published in the adjunct proceedings of the conference *UbiComp 2018*.
- The work described in Section 6.1 is based on the master thesis project “Notification Mechanisms for Smart TVs,” Rodrigo Ventura Fierro in 2015. The author initiated the project. Niels Henze was the primary supervisor, and the author the second supervisor. Albrecht Schmidt was the examiner of the thesis. Rodrigo Ventura Fierro conducted the studies described in the section with the support of the author and Alexandra Voit. Rodrigo Ventura Fierro developed the first iteration of the lab study system. Based on this, the author developed the system used in the lab study. The author drove the resulting paper [174] with the help of all mentioned parties. Additionally, Sven Mayer helped with the statistical analysis. The paper was published in the proceedings of the conference *TVX 2016*.

- Section 6.2 is based on the student project “Evaluation of Using Public Displays for Reading Personal Content within Semi-Public Places,” by Gisela Kollotzek, Lucas van der Vekens, and Marcus Hepting in 2017. The project was initiated by the author. The author and Alexandra Voit supervised the project, Albrecht Schmidt was the examiner. Niels Henze and Florian Alt provided additional support and insights. The author outlined the architecture of the system and developed the smartphone app. The students implemented the server and the public display UI and conducted the study. The author and Alexandra Voit analyzed the collected data and interviews. The resulting paper [183] was published in the adjunct proceedings of the conference *CHI 2018*.
- The system described in Chapter 7 was initially created by the author in 2015 and continuously developed since then. In 2018, the author open-sourced the system. Alexandra Voit and Niels Henze provided feedback for the workshop paper [181], which was presented at the third iteration of the *UbiTtention* workshop [173] and published in the adjunct proceedings of the conference *UbiComp 2018*.

1.4.3 Funding

The following funding bodies partially funded the research for this thesis:

SimTech Cluster of Excellence[1] Parts of the research were funded by the SimTech Cluster of Excellence at the University of Stuttgart within the project network “Reflexion and Contextualisation” using the working title “Modeling Human Behavior for Smart Notification Management in the Context of Ubiquitous Computers.”

Design of Adaptive and Ambient Notification Environments (DAAN)[2] The research for this thesis was partially funded by the German Ministry of Education

[1] `https://www.simtech.uni-stuttgart.de/`
[2] `http://daan.dfki.de/`

and Research (BMBF) within the DAAN project [139]. Project partners were the German Research Centre for Artificial Intelligence (DFKI), Deutsche Telekom, IXDS, University of Stuttgart, Intuity Media Lab, and UdK Berlin.

SFB-TRR 161[1] This work was partially funded by the project "Metrics for Mobile Visualization and Interaction Techniques through Research in the Large" within the SFB-TRR 161 "Quantitative Methods for Visual Computing", supported by the German Research Foundations (DFG).

1.4.4 Collaborations

Socio-Cognitive Systems (SCS) Group, University of Stuttgart Within the Socio-Cognitive Systems group at the University of Stuttgart, led by Prof. Dr. Niels Henze, the ongoing collaborations resulted in a number of publications co-authored with Sven Mayer [90] and Alexandra Voit [158, 159, 162, 164–169].

Human-Computer Interaction (HCI) Group, University of Stuttgart The Human-Computer Interaction group at the University of Stuttgart, led by Prof. Dr. Albrecht Schmidt, closely worked together with the Socio-Cognitive Systems group. The following co-authored publications are the result of collaborations: Alireza Sahami Shirazi et al. [137], Katrin Wolf et al. [192], Lars Lischke et al. [84], Matthias Hoppe et al. [62], Thomas Kubitza et al. [77], Tilman Dingler et al. [34], and Yomna Abdelrahman et al. [1].

External Collaborations Further collaborations include Jonas Auda et al. (University Duisburg-Essen) [10] and Frederik Wiehr et al. (DFKI) [190].

Doctoral Colloquia Parts of this work were discussed at two doctoral colloquia. The first doctoral colloquium was held in conjunction with the conference TVX 2016. It was chaired by Teresa Chambel (University of Lisbon) and Sharon Strover (University of Texas at Austin). Frank Bentley (Verizon Media) supported the discussions. The second doctoral colloquium was held in conjunction with the conference MobileHCI 2017 [172]. It was chaired by Céline Coutrix

[1] `https://www.sfbtrr161.de/`

(Université Grenoble Alpes), Jennifer Pearson (Swansea University), and Andrés Lucero (Aalto University). Mikael B. Skov (Aalborg University) supported the discussions.

Workshop Series Parts of this work have been presented at a series of workshops on smart attention management. The *Smarttention, please!* workshops in conjunction with the conferences MobileHCI 2015 [131, 176] and 2016 [159, 175, 185]. The *UbiTtention* workshops in conjunction with the conferences UbiComp 2016 [77, 161, 164, 190], 2017 [105], 2018 [166, 173, 181], and 2019 [40]. The *Intelligent Notification and Attention Management on Mobile Devices* workshop in conjunction with the conference MUM 2017 [179].

Internships Between October 2016 and February 2017 the work on this thesis was briefly paused due to an internship at Google Research, which was hosted by Yang Li. The work was paused again between May and August 2018 for a second internship at Google Research, which was hosted by Pingmei Xu.

1.5 Thesis Outline

This thesis consists of eight chapters, a bibliography, and an appendix. We present the results and evaluations of multiple empirical studies, a review of related work, discussions, and a conclusion. The thesis is structured as follows:

Chapter 1 - Introduction Motivates the thesis, defines the research questions, and outlines the research context.

Chapter 2 - Background and Related Work Provides background and a review of related work on notifications.

Chapter 3 - Notifications on Mobile Devices Describes the assessment of mobile notifications, specifically mobile notification drawers and an approach for annotating notifications in user studies.

Chapter 4 - Managing Mobile Notifications Introduces two approaches for improving the management of mobile notifications: A notification dashboard and the ability to “snooze” notifications.

Chapter 5 - Notifications in Multi-Device Environments Reports a quantitative study on notifications in environments with multiple different devices and a qualitative follow-up study.

Chapter 6 - Notifications on Large and Pervasive Displays Further expands the set of devices that can notify the user by considering smart TVs and public displays.

Chapter 7 - Notification Logging Framework Describes the architecture and use-cases for an open-source notification logging framework for mobile devices that was used throughout this thesis.

Chapter 8 - Conclusion and Future Work Summarizes the thesis and the research contribution and outlines future work.

2

Background and Related Work

This chapter provides background information and an overview of related work on ubiquitous notifications. The chapter is structured in three parts. We first present notifications and notification systems on current operating systems. We then provide a summary of the author's prior work that directly inspired the creation of this thesis. Finally, we provide an overview of related work on notifications focusing on mobile notifications. This chapter is meant as an introduction to the topic of ubiquitous notifications. We will complement the related work throughout the following chapters as we expand the focus on more devices. Parts of this chapter are based on the background and related work sections of the publications [166, 174, 177, 180–185] that chapters 3 - 7 are based on.

2.1 Notification Definition

The Cambridge Dictionary defines *notification* as "a message that is automatically sent to you on your mobile phone or computer" [17]. The related term *push notification* is defined as "a message sent to a smartphone relating to one of its apps, even when it is not running, or the act of sending such messages" [18]. Although the definition explicitly mentions smartphones, all current dominant operating systems for devices like tablets, smartwatches, desktop PCs, and laptops support (push) notifications. In the following, we provide a brief overview of current notification systems on different devices.

2.2 Notification Systems on Different Devices

Notifications are not a new phenomenon. Landline phones inform users about incoming calls by ringing. Mobile phones that predate smartphones supported ringtones for incoming calls, SMS, and sometimes reminders. Applications running in the background of desktop computers and laptops informed users about various events, such as system alerts, new emails, and instant messages. However, the scope of notifications mainly was limited to a specific set of events on specific devices. Always connected smartphones that can be easily extended by apps downloaded from app stores removed limitations. Notifications are now an operating system (OS)-level feature that all kinds of apps and services can use for all kinds of events. Other devices followed this approach. Notifications are now a feature on all popular operating systems that application developers can expect to be available, typically through a well-defined application programming interface (API). This thesis focuses on these "modern" notifications and different kinds of smart devices. We will now briefly summarize the current state of notifications on current smart devices.

2.2.1 Smartphones

Google's *Android* and Apple's *iOS* are the current dominant mobile operating systems. Notifications are shown on the lock screen and, therefore, one of the first things users see when turning on the screen of the device (see Figure 2.1a). Apps can trigger notifications, typically represented as rectangular boxes with icons and text, and optionally accompanied by auditory and tactile cues. These notifications are collected in the notification drawer, which can typically be accessed at any time by swiping from the top of the screen (see Figure 2.1b). Users have different levels of control about which apps can trigger notifications and how they are alerted. This ranges from individual settings for specific apps to "do not disturb" modes that affect all notifications. The default interaction with a notification is

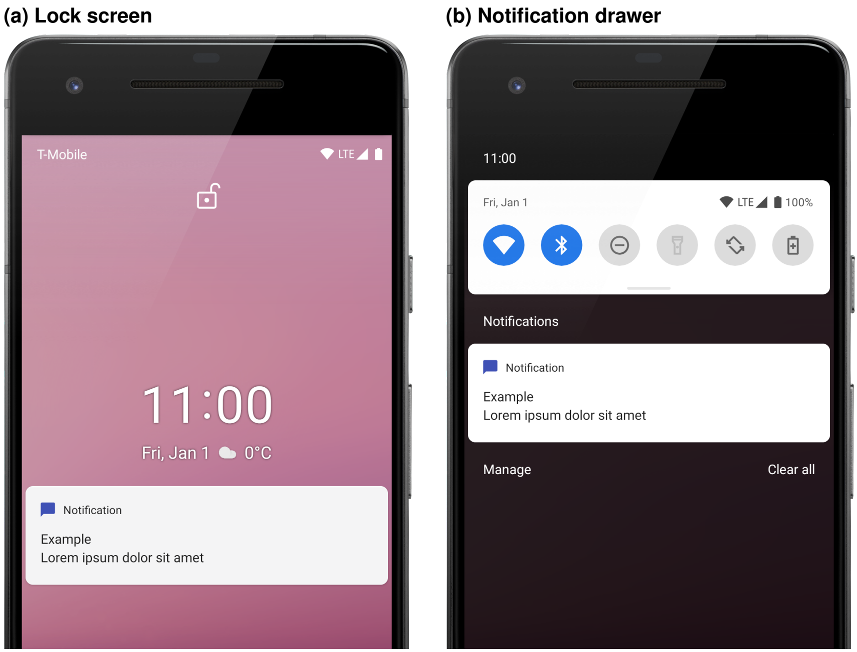


Figure 2.1: Exemplary smartphone notification on Android 11 shown on the lock screen (left) and in the notification drawer (right).

either tapping it or dismissing it by swiping it away. On recent versions of *Android* and *iOS*, notifications offer more interaction options. For example, additional buttons trigger other actions or inline replies for instant messages.

2.2.2 Tablets

Android and *ipadOS* (a variant of *iOS*) are also prominent operating systems for tablets. Overall, notifications are implemented in the same manner as on smartphones. Due to the increased screen size there are slight visual tweaks (see Figure 2.2a), but the overall user experience is very similar to smartphones. One difference is that tablets are often equipped with vibration motors due to their size and notifications can therefore be limited to visual and audio cues only.

Figure 2.2: Exemplary tablet (left) and smartwatch (right) notification.

2.2.3 Smartwartches

With *Wear OS* (previously *Android Wear*) and *watchOS* (a variant of *iOS*), variants of *Android* and *iOS* are also represented on smartwatches. Here, notifications are adjusted to fit the smaller form factor (see Figure 2.2b). Interaction is more focused on pre-defined actions that allow users to take action on notifications without requiring them to open app experiences in full screen. There is also a stronger focus on vibrotactile alerts that subtly tap users on the wrist to inform them about new events. A major difference compared to smartphones and tablets, where the source of notifications is typically an app on the device, smartwatches often allow users to "mirror" notifications from a connected device. This allows smartwatches to conserve battery life by only requiring a wireless connection to the smartwatch. All processing of the events and notifications is done on the connected smartwatch. However, recent smartwatches also allow standalone apps on the device itself, to slowly untether the requirement of a connected smartwatch. Another aspect to consider is that, due to the limited form factor of smartwatches, users might be inclined to attend to smartwatch notifications on a different device with a large screen or better-suited input options.

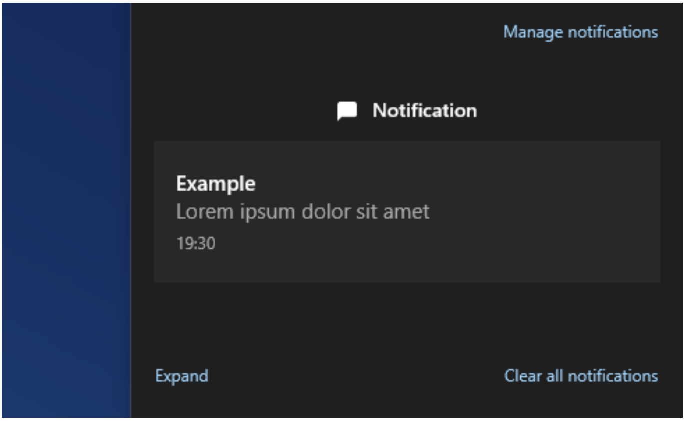


Figure 2.3: Exemplary notification on Windows 10.

2.2.4 Desktop PCs and Laptops

The dominant operating systems for desktop PCs and laptops, Microsoft *Windows* and Apple *macOS*, adopted OS-level notifications similar to smartphones. This includes a standardized API, similar-looking notifications, and a notification drawer. They also offer similar options to control notifications for individual apps and modes that affect notifications from all apps, such as “Do not Disturb” or “Focus Mode.” The exact controls differ on every platform and even on different OS versions.

2.3 Prior Work by the Author

This thesis continues prior research on notifications that the author conducted and was involved in. Although this research is not part of this thesis, it is crucial as a foundation for this thesis. Therefore, we will summarize this work in the following. We will refer to the author of this thesis as Weber throughout the following two sections.

2.3.1 Researching Mobile Notifications at Scale

In 2014, Sahami Shirazi et al. published the first paper on the large-scale assessment of smartphone notifications [137]. During this time, Weber worked as a student assistant, built the system, and collected the data for the paper. The "Desktop Notifications for Android" system was built and an launched at the end of 2012, the data was collected in January - July of 2013, and the paper was published in early 2014. This work contributed the first assessment of notifications at scale, breaking down what kind of notifications users like and dislike, and derived design guidelines for mobile notifications. The data used for the paper consisted of almost 200 million notifications from over 20,000 apps and over 40,000 users. This section summarizes the paper and provides additional background and insights from a perspective of several years in the future.

Apart from the direct findings, this work also set an example of how to investigate notifications at scale by giving users a reason to install the study apparatus. The apparatus was a service that allowed users to view their smartphone notifications on their laptop or desktop PC. It consisted of three major parts: an Android app, a browser extension for Mozilla Firefox and Google Chrome, and the back-end server. Users would install the Android app on their smartphones. The app would then generate a secret code and listen for new notifications on the device. Any new notification would then be forwarded to the server and stored in a database along with the code. The users would then enter the secret code in the browser extensions, periodically polling the server with the code to fetch new notifications. If a new notification was found on the server, the browser extension would display it on the user's desktop.

Users were able to exclude notifications from specific apps from being sent to the server. This functionality was included in the first version of the app for two reasons. First, to prevent apps that create many notifications from spamming the user. Second, to give users control of sending notifications with sensitive content to the server and eventually their laptop or desktop PC. Since the notification content was not encrypted on the server, it was accessible by the researchers. However, the analysis was limited to the notification metadata.

The system was used to collect multiple sets of data. First, the metadata of the notifications that passed through the server. Second, the list of apps that were excluded from sending notifications. Third, answers to a questionnaire that was randomly attached to the notifications on the desktop in the form of a button. Finally, the time between creating a notification and the user clicking on it.

This led to several interesting findings. For instance, 50% of clicks on notifications happened in the first 30 seconds after the notification was created. Further, the app *WhatsApp* dominated the list of created notifications. The system also collected 4,964 quantitative ratings and 796 qualitative answers from 4,816 users. This means that the system only collected about one rating per participating user. However, since this was done at scale, it provided insights into how users perceive notifications. The click time was found to be faster for apps that were rated important, and apps that were excluded from sending notifications received lower importance ratings. Finally, notifications from messaging apps notably stood out as more important than other notifications.

The researchers concluded the paper with the following five insights about mobile notifications [137]:

1. *“[The] nature of notifications is disruptive”*
2. *“Important notifications do not necessary cause immediate attention”*
3. *“Notifications are for messaging”*
4. *“Important notifications are about people and events”*
5. *“Not all notifications are important”*

While the system provided novel insights into mobile notifications and users’ preferences and interactions, it was not without limitations. The system simply tried to send out notifications once and then discarded them. If the Android device was offline when a new notification was created, it would not retry to send it. Users also had to dismiss notifications twice. Once on the smartphone and once for the copy on the desktop. This, and the aspect that users might see notifications on the desktop before the smartphone, might also have altered their behavior when interacting with notifications on the smartphone. Rating the notifications for their importance was also done on the desktop instead of the smartphone. Moreover, since there was only about one questionnaire rating per user, there were few insights into generalizable findings, and there was a bias towards apps that created more notifications. Finally, the system required the user to install and set it up on two devices, which might have biased the user base towards more technically savvy users.

Although the system was built to research mobile notifications, mirroring notifications on other devices opened new research questions regarding multi-device notifications.

2.3.2 Expanding to Multi-Device Environments

In late 2014, Weber extended the "Desktop Notifications for Android" system for his diploma thesis [171]. The update addressed shortcomings of the initial version of the system. It replaced the connection code for pairing devices with a one-click sign-in button that allowed users to sign in with their Google account. All devices signed in with the same account were automatically paired. The system allowed pairing a virtually unlimited number of Android devices and web browsers. Further, the system was updated to broadcast notifications from one device to all other devices, including other Android devices. The system was updated to differentiate between Android smartphones, Android tablets, and web browsers. The simple boolean blacklist was also updated to allow disabling sending notifications to specific types of devices. Users could control whether notifications should be broadcasted to other Android smartphones, Android tablets, and web browsers for each app. Additionally, a new "Private Mode" allowed sending the information that a new notification from a specific app was triggered but omitted sending the actual notification content to the server. For instance, the user would see an email notification on the origin device, including the content of the email, and broadcasted notifications without the content on other devices. Furthermore, dismissing a notification on one device dismissed the notifications on all devices at once. Since users could accidentally dismiss a notification across devices, the update also included a notification history accessible within the Android app. Last, the user could see a list of all devices paired with the account.

With this updated system in place, Weber collected new data from over 36,000 apps and more than 33,000 users. A previous limitation was that questionnaires about notifications were triggered on the desktop. This was also changed to show the questionnaires as a notification on Android. Users were asked whether a specific notification should be shown on a specific device. The options were smartphone, tablet, smartwatch, smartglasses, desktop PC, laptop, and TV. Smartphone, desktop PC, laptop, and tablet received the highest agreement ratings, followed by smartwatch, smartglasses, and TV. Another questionnaire was attached to the notifications of receiving devices (both Android and in the browser). It showed that "messenger" notifications were valued on every type of device. An analysis of the blacklist and the "Private mode" revealed that apps in the "Tools" category were blacklisted most often. In contrast, communication apps were set to the private mode most often.

2.3.3 Limitations and Learnings

While this improved system enabled Weber to collect more data and gain first insights into the differences between devices and apps in multi-device environments, there were still several limitations. The app had a large user base, but all changes required a considerable effort to coordinate to implement changes on Android across multiple Android versions, the server, and the browser extensions across multiple browsers and operating systems. The large number of notifications sent through the system every second made it necessary to consider exactly what data to log. Further, handling the actual notification content was still a challenge due to privacy concerns. Finally, user feedback via email and the Google Play Store comments made it clear that users downloaded the system for its functionality and disliked the questionnaires. When shown repeated questionnaires, users clearly expressed their discontent. These limitations made it increasingly difficult to adapt the system to address new research questions.

This resulted in several learnings for future work. For example, research probes should be limited to more focused apps to reduce the system complexity. Data collection could be improved by batching log data on-device and periodically sending the data to the server. The data collection should also follow data minimization rules by only collecting data necessary to answer the research questions. Privacy sensitive data should be hashed or encrypted. Finally, depending on the research question, research probes should be deployed in environments with different levels of control.

2.4 Related Work

In this section, we provide an overview of related work. We first focus on mobile notifications and then expand the scope beyond mobile. We then focus on interruptions caused by notifications and potentially adverse effects. Subsequently, we provide an overview of notification management approaches. Finally, we discuss research in the wild and using app stores for research.

2.4.1 Mobile Notifications

Users are confronted with more and more notifications in their daily lives. Notifications are a popular method to engage users and inform them proactively, e.g., about new messages, events, or updates. Notifications use visual, tactile, and auditory cues to gain the users' attention [66].

2.4.1.1 Communication and Messaging

Nowadays, smartphones, tablets, and an increasing number of wearables have become an essential part of our everyday life. Pielot et al. showed that while communication-related notifications help to make users feel more connected to others, receiving too many notifications can get overwhelming [121]. In an in-situ study with 15 participants, Pielot et al. investigated how users interact with smartphone notifications. Over the course of one week, participants received an average of 63.5 smartphone notifications per day, mostly from instant messaging and email applications. Furthermore, the study showed that notifications are viewed within minutes, even when the smartphone was put in silent mode. In 2018, Pielot et al. revisited mobile notifications in a study with 278 participants [127]. The results again showed the importance of messaging notifications. Participants were fast to attend messaging notifications, while other types of notifications were either removed quickly or left unattended for longer periods.

Messaging is a recurrent topic in related work. Instant messaging is a flexible way of communication, that can vary between synchronous and asynchronous discussions [11, 70, 98]. Mobile phones enabled text messaging as a popular communication method [13]. Researchers investigated "traditional" SMS usage and compared it with modern instant messaging (IM) apps such as *WhatsApp* [23].

Church et al. found that cost and social influence are reasons for *WhatsApp* overtaking *SMS* messaging [23]. Dingler et al. explored the attentiveness of users toward mobile messages [33]. They found that users were attentive to messages for approximately 12 hours per day. Researchers found that participants attend notifications about individual (1:1) chats faster than group chats [127]. Avrahami et al. showed that the responsiveness toward instant messaging is affected by the context and the presentation of messages [11]. Mehrotra et al. found that the sender of messages can have an impact on how notifications are perceived [95] and Pielot et al. identified features to predict if a user will attend a message within a specified period [122].

The number of generated notifications is constantly increasing [121]. Ill-timed notifications can also distract or interrupt the recipient [88, 121]. They can be distracting, might cause negative emotions, or are just not important for their recipients [114, 121, 126, 137]. Czerwinski et al. explored the adverse effects of instant messaging interruptions on different kinds of tasks [30]. Other work looked at the attentional cost of receiving notifications [147] and relevant interruptions [29, 44, 49, 71]. A notification recipient might benefit from valuable information that he or she receives in a proactive manner [30, 95].

2.4.1.2 Interactive Notifications

Notifications are not limited to messaging. Dingler et al. investigated the use of notifications for microlearning sessions on the go [34]. The researchers built an Android app that triggered microlearning notifications at random times and depending on a model [123] that also included the time of the last received notification.

2.4.1.3 Notification Modalities

Using visual, tactile, and auditory cues, these devices can use notifications to gain the user's attention [66]. Hansson et al. compared public with private and subtle with intrusive notification cues [54]. The researchers presented a model for visualizing the difference between tactile and auditory notifications. According to their model, tactile notifications fall into the subtle and private category, while auditory notifications are intrusive and public. Exler et al. investigated the perceptibility of different notification types depending on the position of the smartphone [39]. The researchers looked at sound, tactile, and LEDs, with the smartphone being placed on a table, in the pocket, or in the backpack. The results of a lab study with 36 participants showed that overall tactile notifications were favored due to the low obtrusiveness. Sound was favored for important notifications, and using a LED was found to be suitable for unimportant notifications.

2.4.1.4 Logging Notifications

A common theme in related work on notification research is the need to log notifications. Pielot et al. [121], Dingler et al. [33], and Mehrotra et al. [94, 95] logged mobile notifications to provide further insights into what kind of notifications users receive on a daily basis and how they are perceived. In a recently published work on the importance of notification content, Visuri et al. were surprised by a participant of a pilot study not clearing their notifications [156]. The researchers asked participants to label notifications regarding how important and timely they were perceived. They found that many users often dismiss or ignore notifications, and the notification content plays an important role in regard to how users act. The researchers suggest applying semantic analysis to detect unwanted notifications, which requires accessing and logging the content of notifications.

2.4.2 Beyond Mobile

Notifications are no longer limited to single devices. With smartphones becoming ubiquitous and new kinds of connected devices entering our everyday lives and homes, notifications follow the users throughout the day. For instance, email notifications were once limited to desktop computers. Today, many kinds of devices can alert about incoming emails, including laptops, smartphones, smartwatches, fitness trackers, and tablet computers. In the future, Internet of Things (IoT) devices like smart light bulbs, intelligent speakers, and pervasive displays will also notify the users. All these devices differ in their modalities used to notify users but also in the modalities users can react to notifications. However, implementation-specific differences determine how users experience these notifications, even for devices of the same type. For instance, in the mobile operating system Android, notifications are designed as opt-out, while on iOS, they are opt-in.

2.4.2.1 Desktop Computers

Research on interruptions caused by notifications predates the current set of smart devices. In 2000, Czerwinski et al. investigated the effects of instant messaging on different tasks on desktop computers [30]. Their results show adverse effects of notifications on the task performance. Further, they found that the adverse effects depend on the task type. The authors argued that with the rise in popularity of instant messaging systems, guidelines for minimizing adverse effects and maximizing the value for the users have to be developed. This can lead to adverse effects, such as increased stress [193], inattention [78], and reduced task performance [30]. Czerwinski et al. investigated the effects of interruptions on task switching on traditional desktop PCs [31]. Notifications on desktop computers tend to provide a passive awareness of incoming information rather than prompting users to change their current primary tasks [67]. When notifications are turned off on desktop computers, some users can increase the performance of their primary tasks; however, other users interrupt themselves to check for information manually [67]. While notifications cause interruptions, they are still valued by users because they provide awareness [67].

2.4.2.2 Smartwatches

An investigation of smartwatch usage revealed that smartwatches are used briefly and frequently during the day [153]. Users value that they can quickly check the information on their smartwatches without being considered rude in social interactions and have the opportunity to decide if there is a need to interrupt their current primary tasks. Furthermore, smartwatches offer less disrupting access to incoming notifications than smartphones [19, 128, 153]. Similar to smartphone notifications, users interact more with notifications about communication with other people [136, 153] and calendar events [136] on smartwatches.

Sahami Shirazi and Henze also conducted an in-situ study about notifications on smartwatches [136]. The researchers collected responses in an online survey on smartwatches from 440 participants and contrasted the results with an in-situ study, rating individual notifications on smartwatches. According to the results, the importance of a notification does not only depend on the notification type but also on the device that shows them. Due to the small size of smartwatches, they are more of a "read-only" device and not a replacement for smartphones. Their results further show that, on smartwatches, the most important notification category was not messaging, but calendar and VoIP. Lee et al. provided further insights about notifications on smartwatches [80]. The researcher explored reducing the distraction of smartwatch users with deep learning. Pearson et al. explored a different use case for smartwatches by using them as public displays [115]. The researchers proposed showing different content on smartwatches for the "wearer," the "glancer," and the "public." According to the results of their studies, it is socially acceptable.

2.4.2.3 Ambient Notifications

Müller et al. all investigated using ambient light to alert users. The light was positioned in the periphery of the users. In a lab study, they found it to be of similar usefulness as traditional pop-up notifications. However, the authors mentioned privacy concerns since it can be seen by others. They further discussed that tactile notifications are better suited for private alerts and that this approach still needs to be investigated in situ. Kubitza et al. integrated notifications in an IoT infrastructure for intelligent living environments. [77]. Using this system, the interconnectivity of the IoT devices can be leveraged to, for instance, alert users about a notification on their smartphone using smart light bulbs, intelligent speakers, and pervasive displays. Voit et al. developed a smart plant system that notifies users about water levels using ambient light or smartphone notifications [163].

2.4.2.4 Embodied Notifications

Schneegass and Rzayev proposed using electrical muscle stimulation (EMS) for implicit notifications [140]. Instead of explicitly gaining the user's attention, such a system could implicitly make the user perform certain actions. For example, making the user twist the arm to look at the watch or move the arm towards the phone. Poguntke (née Kettner) et al. developed a wrist-worn device that applies pressure feedback [74]. According to the researchers, this device can provide tactile notifications with reduced stress levels compared to traditional vibrotactile feedback. With *Slappyfications*, Günther et al. provided a humorous take on notification research [53]. The researchers proposed using pokes and slaps to notify users. As a final level of escalation, they proposed the *STEAM-HAMMER* to ensure users cannot miss a notification.

2.4.2.5 Multi-device Environments

While a body of work investigated notifications on individual devices, little is known about notifications in multi-device environments. Weber found that there is a need for a mechanism to coordinate the distribution of notifications across the user's devices [176]. Such a mechanism has to take multiple factors into account, such as when a notification should be optimally delivered and which of the user's device(s) should display the notification. Regarding when notifications should be optimally delivered, Okoshi et al. developed *Attelia II*. This system delivers notifications at identified breakpoints based on the user's multi-device usage and the user's physical activities [110]. The evaluation results of *Attelia II* revealed that delivering notifications at breakpoints in multi-device environments reduces the perceived workload of the user. Fallman and Yttergren proposed a system for mobile phones that detects nearby users and chooses an appropriate notification modality accordingly [41]. With *NotifyMeHere*, Mehrotra et al. explored intelligent notification delivery in multi-device environments [92]. Under the assumption that smartphone notifications are handled on another device if the user did not interact with the smartphone when the notification was dismissed, the authors explored models of whether a user wants to be notified on the smartphone or "another" device.

2.4.3 Interruptions and Adverse Effects

While notifications allow us to be connected, they can also cause interruptions. The disruptive nature of interruptions and task switching has been an important research topic for many years [31, 63]. While not all interruptions are disruptive [47], Adamczyk and Bailey showed that different timings of interruptions have different effects on users [2]. Mehrotra et al. found that the perceived disruption of a notification is influenced by several factors, including the notification's presentation, the relationship between the sender and receiver, and the task the user is engaged in [94, 95]. Prior studies investigated what makes interruptions disruptive [47]. Interruptions can delay task completion by up to four times [83]. While interruptions may cause inattention [78], intense phone use does not predict negative well-being [72]. In a world of constant connection, being unavailable is an interesting research topic [15]. Aranda and Baig discussed how users are more and more dependent on smartphones, difficulty to disconnect, and "the fear of missing out" [9]. Mehrotra et al. investigated the effect of cognitive and physical factors on the response time and the disruption caused by interruptions through incoming notifications [95]. In terms of negative effects, work by Leiva et al. shows that interruptions caused by mobile notifications introduce a significant overhead when completing tasks [83]. Recent work by Kushlev et al. shows that smartphone notifications increase inattention and hyperactivity symptoms [78].

Smartphone users often do not realize how many notifications they receive [185]. Sahami et al. found that a large number of notifications are issued by messaging applications [137, 176]. On the one hand, users value notifications issued by such applications. On the other hand, not all notifications that users receive are considered important. Church and Oliveira compared *SMS* to instant messaging applications like *WhatsApp* [23]. Their study revealed several concerns regarding *WhatsApp* messages and notifications, e.g., coping with too many messages or interruptions and the fear of missing business-related messages if notification modalities are switched off. Understanding how users handle messaging notifications might help to build messaging services which do not overload users by issuing too many notifications. Further negative effects include decreased productivity and slower and more error-prone performance [2, 12, 49, 117, 138].

Exler et al. surveyed 68 participants on how smartphone notifications are interrupting and disturbing at specific locations [36]. They found that users are more receptive to interruptions while waiting (e.g., bus stations and parking lots) and less at movie theaters, libraries, and restaurants. Mayer et al. evaluated how mobile notifications are disrupting conversations [89]. The researchers set up a simulated conversation environment, and used eye tracking combined with a qualitative analysis.

Simply disabling notifications entirely is no suitable solution [125, 126]. Thus, managing notifications to not continuously disturb the user is a crucial task. The type of the primary task, its complexity, its duration, the length and number of interruptions influence the perceived difficulty of continuing a task after an interruption [31]. In a diary study, Czerwinski et al. showed that returning to tasks after being interrupted is hard [31]. Vardhan et al. discussed the balance of convenience and privacy of mobile notifications [152], and Lee et al. investigated smartphone "overuse" and the role of messaging [82].

2.4.4 Notification Management

A body of prior work has explored how mobile notifications can be better managed. Researchers investigated what users do when they sense notifications [21] and which strategies users apply to cope with notifications. Gallud and Tesoriero suggest moving from sound to visual notifications [45]. Auda et al. explored a system for rule-based notification deferral by suppressing, summarizing, or automatically snoozing notifications [10]. Mehrotra et al. took this a step further by automatically suggesting rules based on usage patterns [93]. The researchers found that the notification's title and the user's location can be used as features to determine whether a message will be dismissed.

2.4.4.1 Call Predictions

Phone calls are urgent notifications that users have to attend in a small timeframe in order to not miss the call [118]. Using anonymous data from 418 users, Pielot created a model to predict whether a user will pick up a call. The researcher was able to predict this with an accuracy of 83.2% by using features such as when the user was last using the device, the time passed since the last call, when the ringer mode was changed last, and the device orientation.

2.4.4.2 Opportune Moments and Breakpoints

A number of research projects are focusing on the approach to delivering notifications at opportune moments instead of delivering them immediately [43, 44, 114, 130, 143]. With *Attelia*, Okoshi et al. developed a middleware that defers notifications to so-called breakpoints - times between two consecutive activities [106, 109–111]. Deferring notifications to these breakpoints has been shown to lessen disruptive effects [44]; however, this has to be balanced with social expectations to reply quickly [177]. *Attelia* runs on the user's smartphone and can detect breakpoints of the user's activity on his or her mobile device. Further, it can detect physical breakpoints through smartwatches. According to the researchers, determining which device to notify the user on is a challenge for future research. Okoshi et al. conducted an in-lab and an in-the-wild study of *Attelia* [108]. The results showed significantly reduced frustration if interruptions were triggered during breakpoints.

Okoshi et al. had the opportunity to conduct a real-world, large-scale study within the *Yahoo! Japan* app [112, 113]. More than 680,000 users participated in this study, with the goal of detecting opportune moments to interrupt users. The researchers found a significant reduction of response time compared to issuing notifications directly. Their model initially performed worse on weekends, but due to the large amount of data collected, they were able to improve the model quickly. Tsubouchi and Okoshi followed up on their large-scale research in the *Yahoo! Japan* app for detecting interruptibility based on activity [149]. The researchers tweaked the features of the model. Thanks to the large amount of data collected, it was possible to adapt and improve the model quickly.

Other approaches explored models to better time interruptions [2, 116, 148, 150]. *SCAN* is another approach of a notification system that takes the social context into account [114]. Fischer et al. investigated mobile phone activity as an indicator of opportune moments to deliver notifications [43]. Iqbal and Bailey investigated the effects of intelligent notification management on users and their tasks [65]. The researchers built a system that uses statistical models to defer notifications until breakpoints, resulting in reduced frustration and reaction time. Using a context-aware computing device, Ho and Intille detected activity transitions [60]. They found that messages delivered in this activity transitions were better received. Using machine learning techniques, Pielot et al. investigated the possibility of predicting the user's attentiveness to text messages [122]. Mehrotra et al. also presented a system that generates notification rules based on received notifications [93]. To voluntary engage users to interact with recommended content, Pielot et al. used a machine learning approach to determine opportune moments for notification delivery [120]. Poguntke et al. investigated different delay modes for notifications [129]. They compared a fixed interval of one hour, a user-defined interval (defaulted to 10 minutes), and a sender-dependent interval (defaulted to an hour). Anderson et al. recently published a survey on attention management systems [5].

2.4.4.3 Sensing Context

Pielot et al. explored boredom detection and using the boredom state to send out proactive recommendations [119, 123]. They proposed sending out fewer recommendations when the user is busy and more when the user is bored. Dingler et al. investigated if detected boredom can be used to engage a user in micro-learning sessions through notifications [34].

Goyal and Fussell explored timing interruptions based on electrodermal activity derived from galvanic skin response [51]. According to their results, this approach resulted in significantly reduced distractions. Exler et al. investigated the detection of a smartphone user's distraction based on typing and touch gestures [38]. The results of their study showed that users typed slower and made more errors depending on the workload. The researchers conclude that this insight can be used as a measure of distraction. Visuri and van Berkel published a survey paper on the importance of attention in human-computer interaction and an overview of mobile sensing [154].

2.4.4.4 People and Content

The importance and urgency of notifications depend on their content and context [94]. Mehrotra et al. used the context of a notification recipient in combination with the content of the notification to realize a non-disruptive notification mechanism [94]. In the case of communication-related notifications, the relationship between the sender and the user matters as well [95]. Users might not accept a notification management system that removes important notifications [95].

2.4.4.5 Research Tools

Pejovic and Musolesi proposed *InterruptMe*, a library for interruption management for Android [116]. Obuchi et al. investigated pushing ESM questionnaires when breakpoints in a user's activity are detected [104]. The authors report up to a 70% improvement in the response time when the user's activity switched from "walking" to being "stationary." Okoshi et al. proposed creating an "Interruptibility Layer" as a middleware on top of the operating system [107]. The researchers highlighted that this likely would need to be developed in collaboration with the creators of the dominant operating system developers, such as Apple (iOS) and Google (Android).

PrefMiner is a system to generate rules for notification management automatically [93]. In his PhD thesis, Mehrotra proposed a framework for intelligent mobile notifications [91]. He explored multiple models to predict opportune moments for notification delivery. Further, Mehrotra et al. investigated how mobile experience sampling can be improved [96]. While mobile experience sampling is a useful source of data, the quality of data the varies. One reason for this is that users might be too busy to attend the experience sampling prompts. The researchers suggested detecting breakpoints for opportune moments to prompt users for questionnaires.

Visuri et al. proposed a cluster-based user model for predicting interruptibility for manual data collection [155]. A use case for this is quantified-self applications, which trigger alert dialogs for data collection.

2.4.5 Research in the Wild

A challenge of research on mobile notifications is that they are highly context-dependent and received around the clock. To overcome this challenge, some prior work in this area moved from lab studies to in-the-wild studies.

A series of publications were concerned with whether it is "worth the hassle" to conduct research in the field. The authors found that at the time (2004), most HCI projects conducted their evaluations in the lab [76]. The authors argue that while "mobile systems are highly context-dependent," conducting studies in the field is "difficult," "time consuming," and the "added value is unknown." The authors tested a system for usability problems in a lab in a field condition in their work. They found similar usability problems in both conditions but argued that there is a "lack of control" in the field condition. On the other hand, they argue that it is challenging to ensure that everything is covered in the lab condition. For this particular system and study, the authors concluded that both lab and field studies have advantages and disadvantages. Since they found similar usability problems in both conditions, they conclude that the added value of conducting studies in the field is "very little."

In a follow-up work in 2006, a different set of authors (Nielsen et al.) also compared lab and field environments in empirical studies [102]. In their comparison, they identified significantly more usability problems in the field. The authors explain that the field condition revealed problems with interactions and cognitive load that was not identified in the lab and concluded that it is indeed "worth the hassle" to conduct research in the field.

A year later, in 2007, Rogers et al. investigated why it is worth the hassle [133]. The authors argue that evaluating applications in ubiquitous computing environments is challenging due to their context of use. They explain that metrics used in traditional studies in a lab are optimized for this controlled environment, thus, failing to capture all aspects in more uncontrolled environments. The authors further mention living labs that are designed to simulate real environments to counter these effects. In the paper, the authors discuss the questions of how long should studies in ubiquitous computing environments be conducted, how much and what data to collect, and how the findings can be fed back into the design process. They discuss the challenges of evaluating ubiquitous computing applications that are used over extended periods while users are on the go and doing other things. They mention adapting existing metrics and heuristics and using new intervention evaluation methods such as the experience sampling method. The authors describe case studies for in-situ research and conclude that this effort was successful but expensive in both time and the required effort. The results of the case studies show that here the lab was no option, as the context was needed to capture all aspects and that the environment directly affects the user experience. The authors conclude the paper with the following statement: “Finally, it is impossible, and nor is it desirable, to capture everything when in situ. The key is to use various methods that reveal both hoped for and unexpected effects of the context of use.” [133]

In 2014, ten years after the first paper asking whether conducting studies in the field is worth the hassle, two of the original authors looked at the state of mobile HCI research in the past decade [75]. Kjeldskov and Skov conducted a literature review regarding field and lab studies in the mobile HCI context. They conclude that in the end, both approaches are needed. They summarize that in lab studies “data is typically gathered with precise instruments [...] in an artificial environment where it cannot be disturbed from the outside.” According to the authors, the advantages of lab studies are the “ability to focus on detail,” “high replicability,” and “large experimental control.” The disadvantages are “limited relations to the real world,” “unknown external validity,” and “typically low level of ecological validity.” They found that in field studies, data is usually “gathered through observations, interviews and surveying techniques.” The advantages are capturing a “large amount of rich and grounded data” with a “high level of ecological validity.” The disadvantages are “unknown biases” in the field, “unknown external validity/generalizability,” and “typically low level of control.” While both approaches are needed, the authors highlighted that with mobile HCI evolving, the need to consider the complexity of the world is increasing. The authors go further to suggest that the uncontrollable nature of field studies should be embraced. They argue that the value of field studies is that they are “real” and “messy.” Further, they highlight the opportunity to conduct field studies over extended periods to capture the sustained use of systems. The authors conclude their paper with the sentiment that conducting studies in the lab or the field should not be a matter of “if or why” but rather a “when and how.”

2.4.6 App Stores and External Validity

Henze and Pielot explored how app stores can provide external validity for mobile HCI [56]. In the article, the researchers discuss that *“considering realistic contexts in traditional lab studies is often not even possible because we know too little about what the realistic contexts are”* [56]. They further provide examples of research probes that leveraged the reach of app stores to distribute applications to users in the wild.

Henze et al. also discussed the trade-off between opt-in and opt-out for consent in in-the-wild studies [57]. While opt-out allows for greater data collection, it also poses legal and ethical challenges. Using multiple case studies, the researchers found that many users may use apps only for short periods and that users expect research apps to offer a similar user experience to commercial products. An example of such a research app was the *Desktop Notifications* service that allowed users to synchronize notifications across devices while enabling researchers to gain insights on notifications in-the-wild from a large user base [176].

In 2013, Henze et al. published “ten steps to conduct a large-scale study” [58]. The steps are as follows [58]: *(1)* Identify the research goals. *(2)* Select a study method and *(3)* devise an incentive mechanism. *(4)* Then select the target platform, and *(5)* develop the application, *(6)* including a mechanism to collect data. *(7)* The app should provide informed consent about what data is being collected. *(8)* Then publish the application, and *(9)* continuously monitor the data. *(10)* Finally, filter and analyze the data to answer the research questions.

Exler et al. discuss the difficulties of creating data sets [37]. In particular, they investigated using community-driven data sets using crowd-funded data. The authors discuss issues and limitations, such as data labeling.

2.5 Summary

In this chapter, we provided a brief overview of how notifications on different kind of devices are currently implemented. We then provided a summary of the author's prior work that directly precedes the work in this thesis. Subsequently, we expanded to scope to provide an overview of related work. We first discussed mobile notifications on ubiquitous smartphones. Then we then looked at work beyond smartphone notifications. A common theme in notification research is interruptions and adverse effects caused by notifications. We provided an overview of work in this field and followed the section by discussing notification management approaches to reduce these adverse effects. Subsequently, we discussed the different approaches of notification research in the lab compared to in-the-wild studies. Finally, we briefly provided an overview of how to conduct studies with a high external validity by leveraging app stores.

Prior work has shown how many smartphone notifications users receive per day. However, what is not yet known is how these notifications materialize on smartphones and how users manage them (RQ1). Another aspect of notifications is the perceived importance which depends on the notification content. Researching notification content is challenging, as this raises privacy concerns. Another open question is assessing notification content in detail while respecting users' privacy (RQ2). While a number of approaches to improve notification management exist, this is not a solved problem yet. The next question is, therefore, how we can support users with managing notifications (RQ3). Looking beyond the smartphone, we have seen that research started to expand to other devices. However, most research focuses on single devices at a time. An open question is how various types of personal devices differ in multi-device environments regarding displaying notifications (RQ4). Nowadays, smart TVs are a common type of device in multi-device environments. There is little research regarding the considerations when displaying notifications on smart TVs (RQ5). Finally, expanding the scope further, public displays are becoming more and more ubiquitous. The final research question concerns the considerations when displaying notifications on public displays (RQ6). In the following chapters, we will address these research questions.

3

Notifications on Mobile Devices

In the past decade, smartphones exploded in popularity. They are always connected and always with the user. Coupled with increased processing power and high-resolution touch screens, this created a paradigm shift in how we consume information. While traditional mobile phones are mostly limited to specific features, such as phone calls, text messages, and alarms, smartphones can be easily extended by downloading additional apps from app stores. And with notifications being a core feature of smartphones, these apps can proactively provide users with information from a multitude of services. From new email notifications to breaking news and social media updates, in many cases users do not need to open apps to receive new information.

Prior work already investigated which kind of notifications users receive on their smartphone [121], how fast they attend notifications [137], and the effect of interruptions caused by notifications [31, 63]. However, what is still missing is an understanding about how notifications materialize on smartphones. Current smartphones show notifications in notification drawers until they are either attended on or dismissed. How many notifications users let accumulate in notification drawers and whether there are different strategies for managing these notifications are still open research questions (RQ1). Further, a major challenge when researching notifications is that they are inherently personal. Prior work focused on reporting aggregated information about notifications to provide external validity but also to protect participants' privacy. Another important research question is how we can enable researchers to gain deeper insights into notifications, i.e., the actual content of notifications, while protecting the participants' privacy (RQ2).

In this chapter, we will first report the results of a large-scale observational in-the-wild study on mobile notification drawers. In the second part of the chapter, we introduce a privacy-aware system for annotating notifications in user studies.

Parts of this chapter are based on the following publications:

D. Weber, A. Voit, and N. Henze. "Clear All: A Large-Scale Observational Study on Mobile Notification Drawers." In: *Proceedings of Mensch und Computer 2019*. MuC '19. Hamburg, Germany: ACM, 2019, pp. 361–372. ISBN: 978-1-4503-7198-8. DOI: 10.1145/3340764.3340765

D. Weber, A. Voit, G. Kollotzek, and N. Henze. "Annotif: A System for Annotating Mobile Notifcations in User Studies." In: *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia*. MUM '19. Pisa, Italy: ACM, 2019, 24:1–24:12. ISBN: 978-1-4503-7624-2. DOI: 10.1145/3365610.3365611

3.1 Mobile Notification Drawers

When users of current smartphones turn on the display of the device, they are usually greeted with a lock screen, consisting of a large clock and a list of notifications below it. On the dominant mobile operating systems Android and iOS, this list of notifications can also be accessed at any time by swiping down from the top of the screen. This universally accessible list is an important feature of current smartphones, as it enables asynchronous communication and provides users with proactive information. The notification list is commonly referred to as the notification *drawer* (Android), notification *center* (iOS), notification *tray*, or notification *panel*. We use the term notification drawer throughout this thesis.

Although a large body of prior work on notifications exists, the notification drawer on smartphones as the central place to view and attend notifications has not been explored in detail so far. However, this is a crucial aspect for a complete understanding of mobile notifications. In the following, we complement prior work by reporting the results of a large-scale observational study on notification drawers of current smartphones. Using a research-in-the-wild approach, we periodically sampled the contents of notification drawers on Android devices. We collected 8.8 million notification drawer snapshots from almost four thousand devices. Based on this data, we present a novel analysis on the number of notifications in notification drawers and the positioning of different notification categories. Further, we propose three different user types regarding the management of notifications.

3.1.1 Notifications on Android

Notifications were an integral feature of the Android mobile operating system since the first version. Notifications are opt-out, meaning that all installed apps can post notifications by default without asking the user for permission. Notifications may use visual, tactile, or sound cues to gain the user's attention [66]. All notifications end up in the notification drawer that is accessible by swiping down from the top of the screen (see Figure 3.1). Since Android 5.0, notifications are shown on the lock screen by default as well. Notifications can contain action buttons [34], expandable text, and images. Users can click on notifications to take action or swipe to the left or right to dismiss them. By clicking "clear all," users can dismiss all notifications at once.

As summarized in Chapter 2, a body of prior work investigated which notifications users receive, how they are valued, interruptions, and means to reduce adverse effects. However, the notification drawer as the central place to view and attend notifications has yet to be investigated. To fill this gap in prior work and to create a more complete understanding of mobile notifications, we explored notifications drawers in an in-the-wild study.

3.1.2 Study

We conducted a large-scale observational in-the-wild study on the content of notification drawers. Our research question was how many and which kind of notifications can be found in notification drawers, and whether different notification management approaches exist.

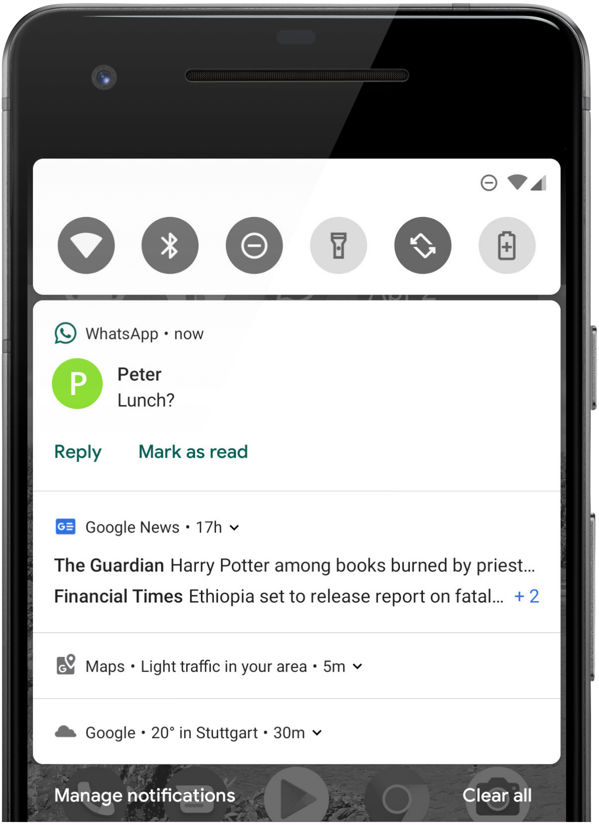


Figure 3.1: The Android 9.0 (Pie) notification drawer showing four different kinds of notifications about a new message, news articles, traffic updates, and the current weather.

3.1.2.1 Apparatus

We developed an Android app that allowed us to snapshot the content of notification drawers in-the-wild in an unobtrusive manner. Our goal was for users to install the app on their own, without explicitly recruiting participants. We developed an Android app that allows users to log and explore their notifications in a local history. The added value for users is the option to look up accidentally dismissed notifications or reflect on notifications they received throughout the day.

The app supports the Android versions 5.0 - 9.0, with Android 9.0 being the most recent Android version when this work was conducted. According to the Android distribution dashboard [6], this covered 96.50% of all active Android devices. The app uses the *Notification Listener Service* API [7] to access the notifications on the device. The notification access for the app has to be explicitly enabled by the user in the device settings. Once enabled, all newly created notifications are stored in a local *SQLite* database. Users can then browse their notifications in a list and select individual notifications to read the text in detail.

3.1.2.2 Data Collection and Consent

After the user installed the app and permitted the app to access the device's notifications, the app displayed a dialog asking the user to opt into the anonymous data collection. Inspired by prior research on asking for consent in in-the-wild studies [124], the dialog contained the options "Agree" and "No thanks." The dialog was only shown once. Users could also enable or disable the data collection at any later point in time in the app's settings. Declining the anonymous data collection did not negatively impact the main functionality of the app in any way. If the user consented to the data collection, the app would periodically snapshot all pending notifications in the notification drawer. We used the *Android-Job* library [35] to schedule the sampling. The library abstracts from version differences in the Android SDK. We set the sampling job to be executed every 15 minutes, which is the minimum amount of time between two jobs. In later versions of Android, these jobs might be deferred if the device uses battery-saving features such as the *Doze Mode*, which defers background processes if the device was not used and not moved for a certain amount of time. Each snapshot contained the following features:

- A randomly generated unique ID for the device (UUID) to associate multiple snapshots with a specific device.

- The current Android version, device model, product name, and device manufacturer.
- The current timestamp and timezone.
- Metadata of all notifications in the notification drawer, such as the package name, timestamp of creation and position in the drawer.

Snapshots generated by the app were limited to metadata and did not contain text or images. The snapshots were stored in a separate local *SQLite* database.

3.1.2.3 Procedure

We published the study app on the Google Play Store. Users from all over the world were able to download it for free. We did not advertise the app in any way. Instead, users found the app using the Google Play Store search or by reading articles and watching videos that reported on the app. If a user decided to opt into the data collection, the locally stored snapshots were periodically sent to a server hosted at the University of Stuttgart using a secure connection. To avoid negatively impacting the device, the app only sent data over Wi-Fi and if the battery was not low. If the server did not acknowledge the data, the app would re-try sending the data.

3.1.2.4 Data Filtering

We defined a set of filter rules on the collected snapshots and excluded all devices that did not match the rules:

1. The time delta between the first and last snapshot is at least one week (7 days).
2. There are at least 672 snapshots for the device. This assumes a snapshot every 15 minutes (4 per hour), for each hour of the day (24), for each day of a week (7).
3. The maximum time delta between two snapshots is less than 48 hours. Larger deltas might happen if a device is turned off for extended periods.

4. No snapshot is missing. On the device, each snapshot is assigned an ascending ID. This ID is sent to the server along with the snapshot, which allows us to identify missing snapshots.
5. No snapshot has invalid timestamps, i.e., the timestamp associated with a snapshot is within the data collection period. Invalid timestamps might happen because of malfunctions of the clock of the device, failed synchronizations with timeservers or incorrect time/date set by the user.
6. At least one snapshot contains at least one notification.

This set of filter rules ensures a valid and consistent data set. It is robust against typical problems of in-the-wild data collection, such as unknown hardware and unstable network connections. In addition to the filter rules, we excluded all snapshots from *Huawei* devices. Our testing showed that many *Huawei* devices have an aggressive battery-saving feature that interfered with the notification logging.

3.1.3 Results

After filtering the collected data, we ended up with $8,830,112$ notification drawer snapshots from $3,953$ devices.

3.1.3.1 Demographic Background

While we did not ask the users about their demographic background directly, we can infer some information from the devices. We found that the language of the devices was set to Turkish most of the time (57.22%), followed by English (17.76%), Spanish (9.89%), and German (5.34%). Overall, we saw 31 different languages (grouped language variants). We also looked at the time zones configured on the devices as reported by the Android system (e.g., "Europe/Istanbul"). Most devices were set to a European timezone (64.99%), followed by Asia (18.54%), America (13.84%), Africa (1.82%), Australia (0.30%), and other (0.52%). In total, we saw 158 different time zone configurations, which shows the international user base of the app. In terms of devices, most devices were manufactured by Samsung (63.22%), followed by LG Electronics (5.59%), and

General Mobile (5.21%). We saw 74 different manufacturers in total. Compared to the global average [6], the devices used more recent versions of the Android operating system. The Android versions used were Android 5.x (7.89%), Android 6.0 (21.35%), Android 7.x (23.13%), Android 8.x (44.06%), and Android 9.0 (3.57%).

3.1.3.2 Collected Snapshots

Overall, we collected snapshots for a minimum of 7 days and a maximum of 110 days ($Md = 20$ days). In this time frame, we collected between 673 and $14,257$ snapshots per device ($Md = 1,631$). We first investigated whether we managed to collect an even distribution of snapshots across the day. We collected $M = 93.14$ ($SD = 8.93$) snapshots per device for each hour of the day. The number of snapshots per hour decreases slightly at night. This was expected, as battery saving mechanisms in modern Android smartphones delayed the execution of our background process when the devices were idle. The average time delta between two subsequent snapshots was 17.74 minutes ($SD = 3.56$), which is close to our target of a snapshot every 15 minutes.

3.1.3.3 Number of Notifications in the Notification Drawer

Each notification drawer snapshot contains zero or more notifications. Counting all notifications of all snapshots revealed that we collected a total of $40,836,340$ notifications. The same notification may appear in multiple snapshots if it has not been dismissed by the user, notifying app, or Android system. Thus, we identified how many unique notifications we were able to capture. For each notification in each snapshot, we extracted the `PACKAGE_NAME`, `NOTIFICATION_ID`, `NOTIFICATION_TAG`, and `CREATION_TIMESTAMP`. The combination of these values allowed us to identify unique notifications across snapshots. We collected between 65 and $55,703$ ($Md = 1,514$) unique notifications for each device, with a total of $10,928,880$ unique notifications.

Snapshots without Notifications About one-fifth of all snapshots (20.53%) did not contain notifications. The other 79.47% snapshots contained between 1 and 160 notifications.

Average Number of Notifications (Total) Next, we calculated the average number of notifications per device. The left side of Figure 3.2 shows, that when considering all snapshots, we saw $M = 4.30$ ($SD = 5.86$) notifications in the notification drawer. The median number of notifications was 2.68. While most devices had less than five notifications on average, we also saw a number of outliers. 77.64% of devices had between $[0, 5)$ notifications on average, 14.87% between $[5, 10)$, and 7.49% more than 10 with a maximum of 70.53.

Average Number of Notifications (Grouped) The previously reported numbers represent the total number of notifications as reported by the Android system. However, in Android multiple notifications can be visually grouped, reducing the number of notifications actually shown to the user. For instance, Figure 3.1 shows a single *Google News* notification with two headlines and an indicator about two additional headlines. The notification can be expanded to allow the user to explore the four individual headlines, and to click or dismiss them individually. Internally, this single notification is represented as four individual notifications for the headlines and a summary notification to visually group them, totaling in five notifications. Many instant messaging and email apps make use of this feature to group conversations. To find out how many notifications are actually visually shown to users, we processed all snapshots and counted each notification group as one. Thus, the notification count in Figure 3.1 would be reduced from 8 to 4.

With this calculation in place, the average number of grouped notifications was 3.40 ($SD = 4.59$), with a median number of 2.17 (see right side of Figure 3.2). As Figure 3.3 shows, 84.59% of devices had between $[0, 5)$ grouped notifications on average, 10.35% between $[5, 10)$, and 5.06% more than 10 with a maximum of 61.28. For the remaining analysis, we report on the visually grouped notifications as this better reflects how users see notifications.

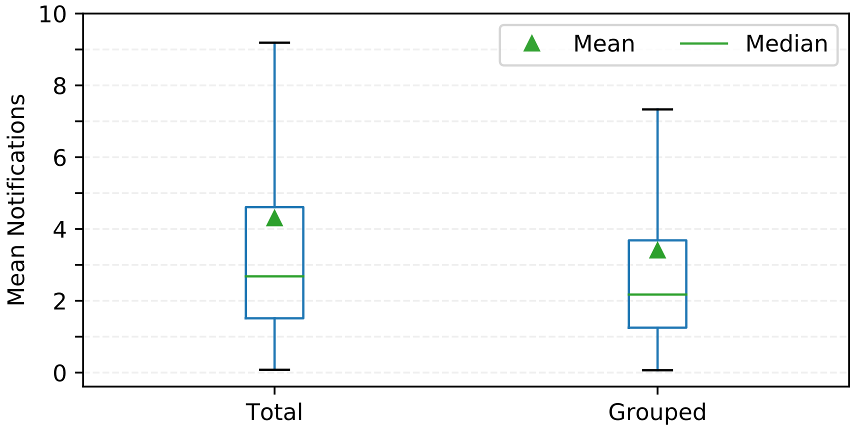


Figure 3.2: The mean number of notifications in the notification drawer over all snapshots. Total: All notifications, as reported by the Android system. Grouped: Visually grouped notifications. Outliers omitted.

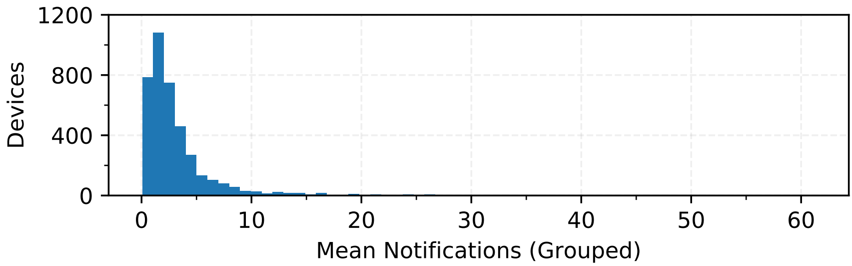


Figure 3.3: Histogram of the mean number of notifications (grouped) in the notification drawer per device. 62 devices had $>$20 notifications on average ($max = 61.28$).

Average Number of Notifications (Per Hour) Previous work has shown that the number of notifications users receive throughout the day drops significantly between midnight and 6am [121, 177]. Interestingly, as shown in Figure 3.4, the average number of notifications in the notification drawer *increases* in this time frame, with a peak at 6am. While users receive fewer notifications at night, they are also likely asleep and therefore do not dismiss notifications. Consequently,

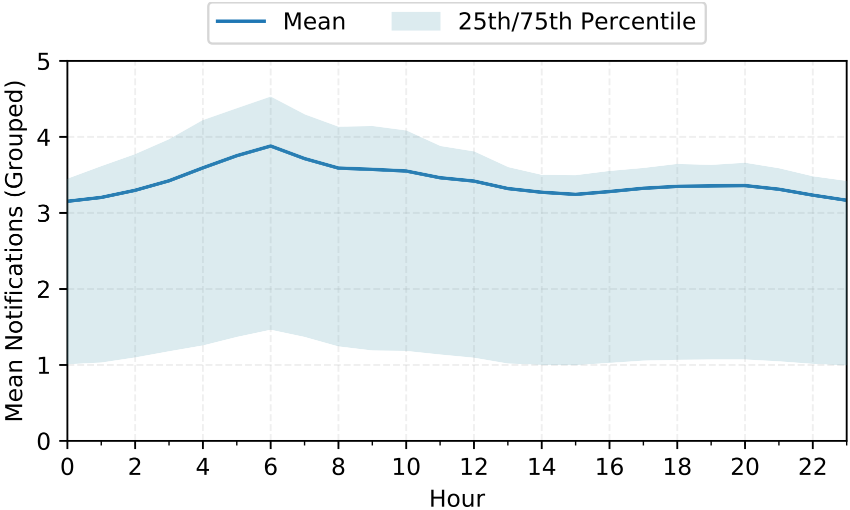


Figure 3.4: The mean number of visually grouped notifications in the notification drawer by the hour of the day. The number increases between midnight and 6am.

the number decreases again as users wake up and start attending the notifications. This shows an opportunity of assisting users in managing their notifications in the morning to ease the start in the day.

Average Number of Notifications (Per Weekday) Looking at the number of notifications in the notification drawer for each day of the week, we saw little differences, with only a slight drop on Sunday ($M = 3.31$) compared to the overall average ($M = 3.40$).

3.1.3.4 Number of Apps

Looking at the notifications in more detail, we saw between 4 and 111 ($Md = 28$) different apps per device that created at least one notification. In total, we saw $8,823$ different apps that triggered at least one notification. Only 24 apps were used on $\geq 1,000$ devices and 908 apps on ≥ 10 devices. A long tail of apps was used on < 10 devices, with over half of the apps (56.24%) only being used on one device.

Of the ten apps used on most devices, five were system apps including the *Google Play Store* ($3,416$ devices). The other five apps were the instant messaging app *WhatsApp* ($3,591$ devices), the social media network *Instagram* ($2,962$), the video-sharing app *YouTube* ($2,773$), the *Google Chrome* web browser ($2,603$), and the *Google Maps* app ($2,381$).

3.1.3.5 App Categorization

In line with prior work, we categorized the apps. We based the categories on the 12 categories used by Weber et al. [177], which in return is based on the work by Böhmer et al. [16] and Sahami Shirazi et al. [137]. Additionally, we introduced the category *Navigation* and extended the categories *Social* to *Social & Dating* and *News* to *News & Weather*.

We focused on the 908 apps with ≥ 10 devices and left the long tail of apps with fewer devices uncategorized. Still, with this number of apps, we were able to categorize 92.0% of the $10,928,880$ unique notifications in the data set. Similar to prior work, we automatically extracted the app category from the Google Play Store. It is important to note that the developers of the apps provide the categories on the Google Play Store. The categories might not necessarily reflect which kind of notifications an app creates. Further, 178 apps were not available on the Google Play Store, e.g., due to them being pre-installed by device manufacturers or by being manually installed by users. Two researchers independently went through the apps and manually categorized them. The categories provided by the Google Play Store were used as guidelines. For apps not available on the Google Play Store, the researchers searched the web for more information. Finally, the researchers compared the labeled categories and discussed conflicts until an agreement was reached. Table 3.1 shows the number of notifications and apps assigned to each category. The categories with the most notifications were *SMS & IM*, *System*, and *Tool*. The categories with most apps were *Tool*, *Shopping & Finance*, and *Media*.

Category	# Notifications	# Apps	Md Devices/App
Calendar & Reminder	170,284	23	43.0
Email	728,464	14	24.5
Game	102,164	111	19.0
Health & Fitness	137,404	24	17.5
Media	637,850	150	24.5
Navigation	235,631	24	18.5
News & Weather	118,797	48	18.0
Phone	454,586	18	133.0
Shopping & Finance	85,758	127	21.0
SMS & IM	3,692,077	45	42.0
Social & Dating	843,004	40	35.5
System	1,671,025	94	32.0
Tool	1,177,235	190	21.5
Uncategorized	874,601	7,915	1.0
$\sum$	10,928,880	8,823	-

Table 3.1: This table shows the number of unique notifications and apps per category, and the median number of devices per app for each category.

3.1.3.6 Notification Ranking

Based on the notification categories, we investigated what users typically see when they unlock their phones or open the notification drawer.

Background Notifications in the Android notification drawer are not simply displayed in chronological order. Instead, the Android system uses a number of signals to rank notifications. The used signals differ between Android versions and might be modified by device manufacturers. Some of the most prominent signals are as follows:

- The time when the notification was triggered and how much time has passed since then.
- The priority level which can be set by the notifying app. The priority level values range from *MIN*, *LOW*, *DEFAULT*, *HIGH*, to *MAX*. In Android 8.0

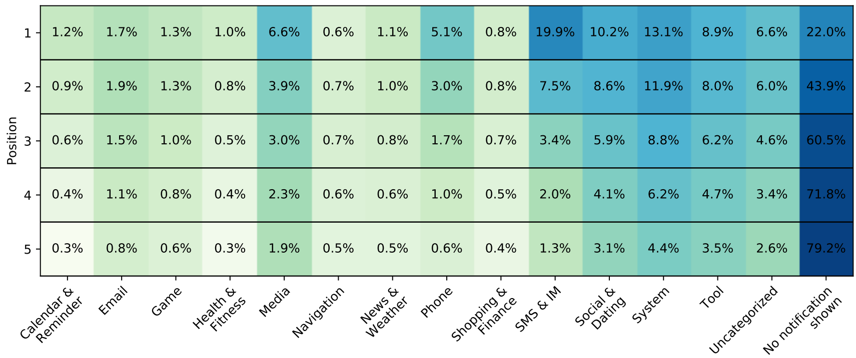


Figure 3.5: Distribution of which kinds of notifications are shown in the first five positions of the notification drawer.

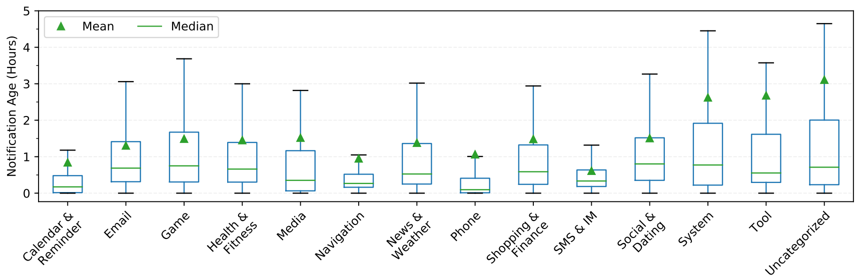


Figure 3.6: The age in hours of the notifications in the snapshots, normalized per device. Outliers omitted.

and newer, the priority level has been replaced by the importance level, which features the same values but allows users to overwrite them in the settings.

- Contacts associated with a notification and whether the contacts are marked as favorites by the user.

The idea behind the ranking is that the most relevant notifications for the user are shown at the top of the notification drawer. In 2014, Sahami Shirazi et al. found that *"notifications are for messaging"* and that *"important notifications are about people and events"* [137]. In 2018, Pielot et al. found that messaging notifications have a much higher conversion rate than notifications from other types [127]. This was also reflected in the Android 8.0 update released in 2017. The update introduced a “visual hierarchy” for notifications by first assigning notifications to one of four sections and then ranking the notifications within each section [8]. Notifications in the *Major Ongoing* section are about time-sensitive content. Examples include ongoing phone calls, navigation, timers, and media controls. The *People to People* section focuses on instant messaging notifications and notifications about missed calls. The *General* section contains most other notifications, including reminders and email notifications. Finally, the *By the Way* section includes non-urgent content, such as weather and traffic updates. On recent versions of Android, these notifications are visually muted by reducing them to a single line and graying them out.

Analysis Since almost half (47.63%) of the devices in the data set were running Android version 8.0 and newer, we expected the notification ranking to be influenced by these new sections. Indeed, Figure 3.5 shows the distribution of notification categories for the first five positions in the notification drawer. We limited the Figure to five positions, as this is typically the maximum amount of notifications a user sees on the lock screen or the notification drawer before having to scroll down. We see five dominant notification categories: *SMS & IM*, followed by *System*, *Social & Dating*, *Tool*, *Media*, and *Phone*. *Media* notifications are prominent in the first position, due to playback control notifications that end up in the *Major Ongoing* section. However, we also categorized many apps as media that likely do not show playback controls. *SMS & IM* and *Phone* are focused around the first three positions in the drawer. This is likely due to them being in the *People to People* section. *Social & Dating* notifications might be part of the *People to People* or *General* sections, resulting in a more even distribution across the position in the notification drawer. Finally, *System* and *Tool* notifications made up a large number of apps and notifications and were therefore presented across the first five positions as well.

Notification Priority Levels Figure 3.7 shows the priority level distribution of the notifications per category. A large number of *Navigation*, *System*, and *Tool*

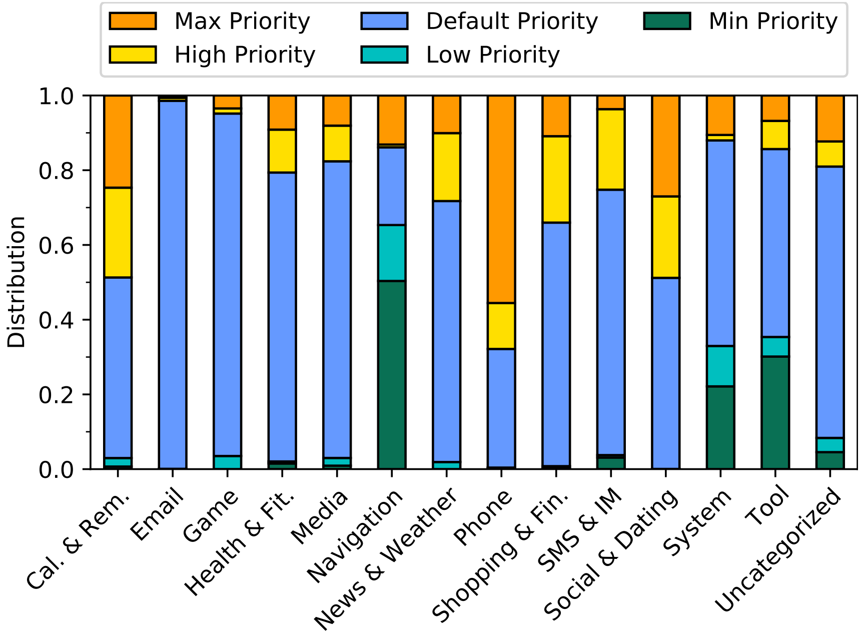


Figure 3.7: The notification priority level distribution per category. The priority level is set by the notifying app.

notifications were assigned the minimum priority level. Those notifications are displayed visually muted on recent Android versions. Notifications related to people and events (*Calendar & Reminder*, *Phone*, *SMS & IM*, *Social*) were assigned the high and maximum priority levels more often. More than half of the *Phone* notifications were assigned the maximum priority value. Interestingly, *SMS & IM* notifications were less often assigned the maximum priority level than *Social & Dating* notifications, however, due to *SMS & IM* being assigned in the *People to People* section, they are likely to be ranked higher. A notable exception of people and events related notifications are *Email* notifications, as almost all had the default priority value.

Notification Age Figure 3.6 shows the age of the notifications in the drawer when a snapshot was taken. We can see that *Calendar & Reminder*, *Navigation*, *Phone*, and *SMS & IM* notifications tend not to stick around as long as the other categories. This might be either because of users reacting faster on these categories of notifications or because the app is often updating the notification. While the nature of our data set does not allow us to know the reason, we know from prior work that users tend to attend messaging notifications faster and more often [127, 137].

Non-clearable Notifications Another reason for some notification sticking around longer than others is that Android notifications can be marked as non-clearable. These notifications cannot be dismissed by users, even when clicking on *Clear All*. Half of the snapshots (51.39%) contained at least one non-clearable notification. We saw a median of one non-clearable notification per snapshot. Of the $10,928,880$ unique notifications, 71.27% were clearable and 28.73% were non-clearable. Most of the non-clearable notifications were from the category *System* (35.15%), followed by *Tool* (16.13%), *Media* (10.77%), and *Phone* (9.18%). Typical examples are active media playback notifications and notifications about ongoing phone calls.

3.1.3.7 User Types

So far, we mainly looked at the data set in an aggregated manner. However, our earlier results on the mean number of notifications in the notification drawer indicated differences in how users manage notifications. To explore this further, we turned to prior work by Whittaker and Sidner, who investigated the management of email inboxes and found three user types [189]. *Frequent Filers* constantly tried to reduce the number of items in their inbox, *Spring Cleaners* made "clean-up" passes in larger intervals of time, and *No Filers* did not make use of filing emails and relied on search instead. Inspired by these user types, we clustered the snapshots according to the mean number of notifications, i.e., $[0,5)$, $[5,10)$, and $10+$ mean notifications. Within those clusters, we found similar usage patterns regarding the notifications in the drawer over time.

Frequent Cleaner Figure 3.8a shows snapshots from 36 days of usage with 24 apps and $M = 0.48$ notifications ($SD = 0.62$, $Md = 0$). 37.67% of the snapshots

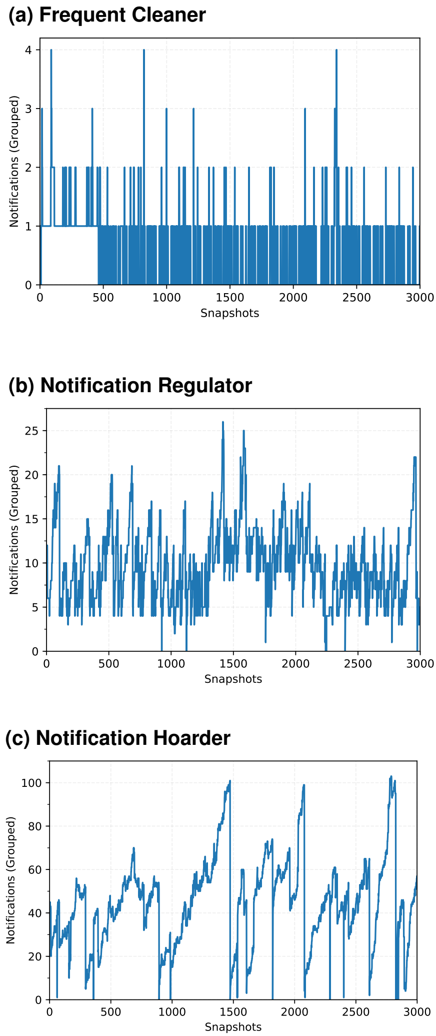


Figure 3.8: Examples for the three user types. *Frequent Cleaners* try to keep notifications out of the notification drawer. *Notification Regulators* have an increased number of notifications in the drawer but keep the overall number in check. *Notification Hoarders* accumulate notifications and dismiss them all at once, by pressing *Clear All* or restarting the device.

contained a non-clearable notification. Similar to *Frequent Filers*, *Frequent Cleaners* try to minimize the number of notifications in the drawer. Even with one-third of the snapshots containing a non-clearable notification, the median number of notifications is zero. This also reminds of the "inbox zero" email management approach of keeping the inbox empty [87].

Notification Regulator Figure 3.8b shows snapshots covering 39 days and 23 apps with $M = 9.69$ notifications ($SD = 4.19$, $Md = 9$). 99.73% of the snapshots contained a non-clearable notification. *Notification Regulators* have a higher number of notifications in the notification drawer, but they take action before the number gets too high.

Notification Hoarder Figure 3.8c shows snapshots covering 34 days of usage with 97 apps and $M = 45.0$ notifications ($SD = 20.01$, $Md = 45$). 99.7% of the snapshots contained a non-clearable notification. This type of user does not seem to dismiss notifications regularly. Instead, they let notifications accumulate and presumably only take action on the notifications that are important to them. In the shown time frame, we can see the number of (grouped) notifications reaching 100 multiple times. We can also see multiple drops were all notifications were cleared, presumably from the user pressing *Clear All* or restarting the device. However, right after the drop, the number of notifications starts to accumulate again. In a recently published work on the importance of notification content, Visuri et al. were surprised by a participant of a pilot study not clearing their notifications [156], a characteristic of *Notification Hoarders*.

Most users we have seen in our data set can be categorized as a *Frequent Cleaner* or *Notification Regulator*. While the number of *Notification Hoarders* is rather small, this behavior seems to be alarming from a notification overload perspective. Grevet et al. suggested a link between high email unread counts and feelings of disorganization [52], something that future work should investigate for notifications.

3.1.3.8 Summary

We conducted a large-scale observational study to gain an understanding of notification drawers in-the-wild. By periodically sampling almost four thousand devices, we showed the average number and the positioning of notifications in notification drawers, and the existence of different user types.

3.1.4 Discussion

Prior work has mostly focused on the *arrival* of notifications, e.g., by developing models for automatically deferring notifications until breakpoints [43, 44, 109]. While this is an important aspect of notification management, it is not the whole story. Even when notifications are deferred, they eventually end up in the notification drawer. The same is true for "silent" notifications that do not trigger vibrotactile or sound feedback or when the user silenced the device. In the end, the user is presented with an ever-filling list of notifications on the lock screen and notification drawer. This list has somehow to be managed; otherwise, the advantages of providing proactive information are lost.

Notification Management Ranking the notifications not in chronological order but based on signals already helps the notification drawer management on Android. In recent Android versions, *Major Ongoing* notifications that often require user interaction (ongoing phone calls, media controls) have a secured spot at the top of the list [8]. Messaging notifications, that were shown again and again to be the most important kind of notifications, are hoisted to the top as well. Still, we argue that this can be improved further. Recently we saw first work towards improving the interaction in the notification drawer. Pielot et al. investigated the dismissal behavior of users [127], and Weber et al. explored new interactions by enabling users to snooze notifications, i.e., temporarily removing and re-triggering them from the notification drawer [177].

Notification Middleware We see many parallels between notification drawers and email inboxes. Users receive many different kinds of emails and notifications, e.g., personal messaging, reminders, promotions, and spam. However, while it is common to filter, label, and categorize emails, notification controls are currently mostly limited to muting and disabling specific apps. Nowadays, using email without a spam filtering middleware is uncommon, and we argue that there is a need for a similar middleware for notifications.

User Types and Digital Well-being Our data set revealed that most users seem to be able to keep their notifications in check. Most users had between zero and ten pending notifications in the notification drawer. We suggested the two user types *Frequent Cleaners*, who try to keep the notification drawer clean, and *Notification Regulators*, who have an increased number of notifications in the drawer but overall keep them in check. However, we also saw a small set of users "hoarding" notifications. On first sight, it seems like these users have given up managing their notifications. The implications of these user types are not yet known. Future work should investigate different notification management strategies and their effects on the users' digital well-being. Possible research questions are whether *Notification Hoarders* are feeling more overwhelmed or feel like they are missing more information than the other user types. The opposite could also be hypothesized. Since those users are spending less time managing notifications, they could feel less stressed than the other user types.

The Importance of Messaging Finally, as shown again and again in prior work, we saw the importance of messaging in the data set. By far most of the unique notifications were of the category *SMS & IM*, they were prominently positioned in the notification drawer, and were quickly attended to, implying a high turnover rate. However, other categories should not be neglected. For future work, we suggest exploring new tools for managing notifications. Notifications could be automatically cleared after a particular time has passed or based on a context change, e.g., for location-based notifications.

3.1.5 Limitations

In this work, we focused on Android devices since prior work on mobile notifications primarily used Android devices as well [33, 93, 109, 121, 122, 127, 137]. Future work should also consider the other current dominant smartphone operating system, iOS. While the notification drawer on iOS is similar to Android, with notifications shown on the lock screen and by swiping down from the top of the screen, the two operating systems differ in important ways. For instance, notifications in iOS are opt-in and opt-out on Android [14, 187]. iOS also makes heavy use of notification badges on app icons that allow apps to gain the user's attention more subtly without posting a notification in the notification drawer.

A second limitation is that we did not record the user interaction in-between snapshots. Therefore, users who receive few notifications and users who receive many notifications but act upon them quickly likely have similar characteristics in this data set. Future work should consider this as well.

3.1.6 Open-Source Data Set

We published the data set and Jupyter notebooks for analysis on our project page[1] under the MIT license. We are confident that this will allow the community to further explore the data set and foster future research on mobile notifications.

3.2 Annotating Mobile Notifications

While the study in the first part of this chapter provided insights on how notifications materialize on smartphones, a deeper understanding of mobile notifications is still missing. For instance, while previous work found that "notifications are for messaging" [137], communication is a broad category which requires closer inspection. In a recently published study, researchers found that participants attend notifications about individual (1:1) chats faster than group chats [127]. Further, messaging is no longer limited to text messages alone. Modern messaging apps support rich media formats such as pictures, videos, and voice recordings. In July 2017, the popular messaging app *WhatsApp* reported 1 billion daily active users sending 55 billion messages, 4.5 billion photos, and 1 billion videos per day [188]. It is not yet known how these types of messages, and corresponding notifications, are perceived by users.

[1] `https://github.com/interactionlab/android-notification-drawers`

Learning how users perceive notifications is a challenging task. The importance and urgency of notifications depend on their content and context [94], which is the reason why notifications are typically assessed in-situ. In the case of communication-related notifications, the relationship between the sender and the user matters as well [95]. One approach to learn about users' experiences throughout the day is triggering questionnaires at random times or using specific events. However, these questionnaires are delivered to users as notifications themselves, which makes it difficult to survey users without influencing them by introducing additional interruptions. Further, the nature of notifications is inherently private and often sensitive. While the content of notifications is essential to understand the perceived importance and urgency, handling the content in user studies must not be an afterthought.

In the following, we introduce *Annotif*, a system for annotating mobile notifications in user studies. Using this system, we conducted a week-long in-situ case study to explore the importance and urgency of mobile notifications. *Annotif* enabled participants to annotate their notifications including the content and context while respecting the notifications' private nature. The results show that participants perceived 38.91% of their notifications as not important and over half (51.75%) as non-urgent. Only 6.33% of the notifications were rated as both very important and very urgent. We discuss influencing factors, including a detailed breakdown of 1:1 and group messaging notifications.

3.2.1 Experience Sampling of Mobile Notifications

The *Experience Sampling Method* (ESM) is a popular method to learn about participants' experience throughout the day by triggering surveys at random times or triggered by specific events [25, 55, 151]. However, these surveys may disrupt participants [50], which is problematic when studying interruptions caused by notifications in the first place. Sahami Shirazi et al. triggered surveys on participants' desktop computer to assess the importance of mobile notifications [137]. However, the surveys were limited to few notifications per participant and only surveyed about the app that created the notification, without considering the content or context. The researchers balanced the limited samples by having a user base of over 40,000 users. However, for most user studies, especially in an academic context, this number of users is unfeasible.

An alternative to ESM is the *Day Reconstruction Method* (DRM) [69]. Instead of surveying participants throughout the day, they are asked to reconstruct the day systematically before assessing it. This method does not capture participants' impressions in the exact moment. Instead, the assessment is done post-hoc. However, by asking participants to reconstruct the day this limitation is reduced, while having the advantage of not disrupting the participants during the day.

To summarize, capturing how users perceive notifications without influencing them is a challenging task. While prior work has shown the importance of the notification content, handling the content in user studies is challenging due to its private nature. What is missing are tools that enable us to gain a deeper understanding of mobile notifications in an unobtrusive and privacy-respecting manner.

3.2.2 The "Annotif" Annotation System

To overcome the challenge of surveying users about mobile notifications, we developed an annotation system consisting of an Android notification logging app, a server application, and a web-based annotation tool (see Figure 3.9).

3.2.2.1 Background on Notifications in Android

Notifications are a core feature of Android. Any Android app can trigger notifications by default. However, users can disable notifications for specific apps. A notification typically consists of a small icon and two lines of text. Notifications are shown in the notification drawer that can always be accessed by swiping down from the top of the screen [180]. In newer versions of Android, notifications are also shown on the lock screen. Notifications can be extended in several ways. Developers can attach sounds and vibration patterns, set priority levels, and group multiple notifications.

The *priority* level is one factor that decides the order of notifications in the notification drawer. Possible priority levels range from `MIN`, `LOW`, `DEFAULT`, `HIGH`, to `MAX`. The notifying app sets the priority level.

Notification *groups* consist of multiple notifications from the same app that share the same *group key*. Apps can set one notification as the *group summary*. For example, consider an instant messaging app that creates a notification for each unread conversation. The app would create N notifications for the N unread conversations, and an additional notification as the group summary. Depending on the Android version, the Android system would only display the group summary while hiding the other N notifications, or allow the group summary to be expanded. When it comes to updating existing notifications, apps may use different strategies. In the example with $N+1$ instant messaging notifications, an additional conversation could cause the app to trigger a single notification for the new conversation and update the group summary, resulting in two notification events. An alternative strategy is to revoke and re-create all notifications, resulting in $N+2$ notification events. It is important to keep this behavior in mind, as notification events in the Android system do not directly correspond to the actual number of different notifications shown to users.

3.2.2.2 Notification Logging App

We developed an Android app to log notifications from smartphones, extend them with context data, encrypt them, and periodically sync them with a server. The app registers itself as a *Notification Listener Service* [7]. The service retrieves events about new and removed notifications from all apps installed on the device. A useful aspect of this service is that it is exempt from battery optimization procedures and, therefore, always runs in the background without interruption.

The left side of Figure 3.9 shows the data flow of the Android app. The *Notification Listener Service* listens for new notifications in the background. Once a new notification event is received, the app first extracts all metadata of the notification. This includes the *package name* (the identifier of the app that created the notification), the time when the notification was created, the app-set priority level, and the notification group key. In line with prior work on smartphone users' concerns [42] and to respect the private nature of notifications, the app extracts

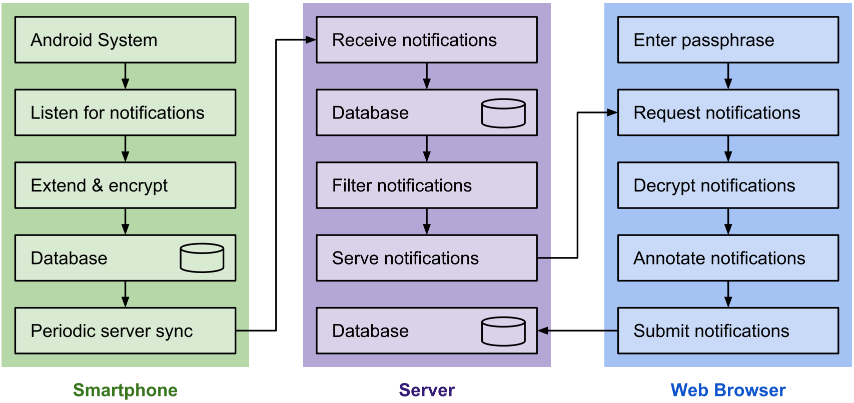


Figure 3.9: The data flow of the system. Notifications are collected on the smartphone and periodically synced to the server. On the server, notifications are filtered and served to the web browser. The notifications are then annotated by the user and sent back to the server. Notification content is encrypted between the smartphone and the web browser.

the notification content and encrypts it using the *Advanced Encryption Standard* (AES) in the *Cipher Block Chaining* (CBC) mode with an encryption key derived from a user-defined passphrase using `PBKDF2`. The app also stores the `SHA-256` hash of the content. This allows the detection of notifications with duplicate content without knowledge of the content itself. The app will also automatically record if the content contains the name of the user or the name of a user's contact. This is done by retrieving the list of saved contacts on the device and searching for the contact names in the content text using a regular expression. Finally, the app associates the device's current location with the notification. The extended and encrypted notifications are then stored in an on-device database and periodically sent in batches to the server using a secure connection.

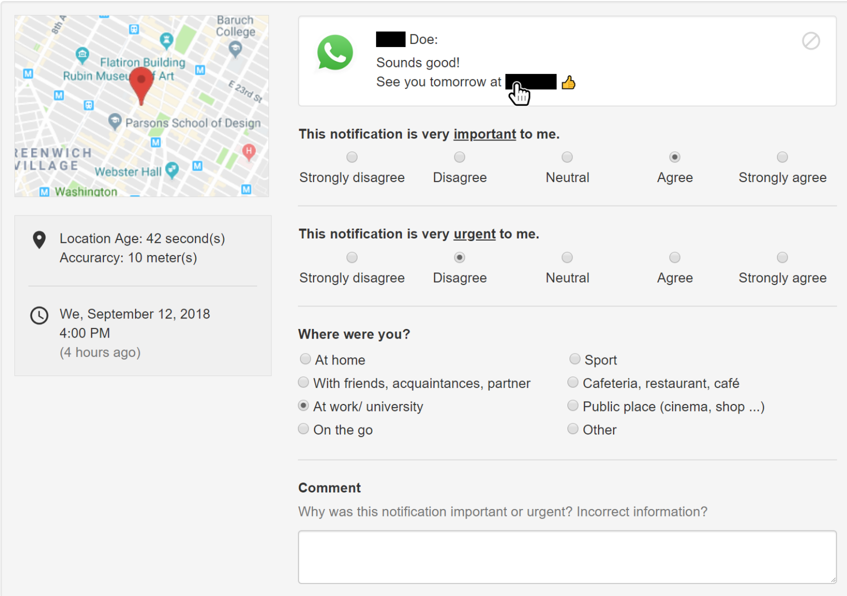


Figure 3.10: Screenshot of the web-based annotation tool for a single notification. The tool provides a map, the absolute and relative time, the app icon of the notifying app, the notification content text, importance/urgency ratings on a 5-point scale, a location selection, and an optional comment field. Clicking on words censors them. Users can optionally censor the entire notification content by clicking on the block icon in the top right corner.

3.2.2.3 Server

The server receives notifications from the Android app (see Figure 3.9). It stores the encrypted notifications in a database and associates them with the user ID. To prepare the notifications for annotation, the server executes the following five filtering steps:

1. For a given user ID, select all notifications that were not yet annotated by the user.

2. Discard all notifications that are older than a specific time period (e.g., 48 hours). This ensures that users only annotate notifications that they still remember.
3. Cluster the remaining notifications according to the app that created them. Split the clusters if they contain notifications from multiple days to one cluster per day. Sort the clusters from old to new.
4. For each cluster, retrieve the group key of all notifications. For each group key in each cluster, check if the group contains both a group summary and other notifications. If this is true, filter the group summary and keep the other notifications. Otherwise, if there is only a group summary, keep it.
5. For all notifications in each cluster, compare the text hash value. If there are multiple notifications from the same app with the same text hash, keep the first instance of the notification and discard the duplicates.

The remaining clusters do not contain notifications from the same app with duplicate content. The clusters are served one-by-one to the web-based annotation tool. For instance, the user would see a list of *WhatsApp* notifications that were created approximately at the same time. After they are annotated, the server stores them in a separate database table. The server would then serve the next cluster of notifications, e.g., a set of email notifications. Users can take a break from annotating notifications at any point in time.

3.2.2.4 Annotation Tool

Users access the web-based annotation tool using a personalized link that contains the user ID. The user is then shown a password field to enter the same passphrase that was set in the Android app. The passphrase is kept client-side and never sent to the server. The annotation tool then requests a new set of notifications from the server, decrypts them using the user-provided passphrase, and renders them (see Figure 3.10). The annotation box consists of the following parts:

Location A map showing the location of the device when the notification was triggered. It also shows the estimated location accuracy and age of the location data.

Date and Time The date and time when the notification was triggered, including a relative description to the current time (e.g., “4 hours ago”).

Notification The content of the notification. Clicking on a word censors it. The “block” icon in the top right corner censors all text at once. Additionally, the icon of the app that created the notification is shown next to the text.

Annotation Form Controls We implemented 5-point Likert scale items to rate the agreement to the statements that the notification is very important/ very urgent. Further, the location can be assigned to one of eight pre-defined labels, and an optional free text field allows users to provide additional information or to report problems.

After the user annotated all notifications in a cluster, the tool verifies that all required form controls were selected and sends the annotated notifications to the server using a secure connection. At this point, the content is no longer encrypted and can be used for analysis. However, users are in control about what is being shared by censoring parts of or the entire content. We want to highlight that the system can be easily modified or extended by logging additional values in the Android app or by replacing the form controls shown in the annotation tool. Notifications are stored in the *JavaScript Object Notation* (JSON) and are extended as they flow through the system. Therefore, the system can be used flexibly in different kinds of user studies.

3.2.3 Case Study

To test the *Annotif* system, we conducted a week-long in-situ study. Participants installed the Android logging app on their personal smartphones and annotated notifications on their personal laptops or desktop PCs.

3.2.3.1 Design

We designed the case study inspired by prior work [137], the *Experience Sampling Method* (ESM), and the *Day Reconstruction Method* (DRM). The Android app on participants' personal smartphones recorded all notifications throughout the day and periodically synced them with the server. The notifications were extended with additional context data, such as the location of the device when the notification was triggered. This allowed the participants to reflect on the context when the notification was received. Participants annotated the notifications on their personal laptops or desktop PCs. They were free to annotate them whenever they had time. However, we set a time limit of 48 hours to ensure that participants could still remember the context.

3.2.3.2 Procedure

We invited the participants to our lab one-by-one and explained the study procedure. We explicitly stated that the participation is voluntary and that they can end their participation at any time. After the introduction, participants signed a consent form and filled in a demographic survey. We then installed the notification logging app on participants' smartphones. On all devices, we verified that the date and time were set correctly, that there was sufficient free storage available, and that location services were enabled. Participants then entered their first, last, and nicknames in the app. The app also accessed the names of the participants' contacts to automatically detect if a notification contained the name of a contact. After the participants set their secret passphrase for the text encryption, the notification logging app was silently running in the background of the smartphones. No further intervention was necessary.

We then showed the participants the annotation tool and explained all aspects of it. After the participants left, we sent out personalized emails with links to the annotation tool. Participants then annotated notifications for one week. Afterward, we sent out another email with instructions on how to uninstall the app and a post-study questionnaire.

#	Total events	Filtered	Missed	Annotated
P1	792	539	0	253
P2	1,219	816	31	372
P3	1,630	955	0	675
P4	2,327	1,518	86	723
P5	810	427	99	284
P6	738	458	0	280
P7	1,321	721	0	600
P8	917	611	0	306
P9	594	335	0	259
P10	2,664	1,855	48	761
P11	1,240	745	9	486
P12	1,563	1,099	0	464
P13	2,856	1,940	191	725
$\sum$	18,671	12,019	464	6,188

Table 3.2: Total notification events, filtered events (duplicates/ groups), missed annotations, and annotated notifications.

3.2.3.3 Participants

We recruited participants from the local area. All participants were German. A requirement for the study was that participants own an Android-based smartphone with Android 5.0 or newer, and a laptop or desktop PC. Thirteen participants participated in the study (7 female, 6 male). They were between 21 and 55 years old (M=26.23; SD=8.44). Four participants were employees, and nine were students. Ten of the thirteen participants stated to use their smartphones for both personal and work purposes. The other three participants only used their smartphone for personal purposes.

3.2.4 Results

All thirteen participants completed the study and annotated their notifications for one week.

3.2.4.1 Devices

Participants used their personal smartphones for the study. The devices had between 151 and 391 apps installed (M=266; SD=92). This number includes system apps and apps that were pre-installed by the device manufacturer. The language of all smartphones was set to German.

3.2.4.2 Received and Annotated Notifications

We logged a total of 18,671 notification events. Breaking this number down per participant, this results in between 594 and 2,856 notification events per participant (M=1,436; SD=750; Md=1,240). As shown in Table 3.2, the server filtered 12,019 duplicate notification events and group summaries. 464 notifications were not annotated due to six participants sometimes missing the 48 hours annotation time window. This resulted in a total of 6,188 – or 93.02% – annotated unique notifications. Again, breaking this number down per participant, we saw that participants annotated between 253 and 761 notifications (M=476.00; SD=198.17; Md=464). The annotated notifications were created by 94 different apps. The instant messaging app *WhatsApp* was used by all participants and dominated the number of created and annotated notifications. 65.22% of the annotated notifications were created by *WhatsApp*, followed by the *Google Play Store* (4.99%), and the instant messaging app *Telegram X* (3.80%). Overall, we saw a median of seven notifications annotated per notifying app.

3.2.4.3 Timings

Most annotated notifications were triggered around noon and in the afternoon. 25.26% were triggered between 10am and 2pm and 29.14% between 6pm and 9pm. The median time between a notification being triggered and finally being annotated was 17 hours and 53 minutes (see Figure 3.11). Two-thirds (72.66%) of the notifications were annotated within 24 hours. Participants annotated notifications throughout the day, with a third (32.97%) of the notifications being annotated between 5pm and 7pm.

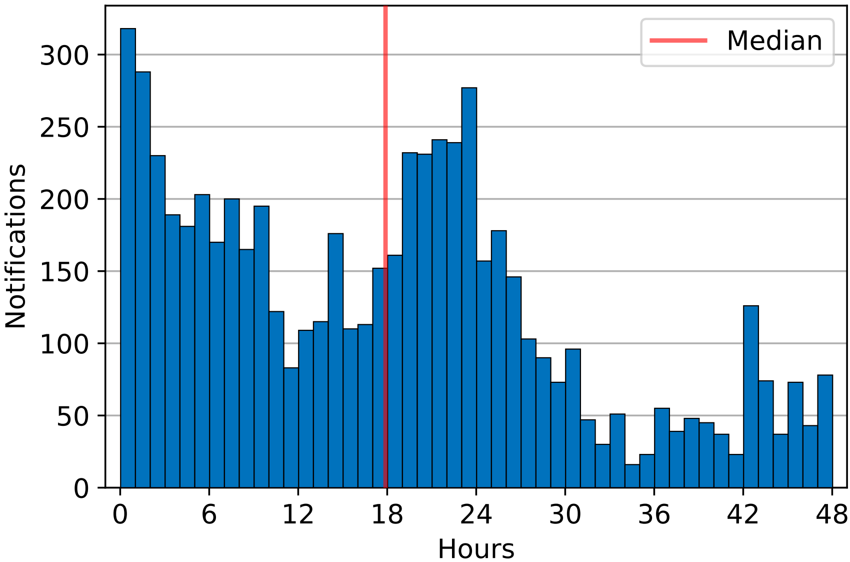


Figure 3.11: Histogram of the time delta between notifications being created on the participants' smartphones and finally being annotated.

3.2.4.4 Annotated Locations

Most of the notifications were annotated with the *Home* label (57.19%), followed by *Work* (14.87%), *On-the-go* (12.78%), *With friends* (9.2%), at a *Restaurant/cafe* (2.81%), *In public* (1.66%), and during *Sport* (1.12%). Only 0.37% of the notifications were annotated with the catch-all *Other* label. This indicates that the map shown next to the notifications in the annotation tool supported participants in assigning the notifications to a context.

3.2.4.5 Optional Comments

Eleven participants used the optional comment field to provide additional information for 4.12% of the annotated notifications. The comments mostly provided more details about the location (such as multiple labels applying) or mentioned that the location was off. This further indicates that participants were able to recall the context for a given notification.

3.2.4.6 Censored Content

Participants made use of the option to censor parts of the content for 32.01% of the annotated notifications. Only 0.82% of the notifications were censored completely. With 92.83%, most censored notifications were *WhatsApp* notifications.

We calculated how many notifications were censored per app. The five apps with the highest percentage of censored notifications were all communication apps: *WhatsApp* (45.56%), *Snapchat* (40.91%), *SMS* (37.50%), *Gmail* (34.55%), and *Facebook* (26.92%). Looking at which parts of the text were censored, we noticed that participants made use of the option to remove the names of their contacts in personal messages, with the message itself often left uncensored. This was interesting, as censoring the name and the entire message requires the same number of clicks in the annotation tool. Participants seemed comfortable with sharing the messages as long as the senders' names were censored. This is an useful insight for future studies on mobile notifications.

3.2.4.7 Importance and Urgency Ratings

The distribution of the importance and urgency ratings can be seen in Figure 3.12. Looking at the agreements to the statement that a notification is *very important*, we found that participants (strongly) disagreed in 38.91% of the cases. 17.42% were rated neutral, and in 43.67% of the cases participants (strongly) agreed. Regarding the statement that a notification is *very urgent*, participants (strongly) disagreed in over half of the cases (51.75%). 22.6% were rated neutral, and only 25.65% of the annotated notifications received (strong) agreement ratings.

For comparison, we looked at the *priority* value that is set by apps for each notification. The five priority levels (`MIN`, `LOW`, `DEFAULT`, `HIGH`, `MAX`) can be compared to the 5-point Likert scale items used for the importance and urgency ratings. We found that for most notifications (87.69%) the `DEFAULT` value was set. Thus, the priority level is not useful to decide on the actual importance or urgency of notifications.

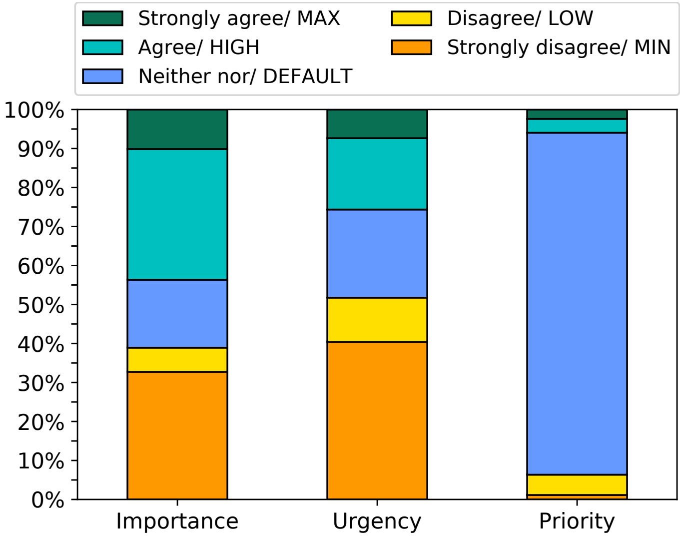


Figure 3.12: Distributions of the agreements that notifications are very important/ very urgent. The Android priority level set by apps for comparison.

3.2.4.8 Correlations

We calculated the Pearson correlation coefficient for the *importance* and *urgency* ratings, and the *priority* value. The results show that there is a strong positive correlation between the importance and the urgency of notifications (r=0.82; p<0.001). The correlation is visualized in Figure 3.13. Notably, participants perceived one-third of the notifications (32.60%) as neither important nor urgent (importance = urgency = 1). Only 6.33% of the notifications were regarded as both important and urgent (importance = urgency = 5).

There is neither a correlation between the rated importance and the priority (r=0.18; p<0.01) nor between the rated urgency and the priority (r=0.16; p<0.01).

3.2.4.9 Messaging Notifications

Of the 6,188 annotated notifications, 65.22% were created by the instant messaging app *WhatsApp*. All participants used *WhatsApp*, which reflects the dominant

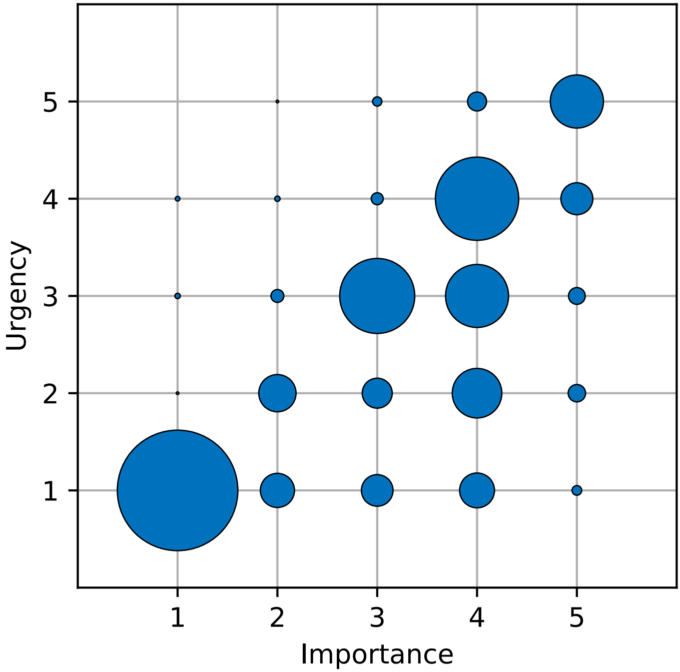


Figure 3.13: Correlation between the perceived importance and the perceived urgency of the annotated notifications. (1=strong disagreement; 5=strong agreement)

market share of the app in Germany. In the following, we provide a closer look at *WhatsApp* notifications according to four aspects. An overview of the ratings can be seen in Table 3.3.

(1) WhatsApp vs Other Apps On average, *WhatsApp* notifications received higher importance and urgency ratings compared to notifications from other apps.

(2) Rich Media Messages Apart from traditional text messages, *WhatsApp* allows users to send different rich media messages, including pictures, videos, and voice recordings. The notifications for these rich media messages contain corresponding emojis (📷, 🎥, 🎤) at specific positions of the text. This information allowed us to distinguish the notification types. The majority of notifications were for text messages (93.26%), followed by photos (4.44%), audio recordings (1.34%), and videos (0.97%). On average, notifications for audio recordings received the highest importance and urgency ratings, followed by text messages, photos, and videos.

(3) 1:1 vs Group Chats *WhatsApp* allows conversations between two users (1:1 chats) and multiple users at once (group chats). Prior work used text heuristics to differentiate between 1:1 and group chats [127]. However, looking at the metadata of *WhatsApp* notifications revealed that 1:1 chats are tagged with the string "`s.whatsapp.net`" and group chats with "`g.us`". This allowed us to reliably differentiate between them, regardless of the user's device language. We found that more *WhatsApp* notifications were from 1:1 chats (56.37%), compared to group chats (43.23%). Only sixteen notifications (0.40%) were without a tag. On average, 1:1 chats were rated as more important and urgent than group chats (see Figure 3.15a), likely because users are not always addressed directly in group chats.

(4) Mentioning the User The notification logging app automatically detected if notifications contain the first, last, or nickname of the users and flagged notifications accordingly. Notifications that contain the user's name received higher importance and urgency ratings than the other notifications. This is not only true for *WhatsApp* 1:1 and group chats, but also over all apps (see Figure 3.15b).

3.2.4.10 Notification Clusters

Participants rated notifications from 94 different apps. We selected all apps whose notifications were rated by at least three participants. We then calculated the normalized importance and urgency ratings for the resulting 18 apps. For *WhatsApp*, we included the normalized overall rating and added 1:1 and group chats as well. The resulting 20 data points can be seen in Figure 3.14. For the Figure, we

	Feature	Importance M	Importance SD	Urgency M	Urgency SD	N
(1)	WhatsApp	3.10	1.34	2.56	1.34	4,036
	Other apps	2.30	1.48	2.13	1.36	2,152
(2)	Voice	4.41	.66	4.00	1.20	54
	Text	3.10	1.34	2.57	1.33	3,764
	Photos	2.86	1.36	2.12	1.17	179
	Videos	2.41	1.16	1.57	.72	39
(3)	1:1 chats	3.47	1.14	2.86	1.30	2,275
	Group chats	2.60	1.42	2.16	1.28	1,745
(4)	1:1 with name	4.22	.88	3.39	1.35	89
	1:1 without name	3.46	1.14	2.84	1.30	2,186
	Group with name	3.80	1.21	3.09	1.29	35
	Group without name	2.57	1.41	2.14	1.27	1,710
	Both with name	4.10	.99	3.31	1.34	124
	Both without name	3.07	1.34	2.54	1.33	3,896

Table 3.3: The average importance and urgency ratings for *WhatsApp* notifications based on specific features.

categorized and color-coded the apps. Applying *k-means* with a value of $k = 4$ revealed the notification clusters C1-C4.

C1 *Incoming phone calls* received the highest ratings on average, closely followed by *alarms*. These types of notifications only contributed to 1.0% and 0.4% of the annotated notifications. It is easy to overlook these notifications when exploring the data set. However, they are of high importance and high urgency for the participants and often require their immediate attention.

C2 This cluster contains notifications about *missed phone calls*, calendar events (*Google Calendar*), *SMS* messages, and *WhatsApp* 1:1 messages. Notably, the *SMS* were not used for messaging. Instead, the notifications informed about

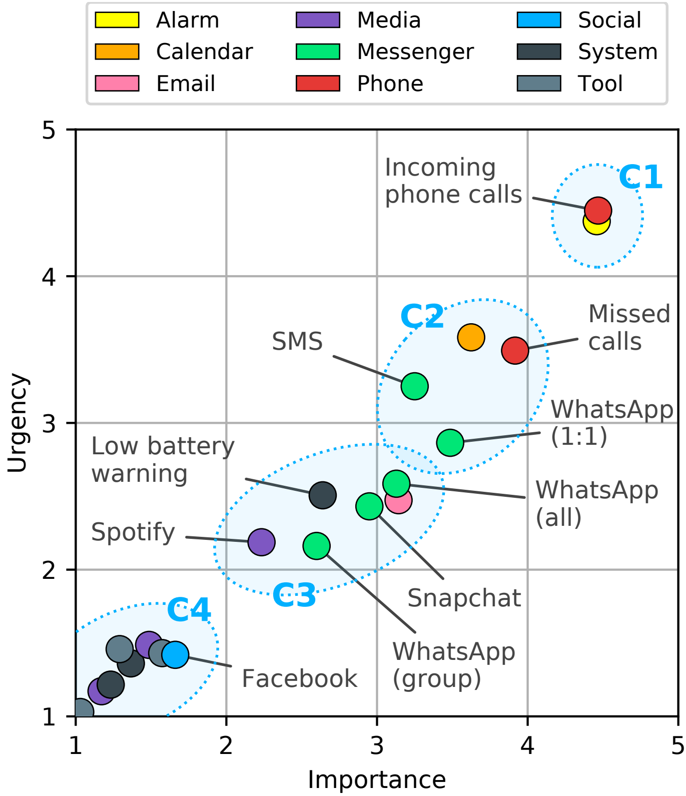


Figure 3.14: Normalized importance and urgency ratings of 18 apps that were rated by at least three participants. Additionally, we included *WhatsApp* 1:1 and group chats.

calls going to the mailbox and phone plan updates. Notifications of this type are relevant to the user because the user is addressed directly, but they do not necessarily require the user's immediate attention.

C3 This cluster contains notifications from *WhatsApp* (overall), email (*Google Gmail*), *Snapchat*, *low battery warnings*, *WhatsApp* group chats, and *Spotify* music. Notifications of this type might not always be relevant to the user, and they are even less time-sensitive.

C4 The remaining eight apps were of the categories social, system, tool, and media. Notifications of this type are "nice-to-have" but neither of importance nor urgency. In some cases, they could even be considered annoying by users.

We found that notifications containing the name of the user (see Figure3.15b) or the name of a contact (see Figure3.15c) can be an indicator that the notification is of higher importance. The urgency ratings are affected similarly. As we have shown, this can be detected automatically by using the contacts stored on the device.

3.2.4.11 Post-Study Questionnaire

After the participants annotated their notifications for a week, we sent out concluding emails. We thanked them for their participation and asked them to fill out a final post-study questionnaire. In the questionnaire, we asked them if they changed their smartphone usage behavior due to the participation in the study and if they were consistent in annotating their notifications. Eleven participants reported not changing their smartphone usage behavior during the study. One participant reported disabling specific notifications, and another participant mentioned uninstalling specific apps. Overall, participants were confident that they annotated their notifications consistently. Two participants mentioned sometimes having a hard time to rate the importance and urgency of notifications. Examples mentioned were notifications from music players about current songs and alarms. Participants also mentioned censoring the names of people to protect their privacy, something we were already able to see when looking at the annotated data.

In a final text field, participants informed us about their study experience. Annotating their notifications helped the participants to reflect on the notifications that they receive on a daily basis:

> *"It was interesting to see how many unimportant WhatsApp messages I receive throughout the day."* (P1)

Another participant realized during the study that she receives a large number of unimportant notifications that she subconsciously dismisses.

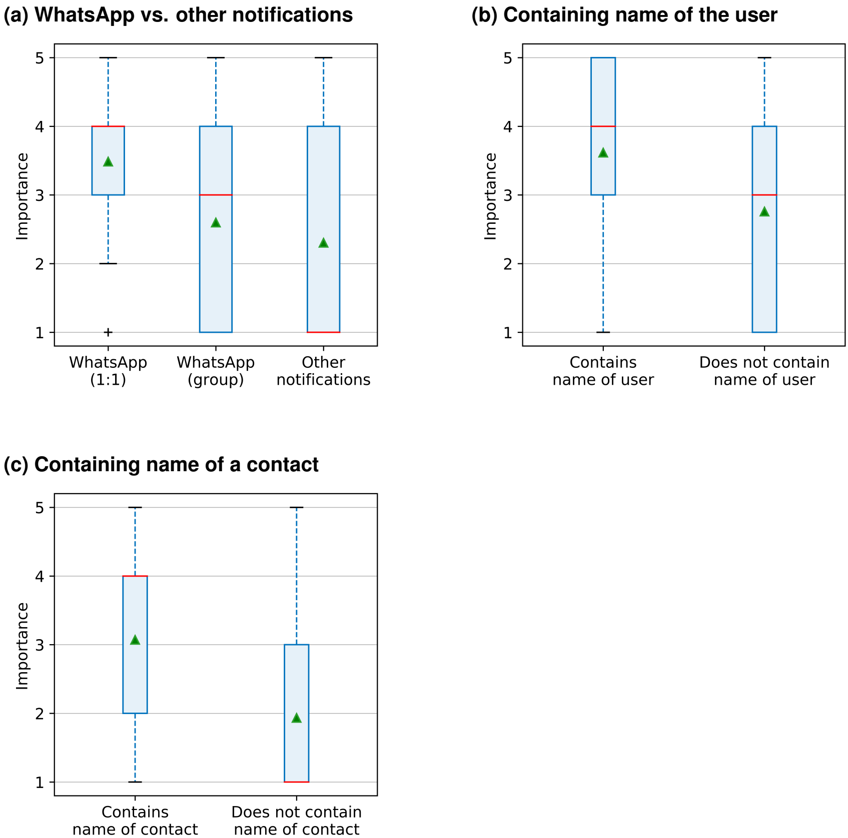


Figure 3.15: Differences in the importance ratings for the conditions (a) *WhatsApp* 1:1 and group notifications compared to notifications from other apps, (b) whether or not notifications contain the name of the user, and (c) whether or not notifications contain the name of a contact. (1=strong disagreement; 5=strong agreement)

> *"Only after annotating I noticed how many notifications I receive that are not relevant to me at all (e.g., weather notifications). Notifications like that I simply dismissed without consciously reading them. [...] I would like to turn off such notifications in the future since I probably still perceive them subconsciously."* (P7)

3.2.4.12 Summary

We conducted an in-situ case study in which participants annotated their notifications for one week. We used the *Annotif* annotation system that allowed users to annotate notifications without interrupting them while preserving the notifications' content and context. We found a strong correlation between the perceived importance and urgency of notifications. Only a small percentage of the notifications were regarded as very important and very urgent. The instant messaging app *WhatsApp* dominated the number of created and annotated notifications. We saw differences in the perceived importance and urgency depending on the type of notification (e.g., 1:1 vs. group and rich media messages). Notifications that contained the name of a contact or in which the user was addressed directly received higher importance and urgency ratings. Finally, we found four notification clusters that can be used to categorize notifications.

3.2.5 Discussion

Using the *Annotif* system, we were able to collect 6,188 annotated unique notifications from 13 participants. This equals an annotation coverage of 93.02% for non-duplicate and non-group notifications, without triggering surveys throughout the day and potentially creating further interruptions. The annotation interface displayed the notifications' content and context (location and time). The comments provided by the participants in the optional free-text field indicated that the participants were able to reflect on the notifications well. Participants were able to screen the text of all logged notifications before sending them to us for analysis. Participants made active use of this functionality. However, we were positively surprised that participants rarely censored all text. Instead, they focused on preserving their contacts' privacy by censoring names. This enables a more throughout analysis of mobile notifications than simply relying on metadata. Future user studies on mobile notifications might benefit from this finding. The *Annotif* system already detected contact names from the users' address books. This might be further extended in the future to automatically pre-censor notifications and, thus, reducing the number of interactions needed in the annotation tool.

The results of the importance and urgency ratings in the case study also pose interesting implications for smart notification management systems. As described in the *Notification Clusters* section, we found four notification clusters (C1-C4) in the data set. Critical notifications (C1) require the user's immediate attention. Examples include incoming phone calls and alarms. While critical notifications are of high importance and high urgency, they only contribute to a small fraction of the notifications users receive on a daily basis. Without a system that enables participants to assess *all* notifications, it is easy to overlook these notifications in larger data sets. On the other hand, we saw a large number of low priority notifications (C4). This is a long tail of apps that create unimportant and non-urgent notifications. Notifications of this kind may be considered nice-to-have or annoying by users.

In some cases, notifications may be lifted from one cluster to an adjacent cluster. We saw that mentioning the user or a contact increased the importance and urgency ratings. In reverse, notifications may drop to a lower priority level if they are received at the wrong time, e.g., personal notifications at work [177]. Finally, an app might display multiple types of notifications that are perceived differently by users. An example we saw in the study were notifications for rich media messages in *WhatsApp*, with voice recording notifications receiving higher ratings. This is a novel finding that was only uncovered by a notification data set with a high annotation coverage.

These clusters can aid designers of future smart notification management systems. We suggest that critical notifications (C1) should never be filtered or deferred by such systems. Low priority notifications (C4), however, can be deferred, shown in batches, or shown as summaries at the end of the day [10]. The biggest challenge for future smart notification management systems are high (C2) and medium (C3) priority notifications. These include communication-related notifications and are responsible for a large number of notifications users receive on a daily basis. Prior work has shown that there is a social pressure to respond as quickly as possible [23] and a fear of missing out [9]. Group chats might consist of highly relevant or completely irrelevant messages. Future work on messaging notifications can benefit from *Annotif*, as the system enables studying message contents in a privacy-respecting and unobtrusive manner.

3.2.6 Limitations

The *Annotif* annotation system focused on the text of the notifications. However, the results of the case study indicate differences in the perception of rich media notifications. In the future, the annotation system could be improved by re-creating the notifications visually more similar to notifications on the smartphone. This includes displaying images associated with the notifications, such as profile pictures of contacts.

Further, all participants in the case study were German. This was reflected in the apps used by the participants. The instant messaging app *WhatsApp* has a dominant market position in Germany. Other markets have different dominating messaging apps, e.g., *KakaoTalk* in South Korea and *WeChat* in China. Future studies should be conducted with a larger number and more diverse sets of participants over longer periods of time to create a more complete understanding of mobile notifications.

Finally, future work should evaluate *Annotif* on a meta level. This includes the workload of annotating notifications over extended periods of time and the overall usability of the system.

3.3 Conclusion

In the first part of this chapter, we complemented prior work by exploring the manifestation of mobile notifications in notification drawers and different strategies for managing these notifications (RQ1). We reported the results of a large-scale observational in-the-wild study, in which we sampled the contents of notification drawers. We collected $8,830,112$ notification drawer snapshots from $3,953$ devices. We systematically analyzed the data set and found users have, on average, 3.4 notifications in the notification drawer. Although users receive significantly fewer notifications at night, notifications accumulate overnight, resulting in more notifications for users to handle in the morning. We found that *SMS & IM* notifications dominate the number one position in the notification drawer and discussed reasons for this. Finally, we suggested the existence of three different types of users regarding the management of notification drawers. *Frequent Cleaners* aim to dismiss all pending notifications in the drawer quickly, *Notification Regulators* receive an increased number of notifications but keep them under control, and *Notification Hoarders* accumulate notifications in the drawer over time and dismiss them all at once.

In the second part of this chapter, we explored an approach for gaining deeper insights on notifications, specifically the actual content of notifications (RQ2). We introduced *Annotif*, a privacy-aware system for unobtrusively assessing mobile notifications in user studies. We reported the results of an in-situ case study in which participants annotated their notifications for one week. The results show that participants perceived 38.91% of their notifications as not important and over half (51.75%) as non-urgent. Only 6.33% of the notifications were rated as both very important and very urgent. We discussed influencing factors, including 1:1 and group messaging notifications, and implications for future smart notification management systems that continue to fulfill users' information need while respecting their digital well-being. Aside from allowing researchers to gain further insights by looking at the content of notifications, the annotation system helped participants to reflect on the notifications that they receive on a daily basis.

4

Managing Mobile Notifications

In the previous chapter, we explored how notifications accumulate in notification drawers and proposed the existence of different user types with regard to managing notifications. Further, we introduced a privacy-respecting annotation tool to investigate notifications in user studies. In a case study, participants mentioned that the annotation tool helped them to reflect on the notification that they receive on a daily basis.

Building on top of that, in this chapter, we explore the research question of how we can support users in managing their notifications (RQ3). In the first part of the chapter, we describe the implementation of a tool that helps users to reflect on their notifications. We describe the components of the system and a brief evaluation.

In the second part of the chapter, we explore a new method to support users in managing notifications in the notification drawer. We report on a large-scale study and a controlled study that we conducted to gain insights about the notification management. Finally, we discuss design implications for smart notification management systems that we derived from the study results.

Parts of this chapter are based on the following publications:

D. Weber, A. Voit, H. V. Le, and N. Henze. “Notification Dashboard: Enabling Reflection on Mobile Notifications.” In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct.* MobileHCI ’16. Florence, Italy: ACM, 2016, pp. 936–941. ISBN: 978-1-4503-4413-5. DOI: 10.1145/2957265.2962660

D. Weber, A. Voit, J. Auda, S. Schneegass, and N. Henze. “Snooze!: Investigating the User-defined Deferral of Mobile Notifications.” In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services.* MobileHCI ’18. Barcelona, Spain: ACM, 2018, 2:1–2:13. ISBN: 978-1-4503-5898-9. DOI: 10.1145/3229434.3229436

4.1 Reflecting on Mobile Notifications

On current mobile phones, apps use notifications to gain the attention of users. However, notifications are not always in the users’ best interest. Apps might use notifications for the sole reason to increase interaction and, therefore, advertisement revenue. Because our attention is limited, it is increasingly important to find means to identify unwanted distractions.

Inspired by the previous chapter and prior work, we saw an opportunity to visualize the notification data for the end-users. We developed the *Notification Dashboard*, a personal single-user application that allows users to reflect on their own received notifications using visualizations. In the following sections, we first introduce our implementation and subsequently explain the available visualizations. Afterward, we summarize the results of three interviews with users of the dashboard.

4.1.1 Notification Dashboard

The *Notification Dashboard* consists of two separate components. The first component is a logging app for Android devices that records all notifications in a local log file. The second component is the dashboard itself that visualizes the log file. We will first present implementation details of the logging app and dashboard and afterward focus on the different visualizations.

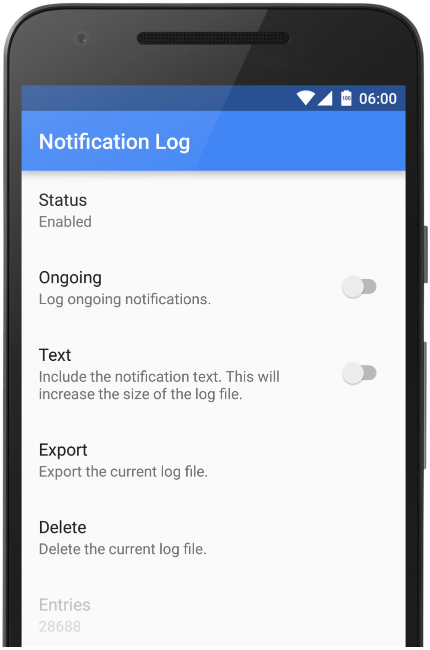


Figure 4.1: Screenshot of the notification logging app that is used to generate log files for the dashboard.

4.1.1.1 Logging App

Similar to the previous chapter, we used an app to log notifications (see Figure 4.1). On registering a new notification, the notification data is extracted and written into a log file. Users can grant the app permission to access the notifications and export or delete the log file. Furthermore, *ongoing* notifications (e.g., downloads or timers) can be filtered, as these types of notifications are typically updated frequently and would produce a huge number of log entries. Another option is the possibility to log either the actual text of notifications or only metadata. At the bottom of the app, the number of recorded notifications is shown.

4.1.1.2 Dashboard Implementation

The dashboard is a single-site web application implemented in HTML, CSS, and JavaScript. Apart from serving the static files, no web server is required. To protect the users' privacy, generated log files are imported and parsed entirely in the browser. We use a web-based implementation to utilize the larger screen real estate on desktop computers compared to mobile devices. Still, the web application was implemented with responsive design in mind and scales according to the size of the screen and is therefore usable even on smartphones. The charts in the dashboard are created using *Highcharts*[1], an interactive JavaScript charting library.

4.1.1.3 Data Visualization

Figure 4.2 shows a full-page screenshot of the dashboard. Figures 4.4, 4.5 - 4.6 show close-up views of the visualizations in the dashboard. The exemplary visualized data contains approximately 10 months of notification data. In the following, we describe the charts used in the dashboard.

General information In the top row, the name of the mobile device which was used for logging the notifications is shown. The second box shows the number of logged notifications. The third box shows the date range of the log file. Clicking on the date opens a date picker to set a custom start and end time. In the last box of the top row, the number of apps that created at least one notification is shown. Clicking on the number opens a dialog with a list of all apps, the number of notifications from each app, and an option to exclude the app from showing up in the dashboard.

Apps Figure 4.3a displays the average daily number of notifications for every app. A toggle allows switching between the top 10 apps and all applications. Hovering over the icons or bars of any of the charts causes tooltips with the exact

[1] https://www.highcharts.com/

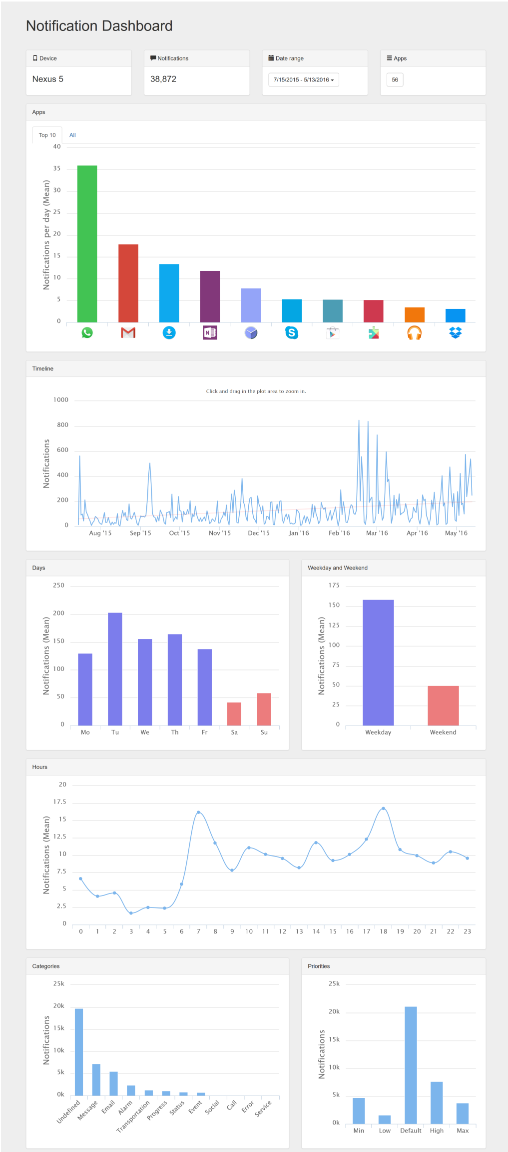
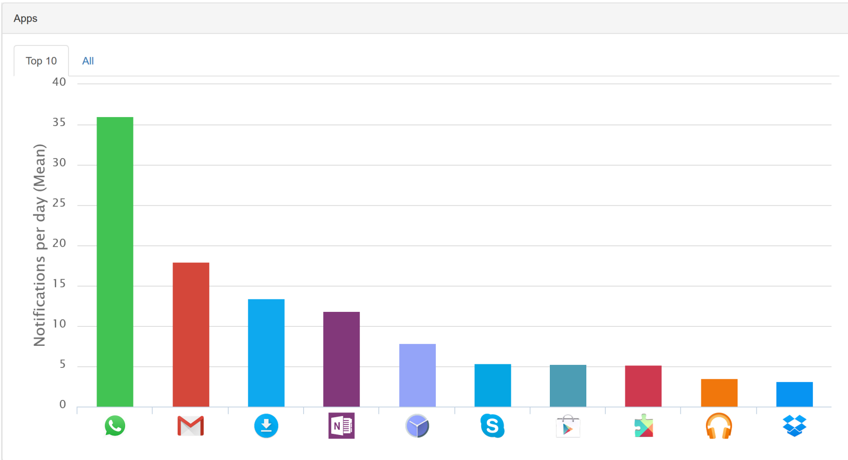
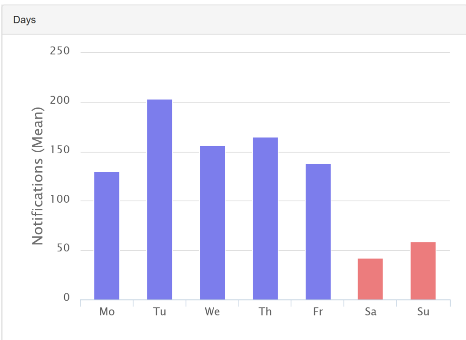


Figure 4.2: Full-page screenshot of the *Notification Dashboard*.

(a) Top 10



(b) Week (Monday–Sunday)



(c) Weekday vs. Weekend

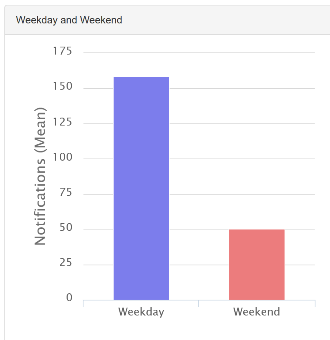


Figure 4.3: Aggregated graphs showing (a) the apps that created the most notifications, (b) notification count for every day of the week and (c) a weekday/weekend comparison.

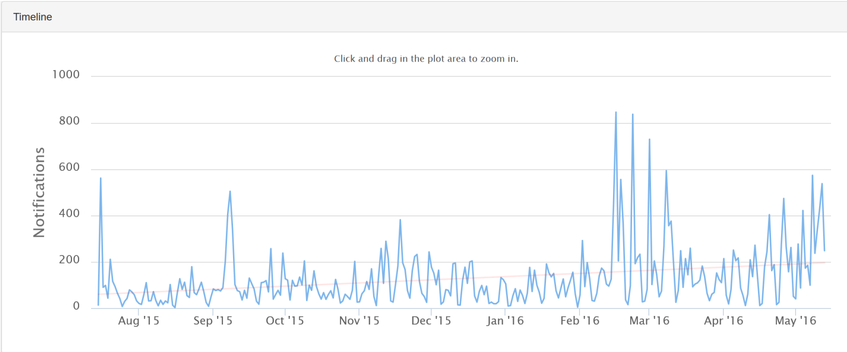


Figure 4.4: The timeline chart shows the number of notifications for each day (blue) and a trend line (red). In this data set, the number of notifications has increased over the course of 10 months.

values to appear. For easier identification, the dashboard automatically fetches app icons from the Google Play Store and extracts the dominant color for each app to color the bars.

Timeline The timeline in Figure 4.4 shows the total number of received notifications per day. Further, a trend line shows if the number of notifications increases or decreases over time. It is possible to zoom into this chart to see portions of the chart in detail.

Aggregated by day Figure 4.3b shows the average number of notifications for each day of the week, and Figure 4.3c compares weekdays with weekends.

Aggregated by hour Breaking down the data from the timeline to the week-view, the fifth row shows an aggregation of notifications for each hour of the day. Figure 4.5 shows the hours 0 to 23 and shows how many notifications were created for the particular hour.

Categories and priorities At the bottom left the categories of the notifications are shown (see Figure 4.6a). The category for every notification is defined by the

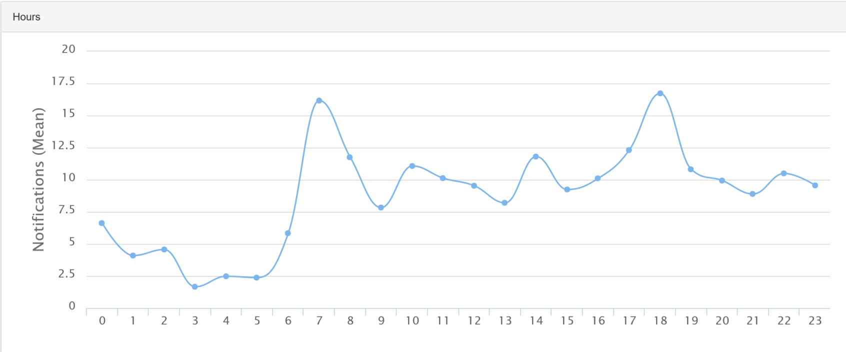


Figure 4.5: Aggregated view of notifications for each hour of the day. In the data set a spike at 7am and a second one at 6pm can be seen. The number of notifications quickly drops off after 0am.

app that issued the notification, for example, *email*, *message*, or *alarm*. In a similar manner, on the bottom right, notification priorities are shown (see Figure 4.6b). Similar to the categories, apps can set the priority of notifications. The possible values are *minimum*, *low*, *default*, *high*, and *maximum*. Notifications with the *minimum* priority do not appear in the status bar, and *high* priority notifications trigger heads-up notifications on newer Android versions.

4.1.2 Evaluation

We conducted three semi-structured interviews with participants (all male, $M = 22.3$, $SD = 2.3$) to collect general feedback on the *Notification Dashboard*. All participants were computer science students. We aimed to gather information on how people reflect on the information shown in the dashboard and their general opinion of it.

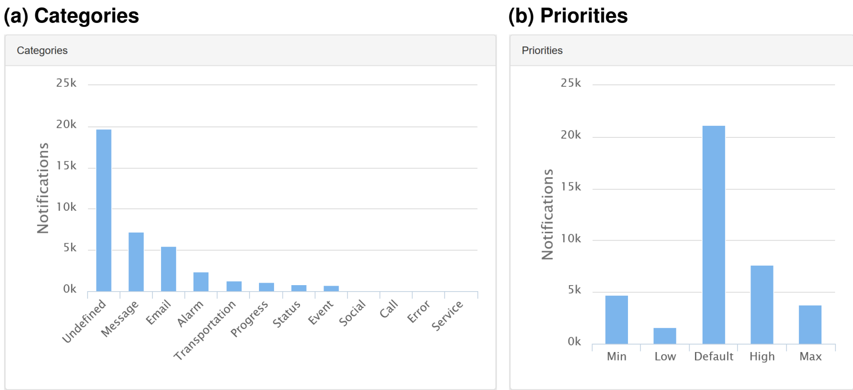


Figure 4.6: Notification categories and priorities are set by app developers.

4.1.2.1 Procedure

One month prior to the interview, we asked participants to install the *Notification Log* app, which collects metadata about their notifications (*ongoing* disabled). We briefed them on its functionality and privacy aspects. No other details were told about the study to avoid influencing them.

One month later, participants were invited to a 30-minute interview session. Each interview was held by two researchers. One researcher took notes, and the other conducted the interview. The interview consisted of the following parts: First, participants were asked to estimate the number of notifications they receive per day. Further, we asked them to guess how many apps are notifying them and which one shows the most. These questions were asked before showing them the *Notification Dashboard* to evaluate their assessment of received notifications on their smartphone.

After the interview, we showed them the *Notification Dashboard*, which visualizes the logs they collected over the past month. A brief introduction about the available visualizations was given, before allowing the participants the explore the dashboard on their own. After approximately 3 minutes of exploring the dashboard, we continued the interview. We asked them about information that they find interesting and how they would use this information to optimize their notification settings. We also asked if they find this information useful when integrated into the operating system itself (similar to battery statistics in Android). To collect ideas for future improvement, we asked them whether any visualization or details are missing that they would like to see.

4.1.2.2 Results

The estimation of the number of received notifications per day shows a high deviation amongst the participants. While P1 assumed that he receives at least 100 notifications per day, P2 guessed that he received 3 per day. P3, in contrast reportedly estimated his number as "*often, maybe 30*". Looking at the logged data in the dashboard, they all noticed that their estimation is off by a large amount. Here, P1 received 200, P2 received 60, while P3 received 100.

When asked about the number of apps that send notifications, we also observed a difference between the estimation and the real number shown in the dashboard. They estimated that a small number of their installed apps are showing notifications (P1: 7; P2: 8; P3: 3), whereas the dashboard shows more (P1: 26; P2: 15; P3: 12). Specifically, our participants were annoyed by notifications from "*Google Now at 2am*" (P1), "*updates for applications, Facebook and Twitter*" (P2) and "*9gag*" (P3). As a result, P2 uninstalled these apps and P3 disabled notifications for 9gag.

After participants explored the dashboard, we asked them about their first impression on the visualizations. All participants immediately noticed that their estimations were off by a noticeable amount ("*I didn't know that Google Now is showing so many notifications.*" - P1). Further, we observed that participants tended to describe characteristics of their notification logs, such as "*it seems like I text more when I'm at the university or going out at night*" (P2) and even try to explain them ("*Peaks [in the number of notifications] may also be due to WhatsApp notifications during [the soccer match]*" – P2). Participants also tried to generalize their notification behavior ("*On weekends, we write less with colleagues*" – P1). P3 liked the visualization but did not find anything surprising except the number of notifications that he wrongly estimated.

When asked about a possible integration into current operating systems, P1 liked the idea that it could be used to "*detect apps that are often distracting*." In contrast, P2 stated that while these "*would be nice*," he would not benefit from it since he usually notices anything that annoys him and acts upon it. Further, P3 stated that he usually has his notification settings on silent so that "*it does not bother [him] at all if notifications are incoming*."

In terms of future improvement, all participants agreed that it should be possible to see detailed information of specific days to investigate peaks in the timeline.

4.1.3 Discussion

The results suggest that people are not able to estimate the number of incoming notifications. The participants were all surprised that the actual amount of notifications was higher than the amount they initially guessed. On the one hand, this may be due to repetitive notifications that people start to ignore because they see them often without acting upon them, e.g., Wi-Fi notifications or system updates. On the other hand, this category of notifications mostly also have a low priority, and trigger neither sound nor vibration. When asked about the first impression of the visualizations, participants indicated their interest in using the visualization to reflect on their own notification behavior. They made this clear to us by voluntarily interpreting trends and generalizing them by referring to their usual smartphone usage behavior.

Most of this work was conducted in early 2016. On August 6, 2018, Google announced a notification dashboard for Android devices as part of the *Digital Wellbeing*[1] initiative. The *Digital Wellbeing* menu item appears in the settings of selected devices. It provides a graph showing the total number of notifications shown per day for all installed and individual apps. Further, it shows a list of apps ordered by the total number of notifications they created and provides shortcuts to the apps' notification settings to enable users to enable or disable notifications for specific apps quickly.

[1] https://blog.google/products/pixel/try-out-digital-wellbeing-find-your-own-balance-pixel/

4.2 User-Defined Deferral of Mobile Notifications

Notifications can induce negative effects. They can be distracting, might cause negative emotions, or are just not important for their recipients [114, 121, 126, 137]. One strand of research aims to reduce distraction by finding more suitable moments for notification delivery. With *SMS* and instant messaging (*IM*) services generating numerous notifications, further research is needed to develop more appropriate notification management services [23]. Previous work found that delaying the delivery of notifications to opportune moments helps to reduce distraction [43, 106, 109, 110, 114, 143]. Contextual information about the user can be used to determine suitable moments to deliver notifications. These moments promise to alleviate negative effects on the recipient of notifications.

Despite the increasing interest in finding opportune moments for delivering notifications, little is known about the type of notification that should be delayed and how long users like the delay to be. Therefore, we developed the Android application *NHistory* that enables users to *snooze* notifications similar to snoozing an alarm clock. In contrast to previous approaches, the app lets the recipient of a notification specify the moment when the notification should be redelivered. Notifications can be snoozed for a specific duration or to a later point-in-time, and *NHistory* reissues them after the specified period. Users can also access a history of all notifications they have received, enabling them to go through the notifications and snooze desired notifications to more appropriate moments. By snoozing notifications to more suitable times, users can reduce the number of pending notifications in the notification drawer. While this is indeed a positive aspect of the app, we want to note that this does not necessarily mitigate adverse effects on the notifications' recipient. In some cases, it might be even worse than an uncontrolled notification delivery, because a snoozed and redelivered notification may induce a second interruption. However, observing how a recipient of a notification postpones its arrival enables us to gain insights into the types of notifications users want to defer and how long users would like the deferral to be.

In the second part of this chapter, we report a year-long in-the-wild study with 295 active users. The goal of the study was to understand which notifications users want to defer and the periods of time users want to defer them. We focused on deriving insights into how users postpone notifications rather than eliminating negative effects that might influence the notification recipient directly. We collected usage data, including applications that issued notifications, how long notifications were snoozed for, and when they were re-triggered. To complement the collected quantitative data, we conducted a second study with 16 participants for one week and subsequently interviewed them. In both studies, snoozing was mainly used to defer notifications related to people and events. Reasons for the deferral of notifications were manifold and aligned with daily routines. Most notifications were deferred to the same day or the next morning, indicating an upper bound for the deferral of notifications. Based on our findings, we derive design implications that can inform the design of future smart notification systems.

4.2.1 Related Work

To alleviate adverse effects of notifications, merely disabling notifications is not a suitable solution [125, 126]. The desire to meet social expectations or the fear to appear rude by not replying to messages on time are reasons for keeping notifications enabled. Further, notifications act as reminders for important events. Stawarz et al. investigated efficient reminders for taking medicine [146]. Time-based reminders offer only a small benefit to users because often the reminder cannot be postponed. Immediate action is required by the user so that taking medicine is not forgotten. Stawarz et al. propose that users should be able to defer notifications issued by reminders if they are not able to respond immediately. To reduce the negative effects of the increasing number of notifications, previous research focused on developing models, rules, and systems that manage notification delivery. However, users might not accept a notification management system that removes important notifications [95].

4.2.1.1 Opportune Moments

To identify moments in which notifications can be best presented to the user, researchers exploited contextual information, notification content, and personal traits. Corno et al. introduced a smart notification system that uses machine learning to manage incoming notifications [26]. These algorithms use information about the context of the user and the user itself, such as location and user's activity. Several approaches used sensor data to develop context-aware notification systems [60, 73]. Kern and Schiele introduced a model that makes use of body-worn sensor data. The model supports notification classification according to the user's current context to determine his or her interruptibility. Horvitz et al. introduced *BusyBody*, a system running on a desktop computer that predicts the cost of an interruption on a user [64]. It analyzes desktop events and the user's context (e.g., if he or she is speaking or not) to train a model. Mehrotra et al. used the context of a notification recipient in combination with the content of the notification to realize a non-disruptive notification mechanism [94]. Notifications were grouped into different categories according to the application that issued the notification as well as the relationship between sender and receiver. They showed that their machine learning based predictors could outperform user-defined rules for notification delivery on smartphones. Further, Mehrotra et al. proposed an interruptibility management solution for mobile notifications [93]. Their system extracts rules to handle notifications automatically based on how the user is interacting with his or her device.

Yuan et al. proposed a model which can determine the interruptibility of its user [194]. The model determines if the user is available to react to an incoming notification using sensory data and additional information such as the mood of the user. They found that including personality traits is important for predicting interruptibility. Siewiorek et al. proposed *SenSay*, a context-aware mobile phone [143]. It uses sensors to acquire contextual information about its user. Using this information, the device can switch to the uninterruptible state. In this state, unwanted interruptions will not be delivered to the user.

Pielot et al. investigated if mobile phone usage patterns can be utilized to train a machine learning model that is able to detect boredom [123]. They suggest that bored people may be more appreciative of incoming notifications. Dingler et al. investigated if detected boredom can be used to engage a user in micro-learning sessions through notifications [34]. Indeed people search for stimulation during boredom, but they found that a mentally demanding task is not suitable. To voluntary engage users to interact with recommended content, Pielot et al. used a machine learning approach to determine opportune moments for notification delivery [120]. Higher user engagement was observed if notifications were issued at these moments. Further, they suggest observing past interest in notification content to reduce future interruptions.

To further mitigate negative effects caused by interruptions, previous research investigated the suitability of breakpoints for notification delivery. A breakpoint promises to be more suitable for notification delivery because they occur between two consecutive activities [101]. Recipients of interruptions during these moments may not be as strongly influenced by interruption related adverse effects as they would be while performing a specific activity or task. Fischer et al. reported that people attended to notifications quicker at the end of a mobile interaction (i.e., calling a contact or reading *SMS*) compared to notifications received at random times [43]. Different tools were suggested to cope with the number of incoming notifications. Ho and Intille used data from accelerometers to automatically detect transitions between physical activities [60]. They suggested that these transitions might be suitable for reducing the negative effects caused by interruptions from mobile devices.

Okoshi et al. proposed *Attelia*, a service that identifies breakpoints for notification delivery [106, 109–111]. *Attelia* runs on the user's smartphone and can detect breakpoints of the user's activity on his or her mobile device based on running applications and machine learning techniques. Further, it can detect physical breakpoints through smartwatches. Investigating breakpoints in the wild, Okoshi et al. conducted a large-scale study with their breakpoint detection system included in a popular Android application. They reduced the response time to the delayed notifications significantly. They also observed a continuously increasing number of clicks as well as a higher level of user engagement. Park et al. proposed the breakpoint-based *Social Context-Aware smartphone Notification system (SCAN)* [114]. *SCAN* can identify breakpoints to which it defers incoming notifications. Park et al. reported that *SCAN* could help the participants to better focus on their social interaction. Pejovic and Musolesi proposed *InterruptMe*, a library for interruption management for Android [116]. With the use of contextual information about the user, the library determines whether he or she is interruptible. *InterruptMe* can inform other apps when an opportune moment for interruption occurs.

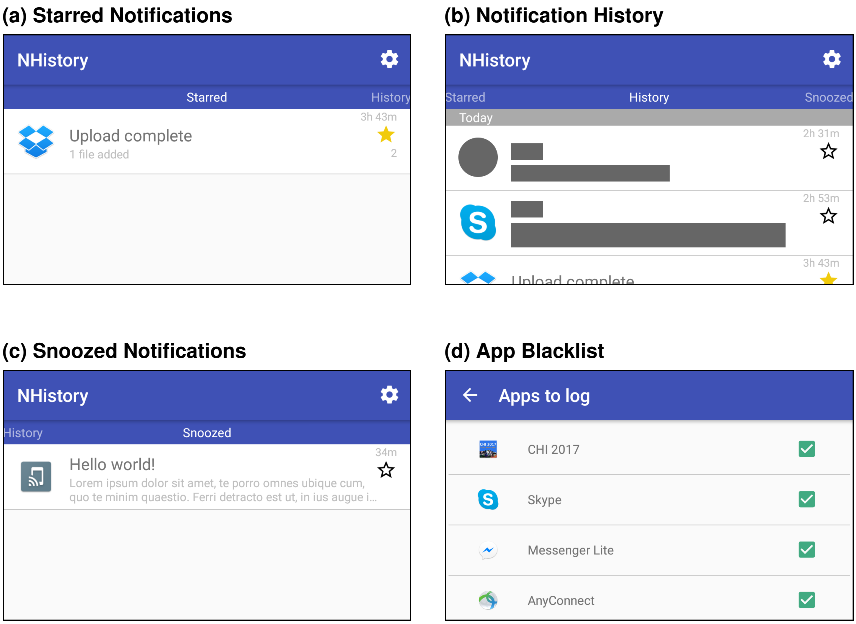


Figure 4.7: Screenshots of the *NHistory* app. (a) List of starred notifications, (b) notification history showing active and past notifications (main view), (c) list of snoozed notifications, and (d) blacklist to exclude notifications from specific apps from the history.

In summary, previous work showed that delaying the delivery of notifications to opportune moments can partially reduce notifications' negative effects. A body of work focused on automatically finding opportune moments for notification delivery. However, little is known about which types of notifications and to what times users would defer notifications manually.

4.2.2 System

In contrast to previous work, we investigate the manual deferral of notifications to study reasons for deferring notifications, selected moments, and types of notifications to better understand why and how users defer notifications. Therefore, we developed an Android application that enables users to defer notifications manually. In the following, we first summarize Android's notification mechanisms and afterward describe how *NHistory* extends the Android system.

4.2.2.1 Notifications in Android

Notifications play a central role in the Android mobile operating system. Notifications are accessible from the notification drawer; a list of active notifications that can be accessed by swiping down from the top of the screen. Android's status bar displays icons to indicate active notifications in the notification drawer. Newer versions of Android display notifications on the lock screen as well. Notifications are ephemeral. Users can dismiss single or all active notifications at once. Dismissing a notification removes it from the system, an action that cannot be undone. An active notification might be seen as a "nagging reminder" to take action while dismissing a notification might cause the user to forget about taking said action. Users have to decide if they want to respond to the notification (e.g., reply to an instant message), dismiss the notification, or just ignore it until a later point in time.

4.2.2.2 NHistory

We developed *NHistory* to investigate the user-defined deferral of mobile notifications. The app provides a timeline of all active and dismissed notifications, sorted by the time of creation. Further, the app allows users to dismiss notifications and automatically re-triggers them after a user-defined duration or at a user-defined point-in-time. This behavior of temporarily muting is known from alarm clocks and generally referred to as "snoozing" an alarm. To enable this functionality, the app uses Android's *Notification Listener* API [7]. When first opening *NHistory*, users are informed about its data collection. To use the app, users have to agree to the data collection explicitly; declining closes the app. After accepting the data collection, the user is prompted to grant *NHistory* permission to access notifications. Then, the app's main view is shown (Figure 4.7b). The *history* view shows a list of active and past notifications. Users can tap on notifications to expand them. "Ongoing" notifications, like downloads, are ignored by *NHistory*. Clicking on the star icon next to a notification allows users to bookmark them. Starred notifications are shown to the left of the history (Figure 4.7a) and can be removed by clicking on the star icon again. To the right of the history, notifications that are currently snoozed are listed (Figure 4.7c). In the settings, users can set a limit of how many notifications should be stored in the history or exclude specific apps from appearing in the history using a blacklist (Figure 4.7d). Users can access *NHistory* by either clicking on its icon in the launcher or through an optional persistent notification in the notification drawer.

4.2.2.3 Snoozing Notifications

NHistory offers multiple ways to snooze notifications. In the app itself, users can long-press on a notification or expand a notification and subsequently click on a snooze button. Active notifications are automatically dismissed from the notification drawer when snoozing them. While the Android system does not allow modifying the notification drawer, we also implemented an option to snooze notifications from the drawer directly. *NHistory* detects when a user dismisses a notification from the notification drawer. The app then creates a temporary notification in its place, that is shown for five seconds. Clicking this temporary "snooze notification" allows the user to trigger the snooze action for the dismissed notification (see Figure 4.8). The user is then prompted to define *how long* or *until when* the notification should be snoozed. We implemented the two "snooze" methods *duration* and *point-in-time* with corresponding options. Both methods provide a set of eight options. The *duration* method features fixed options, while the *point-in-time* options depend on the time of the day (see Figure 4.9). Both methods feature a custom duration or custom point-in-time as the first option. On the first start of the app, one of the snooze methods is randomly assigned. Users can change the assigned method in the settings.

4.2.2.4 Data Collection

NHistory collects information about the device, notification metadata, and interaction with the app. Upon installation of *NHistory*, a random identifier is created. All requests to our server at the University of Stuttgart use a secure connection

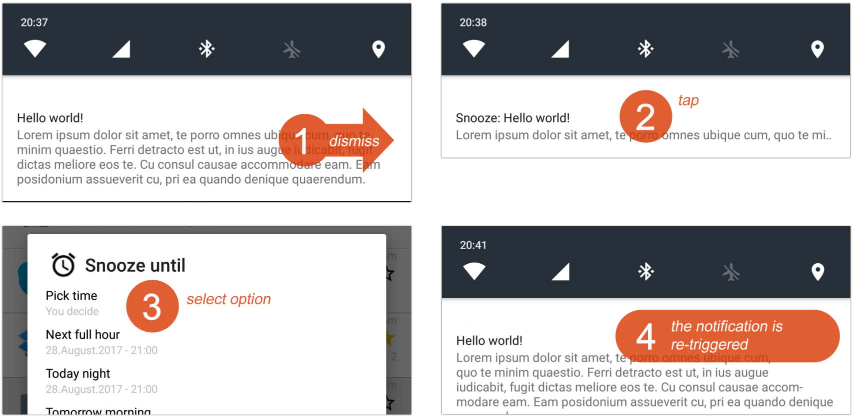


Figure 4.8: Snoozing a notification via the notification drawer. 1. The user dismisses the notification. 2. *NHistory* briefly shows a "snooze notification." 3. Tapping on the snooze notification opens the snooze options. 4. After the selected duration, the notification is re-triggered.

(a) Duration Method **(b) Point-in-Time Method**

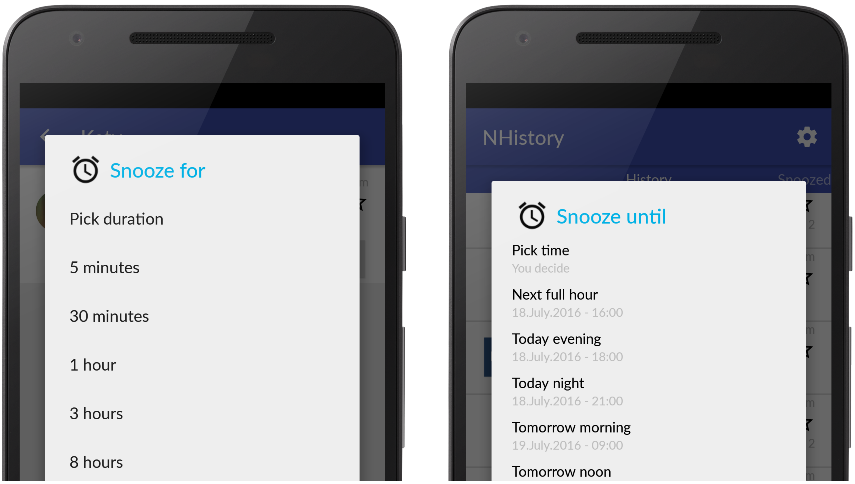


Figure 4.9: The available snooze options depend on the selected method. Left: Duration. Right: Point-in-time (depending on the time of the day).

Category	Normalized%	Total Events	Users	Apps
Calendar/Reminder	18.68	430	41	26
Email	8.19	111	30	11
Game	2.95	22	6	7
Health/Fitness	.25	15	5	5
Media	3.08	26	12	9
News	1.82	8	5	6
Phone	8.45	102	22	10
Shopping/Finance	3.75	26	16	13
SMS/IM	28.23	251	64	30
Social	9.20	73	28	20
System	6.56	40	24	12
Tool	8.83	87	27	36

Table 4.1: Normalized distribution of the initial snooze events for all users and all categories for the in-the-wild study (1,191 events).

and include the identifier. Device information is sent periodically, including device type, Android version, and system language. The app records all major user interaction, including starring and snoozing notifications, and blacklisting apps from the history. Further, package names and timestamps of all notifications created on the device are recorded. We did not record any text or information that could be used to identify users. All recorded data is queued for sending until a Wi-Fi connection is established.

4.2.3 In-The-Wild Study

To gain insights into the types of notifications users want to defer, and for how long, we conducted an in-the-wild study.

4.2.3.1 Method and Participants

We released *NHistory* on the Google Play Store as a free download. People across the globe were able to download and use the application. The app automatically reported anonymized usage and notification metadata back to us. Users who downloaded the app were informed about the data collection on the first start of the app. All users had to consent to the data collection to use the app.

Between January 12, 2017, and January 12, 2018, *NHistory* was installed on $1,555$ devices. According to the Google Play Store statistics, the app was downloaded from 95 countries, with most downloads originating from India (43.08%), the United States (12.43%), and Germany (4.25%). The most popular device languages were variants of English (81.28%), followed by German (3.67%). Android versions 6.0 (33.89%), 7.0 (27.65%), and 7.1 (16.98%) contributed the most to the user base. Only 13 users installed the app on Android tablets; the remaining $1,542$ installs were on smartphones. Thirty users rated the app, resulting in an average rating of 4.37 stars (1=worst; 5=best).

4.2.3.2 Results

Of the $1,555$ users who downloaded the app, over half (876) agreed to the terms of the study on the first start. It is important to note that users tend to try out free apps and quickly uninstall them if they do not fulfill their expectations. Indeed, a number of users uninstalled the app right after the setup. 581 users used the app for less than two days. Thus, we excluded them from our analysis. The remaining 295 users used the app between 2 and 360 days (M=46; SD=68; Md=15 days).

Notifications and Apps In total, we logged $20,345,277$ notifications from $3,667$ apps. Users had on average 44 apps notifying them (SD=27). We recorded on average $1,960$ notification events per user per day (SD=$3,942$; Md=832). At first glance, these numbers seem unusually high. The reason for these numbers is that in Android updating an existing notification is realized by replacing the original notification. Some apps continuously update notifications in the background, e.g., to display location updates or battery statistics. Although these background updates might happen every other second, they are often not noticeable to the user because they happen silently. Since we did not record the text of notifications, we were unable to filter these updates. The apps that created the most notifications were *Google Maps* ($2,049,889$ notifications; 210 users), *Power Clean* ($1,665,971$ notifications; 6 users), and *WhatsApp* ($1,325,423$ notifications; 165 users).

Most notifications were created between 8pm and 10pm, peaking at 9pm. At 4am the least number of notifications were created. Regarding the days of the week, the notifications were evenly distributed over weekdays and the weekend.

Notification History Blacklist 109 of 295 users made use of the option to exclude apps from the history. These users excluded between 1 and 435 apps (M=46; SD=81; Md=9), totaling in $2,760$ different apps. Some users added apps to the blacklist that did not yet post notifications. The apps excluded by most users were the *Android OS* (44 users), the *Android System UI* (43 users), and the *Google Play Store* (42 users).

Starred Notifications Only 52 of 295 users starred notifications. Users starred between 1 and 48 notifications (M=3; SD=7; Md=1) for at least one hour, totaling in 159 star events from 58 apps. Notifications from apps that were starred by more than one user include *Google Calendar* (32 events; 4 users), *Facebook* (7 events; 5 users), and *Google Keep* (6 events; 4 users).

Snoozed Notifications We recorded $2,648$ snooze events from 151 users and 219 apps. Since users interested in the snooze functionality would likely try it after installing the app, we excluded snooze events from the day when *NHistory* was installed, resulting in $2,390$ snooze events from 129 users and 185 apps. 66 users were randomly assigned the *point-in-time (pit)* method during installation, and 63 users were assigned the *duration* method. 79 users stayed with their assigned method (42 *duration*, 37 *pit*), 32 users switched to the other method (14 from *duration* to *pit*, 18 from *pit* to *duration*), and 18 users switched but returned to their assigned method (7 *duration*, 11 *pit*).

Looking closer at the source of the $2,390$ snooze events showed that $1,191$ individual notifications were snoozed. 728 notifications were snoozed once, and 463 notifications were snoozed multiple times, with a single notification being snoozed 18 times by a user. We will now report on the $1,191$ initial snooze events, followed by the re-snooze events. The $1,191$ initial snooze events by 129 users

Snoozing of \ Compared to	Health/Fitness			SMS/IM		
	T	diff.	p	T	diff.	p
Calendar/Reminder	5.00	↗	$< .001*$	40.61		$= .002$
Email	27.00	↗	$< .001*$	51.50	↘	$< .001*$
Game	3.20		$= .002$	39.50	↘	$< .001*$
Health/Fitness	—		—	9.50	↘	$< .001*$
Media	2.67		$= .001$	38.00	↘	$< .001*$
News	2.50		$= .002$	25.50	↘	$< .001*$
Phone	17.50	↗	$< .001*$	64.00	↘	$< .001*$
Shopping/Finance	3.50		$= .001$	65.00	↘	$< .001*$
SMS/IM	9.50	↗	$< .001*$	—		—
Social	7.00	↗	$< .001*$	73.50	↘	$< .001*$
System	20.00	↗	$< .001*$	61.00	↘	$< .001*$
Tool	7.00	↗	$< .001*$	34.47		$= .001$

Table 4.2: Results of the post-hoc analysis with Wilcoxon signed-rank tests of the normalized snooze categories (in-the-wild study). p-values are Holm-Bonferroni adjusted. Significant differences are marked with "*", significant less snoozed categories with "↘", and significantly more snoozed categories with "↗".

break down to 1 to 402 snooze events per user (M=9; SD=36; Md=3). The apps that were snoozed the most were the *Google Calendar* (340 events; 18 users), *WhatsApp* (72 events; 21 users), and *Outlook* (58 events; 4 users).

To abstract from single apps, two researchers independently categorized the 185 apps. The resulting 12 categories are based on Google Play Store listings and prior literature [137] (see Table 4.1). Disagreements were discussed until an agreement was reached. Table 4.1 shows the normalized distribution of snooze events based on the categories. The categories with the most snooze events were *SMS/IM* (28.23%), *Calendar/Reminder* (18.68%), and *Social* (9.20%). We conducted a Friedman test to compare the normalized data of snoozed notifications from different categories. The results show that the category influences the users' snoozing behavior ($\chi^2(11) = 189.96, p < .001$).

We conducted a post-hoc analysis with Wilcoxon signed-rank tests and applied a Holm-Bonferroni correction (see Table 4.2). The results show that notifications

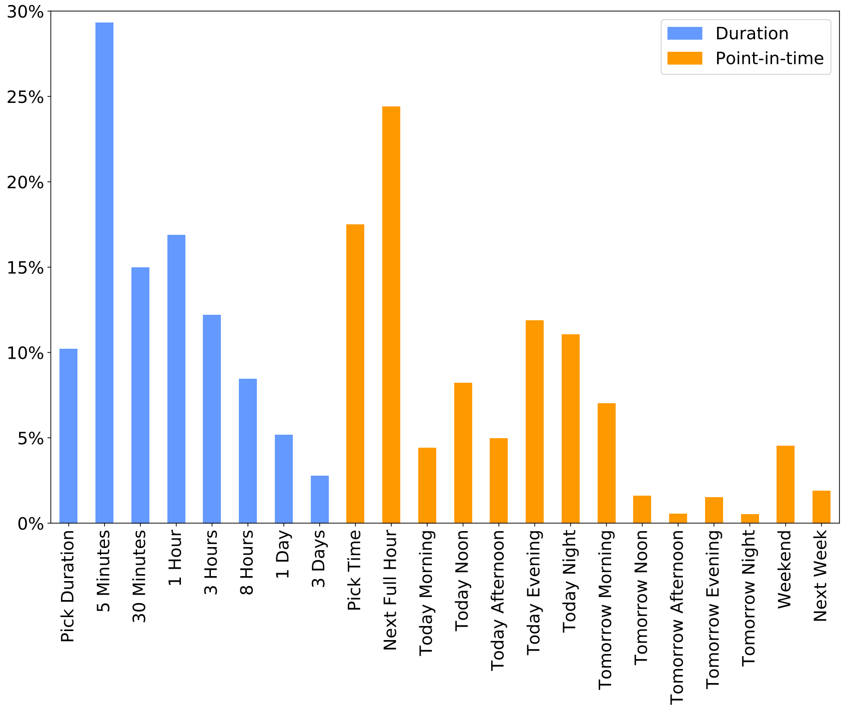


Figure 4.10: Normalized distribution of selected options for all 1,191 initial snooze events, both methods, and all users (in-the-wild study). Duration: 797 events from 75 users. Point-in-time: 394 events from 64 users.

from the category *Health/Fitness* were significantly *less often* snoozed than from the categories *SMS/IM*, *Calendar/Reminder*, *Social*, *Tool*, *Phone*, *Email*, and *System*. Further, notifications from the category *SMS/IM* were significantly *more often* snoozed than all other categories except *Calendar/Reminder* and *Tool*.

The usage of the *duration* and *point-in-time* methods was equally distributed. Normalized over all users, 54% of the 1,191 initial snooze events were executed using the *duration* method and 46% using *point-in-time*. We normalized the selected options for all users and both methods (see Figure 4.10). The most popular options for the *duration* method were “5 minutes” (29.32%), “1 hour” (16.88%), and “30 minutes” (14.99%). The custom duration option was mainly used to fill the gaps between the pre-defined options, with most durations being between 10-15 minutes and 2 hours. The median custom duration was 1 hour, and the maximum duration 5 days. For the *point-in-time* method, the most popular options were "next full hour" (24.42%), picking a custom time (17.50%), and "today evening" (11.87%). For custom points-in-time, users mostly selected times dividable by 15 minutes, with a median snooze time of 5.62 hours, and a maximum point-in-time of 4.67 days in the future.

Figure 4.11 shows the posted, snoozed, and re-triggered times of the initial snooze events. Most notifications that were snoozed were posted in the morning and early noon. Users quickly attended the notifications, resulting in low reaction times (Md=35*min*). Consequently, the distribution of the snooze events follows the posted events, peaking in the morning and dropping over the course of the day. During the day, we observed notifications being snoozed for shorter time spans. In the evening and until after midnight the time spans notifications were snoozed for increased by several hours, as users snoozed the notifications until the next morning. Most snoozed notifications were re-triggered during the day, between 8am and 10pm. We can see peaks at 9am, 12pm, 3pm, 6pm, and 9pm. 9am being the highest peak can be explained by notifications that were snoozed for a short time in the early morning and notifications from the previous day. These times are likely influenced by the pre-defined options of *NHistory* but might also correlate with before and after work hours, and lunch breaks. We want to highlight that 79.26% of the notifications were posted, snoozed, and re-triggered on the same day. Only 16.54% were re-triggered on the following day and 4.20% over multiple days. This results in an overall mean deferral time of 689 minutes (SD=1,543; Md=274), between a notification being posted and eventually re-triggered.

As mentioned earlier, 463 notifications were snoozed more than once (Min=2; Max=18; Md=3). These notifications were mainly from the categories *Calendar/Reminder* (55.94%), *SMS/IM* (15.77%), and *Email* (10.15%). Looking at the times when these notifications were initially posted and finally re-triggered, we still found that most were on the same day (69.61%), some on the following day (20.49%), and more than two days being an exception (9.91%).

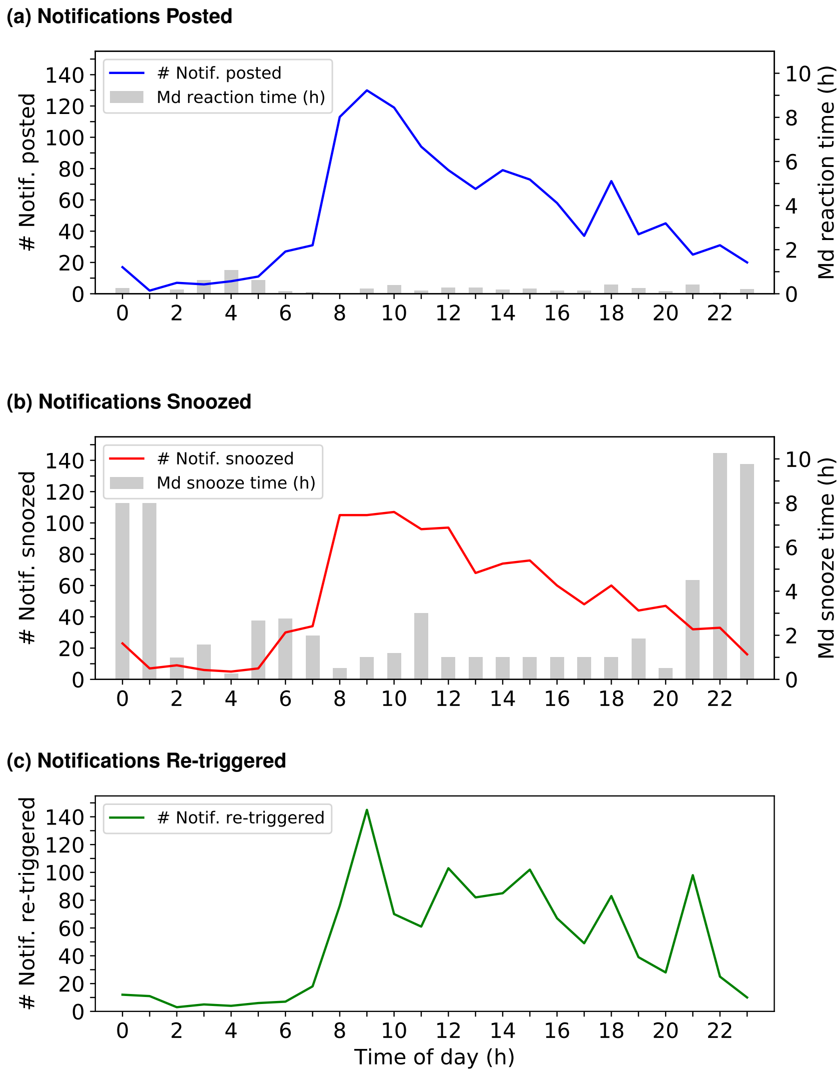


Figure 4.11: Plots showing the distribution of the 1,191 initial snooze events from the in-the-wild study. The line charts show the time distribution when the notifications were posted (top), snoozed (middle), and re-triggered (bottom). The bar charts show the median reaction time (top) and median snooze time (middle).

Category	Normalized%	Total Events	Users	Apps
Calendar/Reminder	4.47	5	3	3
Email	13.66	21	5	5
Game	3.33	1	1	1
Health/Fitness	2.76	3	2	2
Media	.35	1	1	1
News	4.39	4	2	3
Phone	4.44	2	2	1
Shopping/Finance	2.22	1	1	1
SMS/IM	49.15	48	13	6
Social	4.11	9	3	4
System	8.03	5	4	2
Tool	3.07	9	2	4

Table 4.3: Normalized distribution of the initial snooze events for all users and all categories for the controlled study (109 events). In both studies, notifications of the category *SMS/IM* were proportionally snoozed most often.

4.2.3.3 Summary

Temporarily snoozing was more popular than permanently starring notifications, indicating the ephemeral nature of notifications. We assume that users only snoozed particularly important notifications that they were unable to attend directly. Users mainly snoozed notifications from the categories *SMS/IM* and *Calendar/Reminder*. They were fast to snooze notifications, with overall low reaction times. The number of snooze events spiked in the morning and declined over the course of the day. Most notifications were re-triggered in the morning, at noon, and in the evening. These times likely correlate with before and after work hours, and lunch breaks. We found that users snoozed notifications mainly to the same day. Only few notifications were snoozed to the following day and even less to the day after.

4.2.4 Controlled Study

The in-the-wild study provided insights into the types of notifications, and times users are interested in deferring notifications. However, we know little about the users' motivation. To complement the quantitative results, we conducted a second in-situ study with a smaller set of participants and subsequently interviewed them [97].

4.2.4.1 Method

In this more controlled study, we invited participants to use *NHistory* for one week. During this study, we used the same version of *NHistory* as for the in-the-wild study and collected the same data. Additionally, we conducted semi-structured interviews at the end of the controlled study to gain further insights. In the interviews, we asked the participants to estimate how many notifications they receive on a daily basis, how the number of notifications affects them, and how they deal with interruptions. Further, we asked them about their opinions on *NHistory* and the provided functionality.

We individually invited the participants to our lab. All participants signed a consent form, informing them about the procedure of the study and the data collection. Further, they were informed that they are allowed to withdraw their study participation at any time. We then asked them to fill out a questionnaire about demographic data. Afterward, we introduced them to *NHistory*. We installed the app on their personal Android smartphones and walked them through the different features. We then asked them to use the app as they see fit for one week. We explicitly told them that they do not have to use it at all if they find it unnecessary. After seven full days of usage, we again invited the participants to our lab and conducted interviews with a duration of approximately 30 minutes each. Participants received EUR 15 for their participation.

4.2.4.2 Participants

We invited 17 participants to use *NHistory* for one week. One participant withdrew the study participation; thus, we excluded this participant from our evaluation. The remaining 16 participants (4 female, 12 male) were 20-36 years old (M=26.50; SD=4.07). Four of them were PhD students, eight students, two software engineers, and two teachers. All but the teachers had a technical background.

4.2.4.3 Results

The device language of the participants' smartphones was set to German in 12 cases, and English for the other four cases. Most participants used a device running Android 6.0 (10) and 5.0 (3). Android 5.1, 7.0, and 7.1 were used once. After the day of installation, all participants used *NHistory* for seven full days to cover every day of the week.

Notifications and Apps During the seven days of the study, the 16 participants received between 489 and $67,332$ notifications. The participants had between 13 and 44 apps notifying them. In total, we logged $102,386$ notifications from 147 apps. The apps that created the most notifications were *GPS Status & Toolbox* ($62,597$ notifications; 1 participant), *WhatsApp* ($12,586$ notifications; 15 participants), and the *Android OS* ($6,079$ notifications; 14 participants). Notification creation peaked between 6pm and 8pm. At 2am the least number of notifications were created. Participants received twice the amount of notifications on the weekend compared to the rest of the week.

Notification History Blacklist Eight of the 16 participants made use of the option to exclude specific apps from the history. These eight participants excluded between 1 and 4 apps, totaling in 13 different apps. The apps excluded by most participants were *Google Maps* (3 participants) and the *Android OS* (2 participants).

Starred Notifications Participants rarely used the option to star notifications. Only three of 16 participants used the feature at least once, resulting in five star events. This again shows that participants were not interested in permanently bookmarking notifications.

Snoozed Notifications We recorded 116 snooze events from 15 participants and 33 apps. One participant did not snooze notifications. We assigned the *duration* and *point-in-time (pit)* methods evenly to the participants. Seven participants stayed with their assigned method (2 *duration*, 5 *pit*), 6 switched to the other method (4 *duration* to *pit*, 2 *pit* to *duration*), and 3 switched but returned to their assigned method (2 *duration*, 1 *pit*).

In total, 109 individual notifications were snoozed. 105 were snoozed once and 4 notifications were snoozed multiple times. We will now report the 109 initial snooze events. The 109 initial snooze events by 15 participants break down to 2 to 25 snooze events per participant (Md=5). The apps that were snoozed the most were *WhatsApp* (39 events; 11 participants), *Blue Mail* (10 events; 1 participant), *Facebook Messenger* (5 events; 4 participants), and *Facebook* (5 events; 2 participants). We again categorized all apps and conducted a Friedman test to compare the normalized data of snoozed notifications from different application categories. The results show that the application category influences the users' snoozing behavior ($\chi^2(11) = 50.06, p < .001$). We conducted a post-hoc analysis with Holm-Bonferroni corrected Wilcoxon signed-rank tests. Our post-hoc analysis found no significant differences between the categories. As shown in Table 4.3, most snoozed notifications were from the categories *SMS/IM* (49.15%) and *Email* (13.66%).

We again saw a similar usage of the *duration* and *point-in-time* methods, and the provided options. However, in contrast to the in-the-wild study, the snooze events were more evenly distributed over the day. Peaks for snooze events can be seen at 10am, 1pm, and 5pm. Most notifications were re-triggered at 12pm, 3pm, 6pm, and 9pm. Compared to the in-the-wild study, 9am was an unpopular option to re-trigger notifications. We again found that most notifications were snoozed and re-triggered on the same day (79.82%), some on the following day (15.60%), and only few on more than two days (4.59%).

4.2.4.4 Interviews

After the participants used *NHistory* for a week, we invited them back to our lab and conducted semi-structured interviews. We used open coding for the analysis of the interviews. Three researchers coded the answers individually. Disagreements were discussed until an agreement was reached.

Handling Notifications and Interruptions Participants estimated that they receive 15-150 notifications from 2-10 apps per day. One participant stated that he perceives the number of notifications he has to deal with as “low,” eight participants perceived them as “between okay and high” and seven participants as “too high.” Six participants felt “never or rarely” being interrupted by notifications, seven “sometimes,” and three “often.” Participants stated that they usually immediately react when they notice a notification. They stated that they attend communication-related notifications as soon as possible and other types of notifications if they have time. Silencing the phone was mentioned as a method to cope with the number of notifications participants receive on a daily basis. One participant stated that he sometimes places the phone out of reach in addition to silencing it. To cope with annoying notifications, participants mentioned simply ignoring them, dismissing them immediately, and revoking the permission to show notifications. One participant reported uninstalling apps because of annoying notifications.

Notification History and Blacklist Five participants stated that they did not use the notification history feature because they saw no need for it. Six other participants stated that they used the history to read notifications. P4 explained that the history enables him to read longer messages than the notification drawer. He further remarked that another benefit of reading notifications in the history is that the corresponding messages are not marked as “read.” Thus, chat partners do not expect him to reply immediately. Participants used the history to snooze notifications and to reflect on notifications they received during the day. P10 used the history to remember important notifications.

Nine participants stated that they saw no need to blacklist apps in the notification history. Three other participants said that they blacklisted apps which create many unimportant notifications. Examples of this kind of notifications were notifications from music apps (P16), GPS tracking apps (P4), system notifications about available Wi-Fi networks (P12), and timers (P16). These examples match the findings of the quantitative results. This indicates that these kinds of notifications are only of relevance for a limited time and participants see no point in revisiting them. Additionally, P3 excluded an app because the app generates notifications with sensitive data.

Starring Notifications Thirteen participants found starring notifications unnecessary. P10 explained that notifications are temporary and should not be saved persistently; instead, he snoozes them if necessary. In contrast, P4 stated that he starred a notification which he described as "cool and memorable." P3 stated that he starred notifications to remember important information and tasks because it allows quick access to the notification.

Snoozing Notifications Participants explained that they snoozed notifications at work, university, or while studying. Further contexts for snoozing notifications were being on-the-go, during sport, gaming, driving, and because they were tired.

The participants provided examples when snoozing notifications was beneficial. For instance, P2 explained that she likes snoozing notifications because she does not have to deal with the notification itself anymore if the system can remind her. Furthermore, participants found that snoozing notifications supports their attention management. P16 explained that he is less distracted if he snoozes interesting notifications to a more appropriate time. Additionally, P4 explained that snoozing notifications helps to keep the status bar clean.

Participants also told us what they disliked. Snoozing notifications manually was sometimes regarded as unnecessary, as it takes the same amount of time as, for example, answering a short message. The "snooze notification" that appeared after dismissing a notification was described as annoying. P3 explained that most of the time she does not want to snooze a notification when dismissing it. Another participant complained about the short period the "snooze notification" was shown ($5sec$) because he was sometimes too slow to snooze the notification directly in the notification drawer and had to open *NHistory* to snooze the notification from the history instead (P4). P2 reported that she was sometimes not able to find a notification in the history to snooze it. Participants disliked that they have to unlock their phones before they can snooze notifications from the lock screen. P11 raised concerns regarding snoozing notifications because he feels a social pressure to answer messages quickly and when a message is snoozed the sender has to wait for a reply. Another participant was concerned about being overwhelmed when snoozing too many notifications during the day to the same time slot. She suggested creating an overview of all received notifications in the evening instead.

Reasons for Snoozing Notifications Participants mainly snoozed notifications to create reminders. For example, participants mentioned that they snoozed *SMS/IM* and *Email* notifications because they were at work and did not want to deal with personal notifications. P12 explained that she concentrated on work and wanted to receive a reminder afterward. This was a common theme in the interviews. Participants snoozed *SMS/IM* notifications because they wanted to avoid switching their current context. Snoozing *SMS/IM* and *Email* notifications was often mentioned in regard to not forgetting a task. Another reason for snoozing notifications was that participants were sometimes not in the mood to deal with the notifications when they received them. For instance, P11 explained that he received "20-30" instant messages from a group chat and wanted to read them later. Participants also had to do other things first before being able to react to notifications. P13 explained that he received a message about a meeting with friends and had to ask his wife before he could accept the meeting. Further, participants snoozed notifications on-the-go to deal with them at home. For instance, P11 mentioned that he received a *Social* notification, which he wanted to read at home on his desktop computer. P13 explained that he snoozed an app update notification because he wanted to install the updates at home using a Wi-Fi connection. P12 snoozed a *Game* notification not to miss an in-game reward.

Duration vs. Point-in-Time Six participants liked the *duration* method to snooze notifications, as it enables them to estimate when they will be able to deal with the notifications. P11 stated that she likes the method because it allows her to decide how long to snooze based on the current situation. Participants found the *duration* method better for short periods of time. P3 especially liked it for snoozing notifications to the same day. However, other participants stated that the *duration* method requires more cognitive effort. P2 explained that she thinks in times of the day and, therefore, she would have to calculate the time distance herself. Further, P7 mentioned he does not know if he has time to deal with the notification in a particular distance in time.

Eight participants stated that the *point-in-time* method supports their daily routines. For example, P7 explained that he usually knows when he can deal with notifications, e.g., in the lunch break or after work. Additionally, another participant explained that he does not have to think about how long it will take until he can deal with the notification, e.g., in the evening. Participants mentioned that they prefer the *point-in-time* method for longer distances in time, e.g., more than an hour (P10) or a week (P3). P9 stated that the pre-defined time slots are not always useful. Further, two participants found that entering a custom time is difficult when being busy. P11 mentioned that the *point-in-time* method requires a higher cognitive effort because he has to calculate the time when he wants to receive the notification.

Usefulness of *NHistory* Eleven participants stated that the app helps them to deal with notifications, four found it somewhat useful, and one participant found it to be not useful at all. Participants liked that the app enables them to have a "clean" notification drawer, and they found the notification history helpful. P4 especially liked that he can dismiss notifications without losing them. P3 liked that she can reflect on received notifications and was interested in how many notifications she receives.

Concerns and Suggestions Participants disliked the "snooze notification" that was shown when dismissing a notification. They suggested that, instead, the functionality should be added to the notification itself using a gesture or long-press action. Two participants would like to have the *duration* and *point-in-time* options available at the same time, and two other participants would like to customize the options. Participants disliked how multiple instant messages are grouped in notifications and snoozing a single conversation is sometimes not possible. P1 and P5 suggested that after snoozing a notification, all following notifications from the same app should be snoozed as well. Further, participants would like to snooze all notifications from all apps for a specific duration or a specific point-in-time. P15 suggested automatic rules to snooze notifications. Participants also suggested using location-based triggers instead of time-based ones. P10 wished that snoozing a notification would create a corresponding calendar event which then could be synchronized with other devices. Regarding the notification history, P4 suggested that "spammy" notifications should be automatically detected and excluded. P8 would like to see statistics about received notifications and P14 suggested an end-of-day summary of all received notifications.

4.2.5 Discussion and Limitations

We observed comparable usage patterns in both studies. Temporarily snoozing notifications was favored compared to permanently bookmarking them. Most snoozed notifications were of the *SMS/IM* category, followed by *Calendar/Reminder* (in-the-wild) and *Email* (controlled).

In both studies, we observed peaks in re-triggered notifications at 12pm, 3pm, 6pm, and 9pm. Additionally, in the in-the-wild study 9am was preferred. These times are likely influenced by the pre-defined options of *NHistory* but might also correlate with before and after work hours, and lunch breaks. In the controlled study, almost a quarter of the snoozed notifications were re-triggered at 6pm, likely correlating with after work hours. This was further strengthened in the interviews with many participants stating that they snoozed personal notifications at work or while studying. We observed that during the day notifications were typically deferred for short amounts of time. However, deferring notifications from the morning or noon to the evening, or from the evening until the next morning were also common use-cases. Another similarity of both studies was that the deferral of notifications was mostly limited to the same day. Few notifications were snoozed until the next day, and more than two days was an exception. Still, in both studies, we observed only a small fraction of received notifications being snoozed. We assume these notifications to be of high relevance to the users but, at the same time, out-of-context and with a low urgency.

A limitation of the in-the-wild study is that we did not collect demographic data from the users. While the Google Play Store statistics indicate a diverse set of users, we have little background information. We assume that the active users were, to a certain degree, tech-savvy, as they found, installed, and configured

NHistory without detailed instructions. In the controlled study, most participants had a technical background and were, therefore, tech-savvy as well. A second limitation is that we limited the data collection due to privacy concerns. This resulted in little knowledge about the notifications apart from which app issued them at what time. Future studies should consider context data and the notifications' content, e.g., using the approach proposed in Chapter 3.

While we conducted the studies, the latest version of Android was announced (March 21, 2017) and eventually released (August 21, 2017). In Android 8.0 ("Oreo") snoozing notifications was implemented natively. Swiping a notification from left to right unveils a "clock" icon. Tapping on the icon allows the user to snooze the notification for 15 minutes, 30 minutes, 1 hour, or 2 hours. This interaction is similar to the suggestions we received in the interviews. Comparing the provided options with *NHistory* and our findings, we notice the lack of long-term options for use-cases such as snoozing notifications until after work or the next morning. Since we concluded the controlled study on March 22, 2017, it is unlikely that the announcement of Android 8.0 influenced the participants. However, we assume that some users of the in-the-wild study read about the feature and subsequently searched and downloaded *NHistory* from the Google Play Store.

4.2.6 Design Implications

From the findings of the in-the-wild and controlled studies, we derived design implications for future notification systems.

4.2.6.1 Consider Context and Daily Routines

Participants in the interviews reported multiple reasons for deferring notifications. For instance, when the user is focused on another task (work, studying), the notification is out-of-context (personal notifications at work), when the user is unable to attend the notification (on the go, driving, sports), or simply when the user is not in the mood to take action on the notification. Users already apply various strategies in these cases, such as muting the phone, putting it away and revoking notification permissions. Future notification systems should follow and build on these strategies.

We observed peaks in deferred notifications to, what we assume to be, before and after work hours, and lunch breaks. Especially in the controlled study, many of the deferred notifications were re-triggered in the evening. Future notification systems should consider the user's context and daily routines. For example, personal notifications at work could be automatically be detected and subsequently deferred to after-work hours. This could be combined with the creation of automatic summaries. Although we offered multiple pre-defined options to snooze notifications, users made effective use of custom durations and points-in-time. Further, participants of the controlled study suggested personalization of these options, indicating that a one-fits-all approach might not be practical.

4.2.6.2 Balance Importance and Social Expectations

Notifications from the *SMS/IM* category were snoozed most often in both studies, highlighting their importance to users. Further important categories include *Calendar/Reminder*, *Email*, and *Social*; reiterating that "notifications are for messaging" and "important notifications are about people and events" [137]. Participants stated in the interviews that they often attend communication-related notifications immediately. Especially for instant messages, there are social expectations to reply quickly [23]. Future notification systems should carefully assess the importance of a notification to decide whether or not it should be deferred and for how long.

4.2.6.3 Mind the Ephemeral Nature of Notifications

We received positive feedback regarding the notification history, as it enables users to reflect on notifications. The history further allows users to "safely" dismiss notifications because they can always look them up afterward, reducing the number of pending notifications in the notification drawer. Still, participants regarded notifications as temporary. Notifications were mostly deferred to the same day or the next morning. Even for notifications that were snoozed multiple times, we observed that deferring a notification for more than two days was an exception. Developers of future notifications systems that defer notifications can use this as an upper bound. Within this time span, we observed a high variance regarding the deferral duration, depending on the time of the day and the users' daily routines.

4.3 Conclusion

In this chapter, we reported multiple studies on how to improve the management of notifications on mobile devices (RQ3).

In the first part of the chapter, we introduced the *Notification Dashboard*, a visualization for notification statistics to enable users to reflect on their own mobile notifications. We implemented the dashboard in two parts. First, we developed a logging application for Android smartphones that records all notifications in a log file. Afterward, we built the dashboard itself, a web application that visualizes the log files and allows users to explore it. To evaluate this first iteration of the dashboard, we conducted an interview study with 3 participants and showed them the visualizations of approximately one month of their own notifications. From these interviews, we derived opportunities for improvements.

In the second part of the chapter, we investigated the user-defined deferral of mobile notifications. To reduce negative effects caused by interruptions from notifications, a body of related work investigated using opportune moments for notification delivery, often based on breakpoints in the user's activity. Our approach instead allows users to manually "snooze" notifications. We focused on deriving insights into how users postpone notifications rather than eliminating negative effects directly, as snoozing a notification might even introduce a second interruption. We developed the Android app *NHistory* that extends the Android operating system in two ways. A notification history enables users to go back to previously dismissed notifications. Further, the app allows users to snooze notifications for a user-defined duration or to a user-defined point-in-time. We explored how and why users make use of the provided functionality in a year-long in-the-wild study with 295 active users and a week-long controlled study with 16 participants. Notifications of the categories *SMS/IM*, *Calendar/Reminder*, *Social*, and *Email*, were snoozed most often. Even for notifications that were snoozed multiple times, we observed that deferring a notification for more than two days was an exception. We conducted interviews to gain insights into when and why people find deferring notifications useful. Participants mentioned avoiding context switches, especially from attending personal notifications during work. As a result, we observed a number of notifications being snoozed to before and after work hours, as well as lunch breaks. Participants raised concerns regarding deferring communication-related notifications due to social expectations to respond as soon as possible. We concluded the chapter with design implications for future smart notification systems. These systems should consider different categories of notifications, such as personal and work-related notifications, the current context of the users, as well as their daily routines. Future work shpuld investigate snoozing notifications with user-defined rules [10], the generation of automatic notification summaries, and location-based triggers.

5

Notifications in Multi-Device Environments

In the previous chapters, we focused on notifications on smartphones. Smartphones are an important type of device, as they are typically always connected and always with the user. However, this is only a part of the bigger picture. When smartphones became popular, they joined the already existing PCs and laptops. Since then, we saw new types of devices becoming popular, such as tablets and smartwatches. In this chapter, we aim to answer the research question how various types of personal devices differ in multi-device environments in regards to receiving notifications. (RQ4). To answer this research question, we report the results of two studies. First, we conducted a quantitative study on notifications on four different types of devices. We logged the device usage of sixteen participants and correlated the usage with their location, number of people in their surroundings, the device proximity, and whether or not a device was suitable for receiving notifications in a certain context. We then complemented our findings by reporting a qualitative study on notifications in multi-device environments. We invited another sixteen participants and conducted semi-structured interviews about notifications on devices that they are using on a daily basis.

Parts of this chapter are based on the following publications:

D. Weber, A. Voit, P. Kratzer, and N. Henze. "In-situ Investigation of Notifications in Multi-device Environments." In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '16. Heidelberg, Germany: ACM, 2016, pp. 1259–1264. ISBN: 978-1-4503-4461-6. DOI: 10.1145/2971648.2971732

A. Voit, D. Weber, and N. Henze. "Qualitative Investigation of Multi-Device Notifications." In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. UbiComp '18. Singapore, Singapore: ACM, 2018, pp. 1263–1270. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274117

5.1 Quantitative Investigation of Notifications in Multi-Device Environments

Smart devices are becoming more and more ubiquitous. From desktop computers to laptops, smartphones, tablets, and smartwatches — always-connected devices have arrived in our everyday lives. One of the core features shared by "smart" devices is the ability to notify users about various events, such as new messages, or software updates. Depending on the application, notifications about an event can be shown on one or multiple devices at the same time. Figure 5.1 shows an exemplary multi-device scenario where a user is interacting with a laptop, while wearing a smartwatch, with a smartphone and tablet placed on the desk. Assuming every device in this scenario has an email client installed, a single email causes all four devices to notify the user. Disruptive effects caused by notifications are therefore amplified by the increasing number of devices around us. On the other hand, a text message on the smartphone might not be shown on other devices. This might prompt the user to pick up the smartphone and therefore interrupts the current work on the laptop.

Most prior research on notifications and interruptions only focused single types of devices in isolation. What is missing is an understanding of how future notification systems should be designed with multiple devices in mind. Therefore, we conducted a study to gain first insights about notifications in multi-device environments. We report a week-long Experience Sampling Method (ESM) study

Figure 5.1: An exemplary multi-device environment with a laptop, smartphone, tablet, and smartwatch.

with sixteen participants and four different types of smart devices. We analyze if the device proximity, interaction, and location are indicators for whether or not a device should be used to notify the user.

5.1.1 Study

To reduce the effect of interruptions caused by notifications, previous work focused mainly on the time to display notifications. While this is certainly important, the large number of devices, including PCs, smartphones, smartwatches, and tablets, suggests that the device that displays notifications is also important. Therefore, we conducted a study to investigate notifications in multi-device environments. In the following, we describe the design of the study, the used apparatus, the procedure, and the participants that took part.

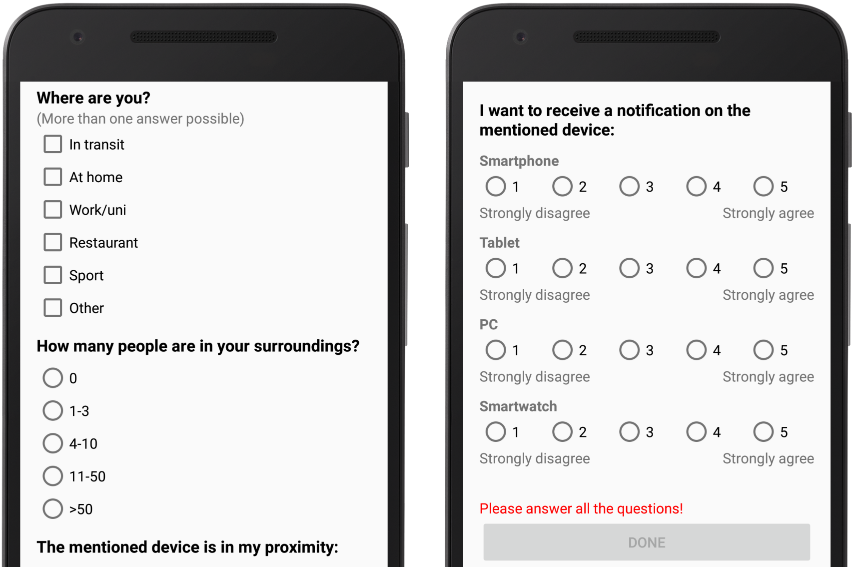


Figure 5.2: Screenshots of the ESM questionnaires that were triggered at random times of the day, consisting of two questions and two statements.

5.1.1.1 Design

In the study, participants used a smartphone, a tablet, a smartwatch, and a PC, the four most commonly used devices that are able to display notifications. Over the course of one week, we collected responses from participants using the Experience Sampling Method (ESM) [25, 28, 55]. To reduce interference with other tasks, we designed the ESM questionnaire in a way that allows completing it without any text input. The questionnaire consisted of two questions and two statements (see Figure 5.2):

- **Q1: Where are you?** Possible answers were *In transit*, *At home*, *Work/uni*, *Restaurant*, *Sport*, and *Other*. When selecting *Other* an optional text field appeared. Participants could select multiple answers to allow combinations, such as working on a train or doing a workout at home.

- **Q2: How many people are in your surroundings?** Possible answers were *“0”*, *“1-3”*, *“4-10”*, *“11-50”*, and *“more than 50”*. Here only one answer could be selected.
- **Q3: The mentioned device is in my proximity.** Followed by a 5-point Likert scale item for each device (smartphone, tablet, PC, smartwatch).
- **Q4: I want to receive a notification on the mentioned device.** Followed by a 5-point Likert scale item for each device (smartphone, tablet, PC, smartwatch).

Participants received EUR 0.20 at the end of the study for each completed questionnaire. In addition to the ESM responses, we recorded activity data on each of the participants’ four devices. For example, we recorded screen-on events and if the user is interacting with the device.

5.1.1.2 Apparatus

To not interfere with the participants’ device usage, we used a dedicated device to present the ESM questionnaires. All participants were equipped with an additional smartphone for the sole purpose of collecting ESM responses. Therefore, we implemented an ESM survey Android application that consisted of a background service and the survey view itself. The background service triggered a survey every 45 to 90 minutes. Between 0am and 6am, no surveys were triggered. We asked the participants to carry the ESM device with them during the active times, and told them that they are free to change the notification volume and to disable the vibration. When a survey was triggered, a survey notification was shown and clicking it opened the survey view. If the notification was not clicked within 10 minutes, it was removed. The ESM answers were stored locally on the ESM device. For the study, we disabled all other apps and data connections on the ESM device, resulting in a battery life of over one week. Therefore, participants could carry the ESM device for the entire duration of the study without having to charge it.

We implemented logging applications for Android devices (smartphone, tablet, smartwatch) and Windows PCs to record the status of each of the four devices. The Android application consisted of a background service, set to “high priority” to avoid termination by the Android system. Because we were concerned about causing a noticeable impact on the battery, which could influence the study results, the application only collected event-based information. The collected events were *display on/off*, *connection status (Wi-Fi/mobile/offline)*, *power on/off*, *headset connected*, and *charging/not charging*. In addition, the app recorded *touch events* using a transparent layer above other apps. This, too, was limited and we only logged one touch event per minute. We used Google Play services[1] APIs to record the current *location* and the current *activity*. The Activity Recognition[2] API returned a probability for the activities *foot*, *bicycle*, *still*, *running*, *tilting*, *walking*, and *unknown* without negatively affecting the device's battery life. The logging application worked on all devices with Android 4.3 (or newer) including smartphones, tablets, and smartwatches.

The Windows application also consisted of a background process and automatically added an entry to the Windows auto start. The application recorded *log-in/log-out* events and *times of inactivity*. *Times of inactivity* were calculated similarly to a screen-saver. Once a minute, the application checked if the user had interacted with the computer by either moving the mouse or typing on the keyboard. If there was no interaction, the inactivity was logged with a timestamp and another timestamp was logged once the mouse was moved again or something was typed on the keyboard. We also recorded the name of the current foreground process to detect, for example, when a video was being watched and, therefore, no interaction happened. Figure 5.3 provides an overview of the data that was collected in the study.

5.1.1.3 Procedure

To capture weekdays and weekends, the duration of the study was 7 full days for each participant. On the day before the start of the study, we invited participants to sign a consent form and to fill a demographic questionnaire. We also gave them a smartwatch and the additional ESM device, and explained how to use them. If the participant did not own a tablet, we also handed out a tablet. Because all participants owned Android smartphones, they were already familiar with the

[1] `https://developers.google.com/android/guides/overview`
[2] `https://developers.google.com/location-context/activity-recognition`

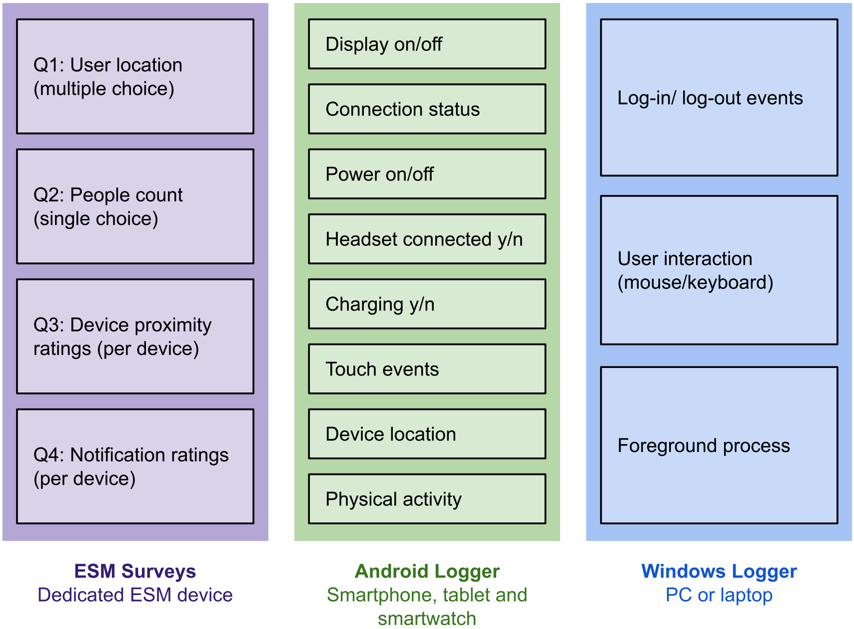


Figure 5.3: Overview of the data that was collected in the study.

operating system on the tablets. We installed the logging applications together with the participants and explained in detail what information is recorded and that they should use all devices as usual. The day after the study, we again invited participants to export the locally stored data, retrieve the devices, and hand out the monetary rewards depending on how many ESM questionnaires were completed. This resulted in a total participation time of nine days.

5.1.1.4 Participants

We recruited participants using a university mailing list by describing the study and stating that we are looking for participants with an Android 4.3+ smartphone and a Windows PC or laptop. We also stated that owning a tablet is preferred but not required. In total, 18 people participated in the study. However, we excluded two participants. In the first case, exporting the log file from the smartwatch failed, resulting in an incomplete set of log files. In the second case, the participant only answered one ESM questionnaire in seven days. Of the remaining 16 participants, 4 were female and 12 male. They were between 19 and 58 years old ($M = 26.25, SD = 8.76$). Eleven participants were students, four employees, and one a retiree. All participants used their own smartphone and PC. Ten participants used their own tablet, and we handed out six tablets. Of the ten participants who used their own tablet, two shared a tablet with their partner but used different profiles. Furthermore, we handed out smartwatches (Motorola Moto 360) and the additional ESM devices (Samsung Nexus S) to all participants. Only one participant used a smartwatch before.

5.1.2 Results

At the end of the study, we collected the ESM responses and the automatically recorded data. In the following, we provide an overview of the collected ESM responses. We investigated if the participants' location, the proximity of the device, and the number of people in their surroundings had an effect on participants' preference for receiving notifications on the four devices. Subsequently, we investigated the effect of device usage on participants' preference. Finally, we analyzed the correlation between the answered questions in the questionnaire (Q1, Q2, and Q3), the corresponding logged events, and the preferred device to receive an incoming notification (Q4).

5.1.2.1 Analysis of the ESM Questionnaires

Participants answered between 14 and 90 ESM questionnaires ($M = 60.31, SD = 21.26$) which totals to 965 answered questionnaires. On working days, more questionnaires were answered than on the weekend. However, even on Sundays, the day with the lowest amount of answered questionnaires, more than 110 questionnaires were answered. Figure 5.4 shows the total number of answered questionnaires for each hour of the day between 6am and 0am (the time the ESM questionnaires were triggered). The number of answered questionnaires increased as the day progressed, with the highest number of answers between 1pm and 2pm. A second peak can be seen between 10pm and 11pm.

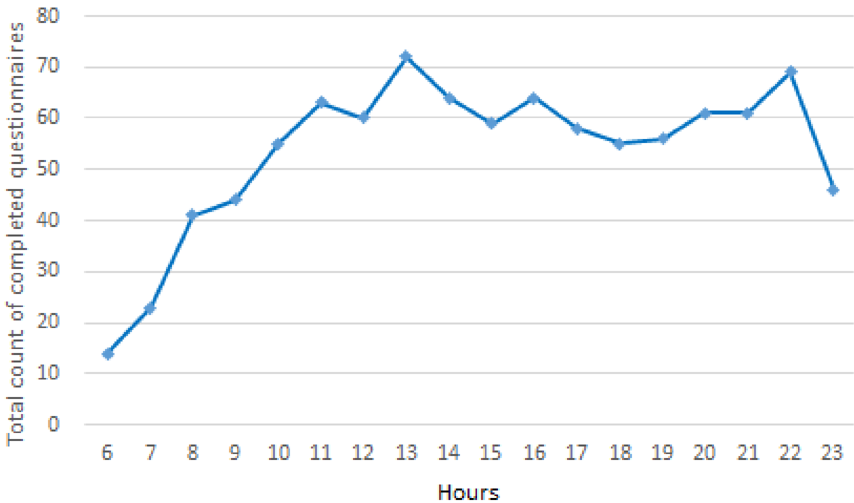


Figure 5.4: Total number of completed ESM questionnaires from all participants for each hour of the day. Between 0am and 6am no questionnaires were triggered.

According to the participants' recorded locations (Q1), we found that the participants were mostly at *home* (70.55%), followed by *work/uni* (14.01%), *in transit* (11.04%), *other* (2.66%), *restaurant* (0.92%), and *sport* (0.82%). One participant mentioned that he did not carry any device when working out and, therefore, might have missed questionnaires. According to the second question (Q2), most of the time participants were either with "1-3" other people (47.46%) or alone (35.54%), followed by "4-10" people people (10.78%), "11-50" (4.77%), and "more than 50" people (1.45%).

We used a Friedman test to investigate if the proximity of the devices significantly differs between devices (Q3, see Figure 5.5). We used Wilcoxon signed-rank post-hoc tests with Bonferroni correction (resulting in a significance level of $p \leq 0.008$) for pairwise comparison. We found that the proximities of the four devices significantly differ, $\chi^2(3) = 19.390$, $p < .001$. The device closest to the participants was the smartphone ($M = 4.31, SD = 0.60$) and smartwatch ($M = 4.31, SD = 0.79$), followed PC ($M = 3.53, SD = 1.38$), and tablet ($M = 2.69, SD = 1.20$). The smartphone is significantly closer than the tablet

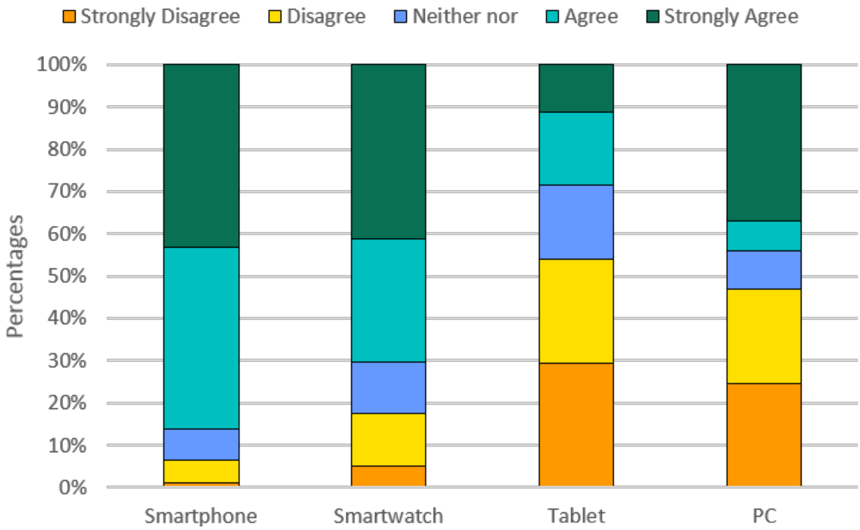


Figure 5.5: Agreements to the statement "*The mentioned device is in my proximity*" (Q3) for smartphone, smartwatch, tablet, and PC.

($U = -3.22, p = 0.001$). Similarly, the smartwatch is significantly closer to the participant than the tablet ($U = -3.09, p = 0.002$). There are no significant differences for all other combinations, $p \geq 0.035$.

We again used a Friedman test and Wilcoxon signed-rank tests to investigate the preferred device for receiving notifications (Q4, see Figure 5.6). We found that the device has a significant effect on participants' preference, $\chi^2(3) = 21.401$, $p < .001$. The most preferred device to receive notifications is the smartphone ($M = 4.12, SD = 1.26$), followed by the smartwatch ($M = 3.69, SD = 1.40$), the PC ($M = 2.56, SD = 1.31$), and the tablet ($M = 1.63, SD = 0.81$). Participants rated the smartphone significantly higher than the tablet ($U = -3.33, p = 0.001$) and the PC ($U = -2.66, p = 0.008$). The rating for the smartwatch is significant higher than the tablet ($U = -3.10, p = 0.002$). There are no significant differences for all other combinations, $p \geq 0.055$.

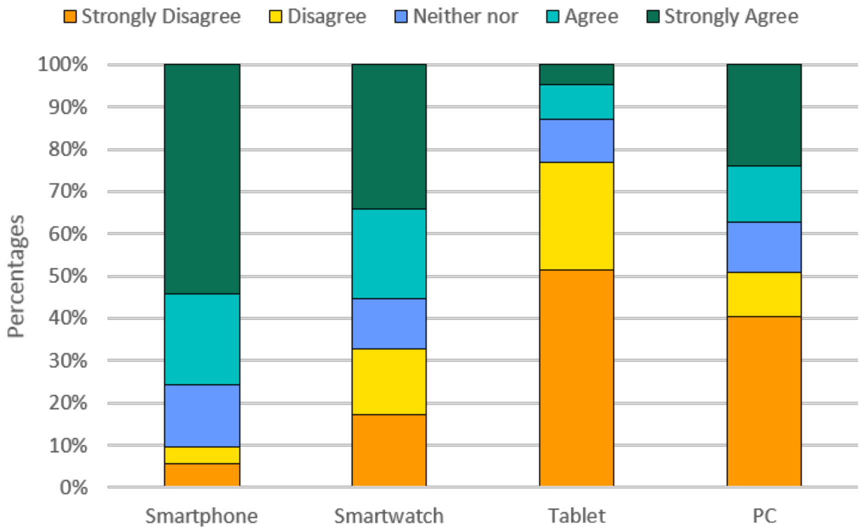


Figure 5.6: Agreements to the statement "*I want to receive a notification on the mentioned device*" (Q4) for smartphone, smartwatch, tablet and PC.

5.1.2.2 Analysis of the Device Usage

Regarding the device usage, the device with the most screen-on events per day (see Figure 5.7a) was the smartwatch ($M = 120.57, SD = 87.79$), followed by the smartphone ($M = 73.25, SD = 43.74$), and the tablet ($M = 7.35, SD = 9.38$). The device with the most touch events per day (Figure 5.7b) was the smartphone ($M = 98.39, SD = 84.89$), followed by the tablet ($M = 20.56, SD = 40.69$), and the smartwatch ($M = 20.51, SD = 19.14$). The device with the highest average active time per day (see Figure 5.7c) was the PC ($M = 4:32h, SD = 3:48h$), followed by the smartphone ($M = 1:50h, SD = 1:38h$), the tablet ($M = 0:39h, SD = 1:03h$), and the smartwatch ($M = 0:17h, SD = 0:14h$). For the PC, the active time was calculated as the time between logging in and out minus the time without user interaction. For Android devices, the active time was calculated as the time the screen was on.

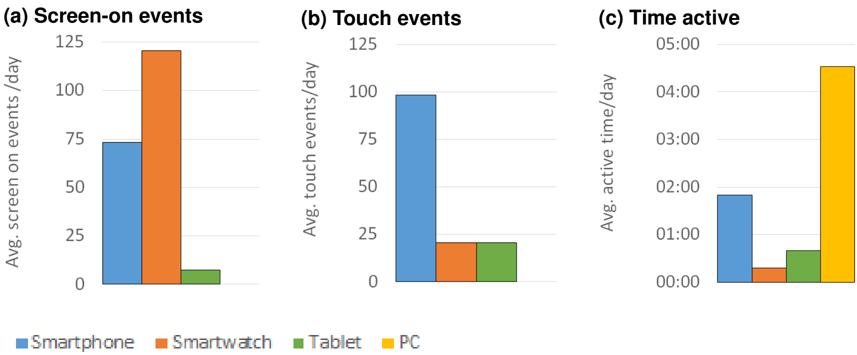


Figure 5.7: Average daily number of screen-on events, touch events, and active time.

5.1.2.3 Correlations

We analyzed the correlations between the device proximity (Q3) and whether participants want to be notified on the device or not (Q4) (see Figure 5.8). First, we calculated the correlation coefficient r for the proximity to the devices, and if the participants want to be notified on the devices for every participant and every device. Then we calculated the average correlation for all participants for the four devices. For a better overview, we only report average effect sizes of $r > \pm 0.1$. Using Cohen's conventions [24] to describe the effect size, for all devices, we found moderate to strong positive correlations between the device proximity and whether or not notifications should be shown on the device. We found the strongest correlation for PC ($M = 0.73, SD = 0.21$), followed by smartwatch ($M = 0.63, SD = 0.30$), tablet ($M = 0.61, SD = 0.24$), and smartphone ($M = 0.45, SD = 0.26$).

Furthermore, we calculated the correlations of participants' location (Q1) and Q4. For *in transit*, we found weak negative correlations for PC ($M = -0.27, SD = 0.17$) and tablet ($M = -0.22, SD = 0.20$). For *at home*, we found weak to moderate positive correlations for tablet ($M = 0.30, SD = 0.33$) and PC ($M = 0.26, SD = 0.31$), and weak to moderate negative correlations for smartwatch ($M = -0.25, SD = 0.35$) and smartphone ($M = -0.09, SD = 0.23$). For *at*

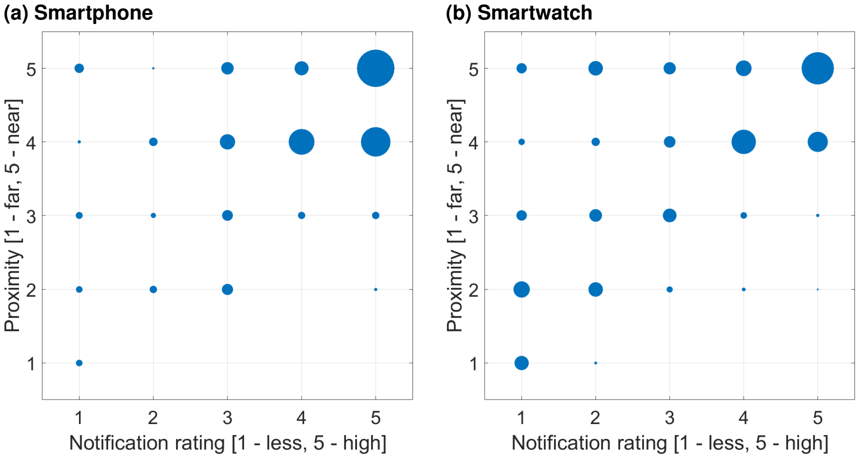


Figure 5.8: Correlations between the proximity to the devices and if the participants want to receive notifications on the devices. The size of the points in the scatter plots represents the frequency of occurrence of the single, normalized ratings.

work/uni we found a weak positive correlation for smartwatch ($M = 0.26, SD = 0.23$) and a weak negative correlation for tablet ($M = -0.18, SD = 0.30$). *Restaurant* and *sport* were not selected often enough for meaningful results.

We calculated the correlations of the number of people in participants' surrounding (Q2) and Q4. When alone, we found weak positive correlations for the PC ($M = 0.26, 0.15$) and tablet ($M = 0.13, SD = 0.22$). When with "4-10" people, we found weak negative correlations for PC ($M = -0.17, SD = 0.23$) and tablet ($M = -0.14, SD = 0.18$). When with "11-50" people, we found weak positive correlations for smartphone ($M = 0.10, SD = 0.11$) and smartwatch ($M = 0.14, SD = 0.26$), and weak negative correlations for PC ($M = -0.14, SD = 0.14$) and tablet ($M = -0.13, SD = 0.14$). "More than 50" was not selected often enough for meaningful results.

We also calculated the correlations between screen-on events right before or after a questionnaire was triggered and Q4. We found moderate positive correlations for smartwatch ($M = 0.34, SD = 0.24$) and tablet ($M = 0.31, SD = 0.28$), and a weak correlation for smartphone ($M = 0.18, SD = 0.20$). We did not log screen-on events for the PC. We also calculated the correlations between whether the devices were *still* (using the Activity Recognition API) and Q4. We found weak negative correlations for smartwatch ($M = -0.29, SD = 0.30$), tablet ($M = -0.27, SD = 0.25$), and smartphone ($M = -0.13, SD = 0.24$). Again, the PC is excluded because no activity recognition events were logged. We also calculated the correlations between the active time and Q4. We found a strong positive correlation for PC ($M = 0.51, SD = 0.20$), a moderate positive correlation for the tablet ($M = 0.37, SD = 0.32$), and weak positive correlations for smartwatch ($M = 0.20, SD = 0.12$), and smartphone ($M = 0.18, SD = 0.20$).

5.1.3 Discussion and Limitations

We conducted an ESM study with 16 participants and 4 different types of smart devices for 7 days. All participants used their own smartphones and PCs. Ten participants also used their own tablets. We handed out smartwatches for all participants. Although all participants were used to the Android platform, in the future, we plan to investigate if longer device usage has an influence on the preferred notification location.

On average, participants preferred to be notified on the smartphone, followed by the smartwatch, the PC, and the tablet. The smartwatch ranking second is interesting, because only one participant had used a smartwatch before. Comparing the device usage of smartphones and smartwatches, we saw more touch events on smartphones but more screen-on events on smartwatches. This is likely because the screen of the smartwatch turns on automatically when tilting the device.

We found that the device proximity influences whether or not the user wants to be notified on the device. To an extent, this finding seems obvious, as notifications will not be noticed when the device is not near the user. Regardless, this is something that should be considered when creating future multi-device-aware notification systems. Past research investigated the possibility of inferring where phones are kept [191], work which should be extended to other devices. For the smartphone, the correlation was only moderate, but this can be attributed to the fact that the smartphone was almost always with the participants. Further, the participants preferred to receive notifications on devices which have an activated screen and they are currently interacting with. On the other hand, *still* devices are less suitable for notifications. Regarding the user's current location, PC and tablet both showed negative correlations for *in transit* but positive for *at home*. At *work/uni* the smartwatch was favored.

To keep the questionnaire simple, we purposely did not specify details about the incoming notification in Q4. In future research, the type and content of notifications should be considered by, for example, conducting interviews. Furthermore, notifications might be device-specific (e.g., available updates) or independent (e.g., email, messaging). We also did not address which modalities should be used to notify the user, which is another important aspect of future research.

5.1.4 Summary

In this first part of the chapter, we investigated notifications in multi-device environments. We conducted a week-long in-situ study using the Experience Sampling Method (ESM) with sixteen participants and four different types of smart devices (smartphone, smartwatch, tablet, and PC). Apart from ESM answers, we also collected device usage data, such as screen-on events, touch events, and whether or not the device has been moved lately. Disregarding the type or content of notifications, we found that the smartphone is the preferred device on which to be notified on, followed by the smartwatch, PC, and tablet. Further, we found that the proximity to the device, whether the device is currently being used and the user's current location can be used to predict if the user wants to receive notifications on a device. These findings provide first insights for the design of future multi-device-aware smart notification systems. A limitation of this study was that we focused on quantitative data and did not collect any qualitative data. To overcome this limitation, we conducted an interview study with another sixteen participants.

5.2 Qualitative Investigation of Notifications in Multi-Device Environments

In the second part of this chapter, we investigate how users cope with the notifications on different devices in their everyday lives. Therefore, we conducted interviews and investigated especially strategies developed by the users how to deal with notifications on their different devices. Furthermore, we were interested in how users use the offered configuration options for notifications on their devices. Our results show that users developed similar strategies to deal with unwanted notifications on their different devices. Furthermore, few users are changing the notification settings on their devices.

5.2.1 Interviews

We conducted a qualitative study with sixteen participants to investigate how participants experience and deal with notifications from different device types. We, therefore, invited the participants to our lab and conducted semi-structured interviews. When the participants arrived, we asked them to fill out a consent form and asked them to provide demographic data. For the interviews, one researcher lead the interview, another researcher took notes, and a third researcher supervised the procedure. The interviews were structured into two parts. In the first part, we

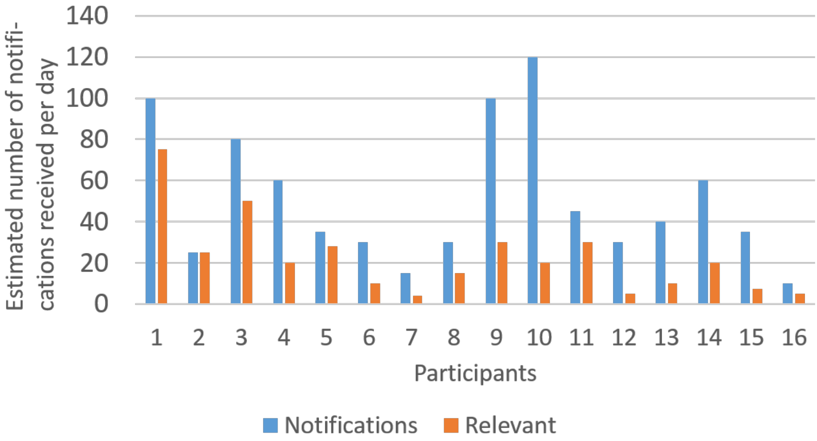


Figure 5.9: The number of all and relevant-only notifications participants receive on their devices on a daily basis. All values are estimates by the participants.

asked the participants about their experience on notifications in general. In the second part of the interview, we asked them about their experience depending on the different device types.

5.2.1.1 Participants

In total, we interviewed 16 participants (8 female, 8 male) that were between 16 and 60 years old ($M = 30.94, SD = 15.11$). Participants had diverse backgrounds; none had a computer science background. Six participants were students of various subjects, four employees, two high school students, two retirees, and two trainees.

5.2.1.2 Part 1: General Questions

For the general questions about notifications, we asked the participants on which devices they receive notifications in their daily lives as well as how many notifications they receive from these devices on a daily basis. All participants owned a smartphone as well as a laptop or desktop computer. Furthermore, six participants owned a gaming console, five a tablet computer, three a smart TV, three a TV set-top box, three an ebook reader, two a smart car, and two a fitness tracker. One participant also owned a traditional mobile phone. The participants estimated that they receive between 10 and 120 notifications per day ($M = 51.56, SD = 32.65$). All estimations are shown in Figure 5.9. We further asked the participants to estimate how many notifications they consider relevant. Here, the answers ranged from 4 to 75 relevant notifications per day ($M = 22.66, SD = 18.94$). Only P2 stated that she considers all notifications that she receives relevant.

We investigated how our participants experience different kinds of notifications. Eleven participants noted that they consider notifications as being useful when they are related to communication. Notifications about calendar events, reminders, and alarms were also mentioned as being useful by four participants. In contrast, system- and security-related notifications were only found useful by two participants. Furthermore, P4 stated that news notifications are relevant. Two participants stated that the usefulness depends on the content of the notifications itself and not on the category. Another two participants mentioned that they consider all notifications as useful.

In addition, our participants reported that they consider notifications as disturbing when they receive them at night (6 participants) and at work/university/school (5 participants). Also, the participants mentioned that receiving notifications is also disturbing in inappropriate situations such as during meetings and appointments (3), when generally being busy (2), while they are talking to others (1), when other people are around them (1), while driving (1), during sport (1), or even while being in a bad mood (1). Furthermore, participants disliked notifications that are mainly used as ads, for example, when lesser-used apps try to grab their attention to keep them in the loop (3). Three participants stated their dislike of "spammy" notifications, and two other participants mentioned "spammy" messages (e.g., group chats from instant messaging). Two participants each disliked notifications from games and update notifications. Further, two participants disliked notifications that are delivered with sounds, and one participant mentioned that visual notifications are sometimes unwanted. One participant disliked the fact that, in some cases, notifications cannot be disabled. Other participants disliked receiving duplicate notifications.

5.2.1.3 Part 2: Device-Specific Questions

In the second part of the interview, we asked the participants specifically about their usage of their devices and how they cope with notifications on their devices.

Smartphones Thirteen participants owned an Android-based smartphone, and the other three participants owned Apple iPhones. We asked the participants to estimate their daily smartphone usage, which resulted in an estimated average usage time of 3.61 hours ($SD = 2.82$) per day. For Android smartphones, the participants estimated that they receive between 5 and 80 notifications ($M = 36.15, SD = 16.67$) per day. They considered between 3 and 50 notifications ($M = 14.54, SD = 5.77$) as relevant. Ten participants stated that they check incoming notifications immediately or within a short time span. Only three participants mentioned that they silent their phones (2) or put their phone away (1) and check the received notifications manually from time to time. P15 explained that he checks notifications displayed on the lock-screen and attends to them if the content is interesting for him. Furthermore, four participants reported that they keep notifications in the notification drawer to attend or answer them later. Two participants stated that they dismiss received notifications immediately to keep the notification drawer clean. However, P12 noted that he attends the content of dismissed notifications later to react to them. If an app triggers mainly notifications that are experienced as unwanted, five participants stated that they dismiss these unwanted notifications without changing the notification settings (i.e., they neither deactivate notifications for this app nor uninstall the application). P12 explained that he is aware that Android offers the opportunity to disable the notifications for specific apps, but he is not using this opportunity as dismissing them is less effort. However, five participants mentioned that they remove the permission for such apps to trigger notifications. Furthermore, four participants using Android reported that they uninstall such an application directly. After we explained how to revoke the notification permission for apps on Android, three participants reinstalled apps and disabled notifications for them. Regarding iPhones, the participants estimated to receive between 20 and 50 notifications ($M = 36.67, SD = 15.27$) per day of which 20 to 30 notifications $M = 26.67, SD = 5.77$) are relevant. All iPhone users stated that they are attending incoming notifications not immediately, but as soon as possible. Similar to the Android users, one participant using iOS mentioned to dismiss notifications from an app that generates unwanted notifications instead of changing the notification settings. Another participant mentioned that she revokes the notification permission for such an app, and one participant uninstalls such applications.

PCs All participants owned a desktop computer or laptop with Microsoft Windows. Participants estimated spending between 1 and 12 hours per day in front of the PC ($M = 4.91, SD = 3.49$). Besides, participants estimated that they receive between 0 and 40 notifications ($M = 15.16, SD = 11.69$) per day. From this number of notifications, our participants estimated between 0 and 25 notifications ($M = 5,72, SD = 6,42$) as relevant. Three participants considered all notifications received on their desktop computers or laptops as useful. Twelve participants mentioned that they perceive such notifications but usually do not attend to them if they see no need to react on notifications; e.g., about available updates or Wi-Fi connections. Nine participants stated that they attend only to notifications generated by specific applications, e.g., by email, calendar, or instant messaging applications. Four participants stated that they dismiss notification without reading their content. Regarding unwanted notifications, six participants reported that they ignore them until they disappear automatically, eight participants reported that they dismiss unwanted notifications, and two participants uninstall applications generating unwanted notifications. None of the participants changed the default settings.

Tablets Five participants owned tablets. Two Android-based tablets (Android 4.3 and 5.1) and three Apple iPads (iOS 9.3.1). Participants estimated to use their tablets between 15 and 120 minutes per day ($M = 51, SD = 41.89$). All participants mentioned that they receive few notifications and the ones they receive are of little importance. On the Android tablets, both participants estimated to receive 2 and 20 notifications per day, of which 0 and 1 notifications ($M = 0.50, SD = 0.71$) are relevant. They mention that most notifications are from games or duplicate email notifications that are already received on other devices. Unwanted notifications are mostly tolerated and sometimes dismissed. No Android tablet user changed the default settings. One iPad user estimated to receive 10 notifications, and the other two iPad users mentioned that the amount is similar to smartphone notifications since they are synced. P14 mentioned that most notifications are from the email app since he did not grant the notification permission to most other apps. However, he ignores email notifications most of the time. He also did not change the default settings but disables the Wi-Fi connection at night.

Smart TVs Two participants enhance their TV with an Apple TV. Both participants stated that they only use the Apple TV on weekends, with an estimated usage time of 2 hours. The only notifications shown are about system updates.

Gaming consoles Four participants owned gaming consoles. Two participants owned a Sony PlayStation 3 and two a PlayStation 4. They estimated a daily usage time between 10 and 120 minutes ($M = 47.50, SD = 49.24$). Participants estimated that they receive between 0.5 and 10 notifications per day ($M = 4.38, SD = 4.19$). Notifications are typically about low battery warnings for the controller, online/offline status changes of contacts, and system updates. The participants mentioned that they ignore all notifications except the low battery warnings.

Ebook readers Only one participant shared her experience with an ebook reader. She estimated a daily usage of 1.5 hours. The only notifications she receives from her ebook reader are warnings about low battery, which she described as useful and typically acts upon immediately.

5.2.2 Discussion

People are surrounded by notifications from different types of devices. The smartphone is the primary notification device for all our participants. The smartphone always turned on and always being with the users means that users can be reached all the time. We heard subtle differences in how the participants describe their dealing with notifications. For instance, Android users mentioned disabling the permission to allow notifications while participants with iPhones mentioned not granting the notification permission in the first place. Since notifications are often used to engage users, it is to be expected that more and more apps make use of notifications on different devices. Our investigation of how users deal with notification showed that few users configure their notification settings or are aware of the configuration options. Therefore, the default configuration of notification settings is an essential challenge for the diverse device types supporting notifications as well as for future notification management systems. Making notifications opt-in instead of opt-out might be a useful first step to manage the increasing number of notifications. Even if users are aware of the offered notification settings, some are not changing their settings for individual apps since this is perceived as more effort than dismissing unwanted notifications. Thus, future devices supporting notifications should also offer options to change the notification settings with less effort.

5.2.3 Summary

In the second part of this chapter, we reported on a qualitative study where we investigated how users perceive and deal with incoming notifications on multiple device types in their daily lives. The discussions with the participants highlighted how more and more devices in the environment could notify the users. The smartphone and the PC were the predominant types of devices in the study, as all participants owned them. One participant owned nine different types of devices that are notifying him. On average, participants estimated that they receive 51.56 ($SD = 32.65$) notifications per day. Strategies to reduce distracting effects of notifications include, disabling (or not enabling) notifications, uninstalling applications, using do-not-disturb functionality, muting devices, or even putting devices in other rooms. However, few users, are configuring the notification settings of their devices – even if they are aware of the offered options such as revoking the permission to generate notifications for certain apps. With more and more devices notifying users, it is necessary to avoid overloading users with notifications. Decisions taken by device manufacturers can drastically change how notifications affect users, e.g., when comparing the opt-out approach of Android with the opt-in approach of iOS. Manually configuration of notifications on all devices might not be feasible in the future anymore, considering the increase of notifications and devices. This will especially become true with the Internet of

Things (IoT). Therefore, identifying the right default notification settings for the different device types is an essential challenge for the device manufacturers in the future.

5.3 Conclusion

In this chapter, we addressed the research question how various types of personal devices differ in multi-device environments in regard to receiving notifications (RQ4). While most prior work focused on notifications on single devices in isolation, we looked at notifications in multi-device environments by conducting quantitative and qualitative studies with sixteen participants each.

In both studies, the smartphone was shown as the most important device to be notified on. All participants owned smartphones, they are always connected, and always with the user. It was considered a suitable all-around device to target for notifications. An interesting type of devices was the smartwatch. None of the participants in the studies owned a smartwatch. For the first study, we handed out smartwatches. Participants who were comfortable with wearing a watch enjoyed the "notification experience," resulting in similar ratings as smartphones. Nowadays, smartwatches are still mostly an extension of smartphones; however, there is a trend towards standalone smartwatches. Both smartphones and smartwatches were rated as suitable devices to be notified on almost during the whole day. In contrast, PCs and laptops also received high ratings but only when in use. Using these types of devices to notify users while in use might reduce device-switching when triggering notifications. An interesting finding in both studies was that tablet computers were not rated as suitable devices to be notified on. Tablets were considered as media-consumption and gaming devices that often stayed at home. Participants mentioned that notifications on tablets were often duplicates that they already saw on other devices, such as their smartphones, or that they did not consider as relevant. Although the tablets used in the studies could be considered smartphones with a larger screen, participants were clear that smartphones are the primary notification devices and tablets are the opposite. Other devices addressed in this chapter were mostly limited to device-dependent notifications, such as low-battery warnings and update alerts. With the exception of low-battery warnings, participants mostly ignored those notifications. Ignoring notifications was a trend that spanned across all devices. Participants mentioned that they rather ignore notifications than spending time configuring them.

Our conclusion for this chapter is that future notification management systems should consider not only the timing of notifications, but also the device the notification is shown on. Designers and developers of future smart notification management systems should consider if a notification is device-dependent or device-independent. Device-dependent notifications could simply be shown on the device itself. For device-independent notifications, one should focus on the last-used devices with currently active devices as the primary target for notifications. Further, consider the user's context, such as location, activity, and number of people around the user. For example, while on-the-go consider posting notifications to a device that allows the user to glance at the notification quickly.

Implementing cross-device notifications also poses a number of technical challenges, such as uninterrupted connections and a shared understanding of the devices' states. To target specific devices or a specific subset of a user's devices with notifications requires a reliable cross-device notification implementation to avoid triggering the fear of missing out [9] by accidentally omitting notifications. A strategy that could be used is broadcasting important and urgent notifications to all devices, and reducing the number of targeted devices with decreasing importance and urgency.

It is important to note that in this chapter, we did not cover all possible device types. Further work should carefully access the properties of smart devices that were not considered in this chapter, e.g., smart speakers, or that differ, e.g., smart fitness trackers compared to smartwatches. As multi-device environments grow with new and deviated types of devices, the results should be reevaluated.

Finally, in this chapter, we mostly considered *personal* devices. A small exception was that some participants shared tablets with their partners. In the following chapter, we will look further into devices that are designed to be shared with multiple users.

6 Notifications on Large and Pervasive Displays

In the previous chapter, we compared different types of devices with regard to receiving notifications. This chapter expands the scope to include large and pervasive displays, such as smart TVs and public displays.

In the first part of the chapter, we explore considerations for displaying notifications on smart TVs (RQ5). We report the results of focus groups, an online survey, and a concluding lab study. Based on the results, we derive design guidelines for notifications on smart TVs. In the second part of the chapter, we explore considerations for displaying notifications on public displays (RQ6). We describe the system architecture and the results of an in-situ study with subsequent semi-structured interviews.

Parts of this chapter are based on the following publications:

D. Weber, S. Mayer, A. Voit, R. Ventura Fierro, and N. Henze. “Design Guidelines for Notifications on Smart TVs.” In: *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video*. TVX ’16. Chicago, Illinois, USA: ACM,

2016, pp. 13–24. ISBN: 978-1-4503-4067-0. DOI: 10.1145/2932206.2932212

D. Weber, A. Voit, G. Kollotzek, L. van der Vekens, M. Hepting, F. Alt, and N. Henze. "PD Notify: Investigating Personal Content on Public Displays." In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI EA '18. Montreal QC, Canada: ACM, 2018, LBW014:1–LBW014:6. ISBN: 978-1-4503-5621-3. DOI: 10.1145/3170427.3188475

6.1 Notifications on Smart TVs

Today's mobile devices and traditional desktop computers inform about new messages, upcoming appointments, events, and general hints using notifications. Notifications are a well-established mechanism to inform a user about a diverse range of information. One of the main use cases is enabling asynchronous communication. A typical notification related to personal communication on all major platforms informs about the sender and shows a text excerpt. In recent years, notifications became one of the core mechanisms on a number of smart devices.

Notifications can provide time-sensitive information. However, they do not always reach the user in time because the device is not in the user's range. For example, Dey et al. [32], showed that users' smartphones are only within arm's reach 53% of the time. Already in 2002, Want et al. [170] proposed to distribute notifications across different smart devices. Sahami Shirazi et al. [137, 176] developed a system that forwards smartphone notifications to desktop computers. Recently, major smartphone platforms started to provide centralized notification mechanisms. Notifications are not only managed on a single device itself but collected and shared across smartphones, tablets, desktop computers, and laptops. Furthermore, a number of new types of smart devices recently became available. The core feature of smartwatches and smart glasses is displaying notifications. Studying smartwatch users, Lyons [86], however, found that 24% of the 50 participants did not wear their watches at home.

Another highly successful type of smart devices are smart TVs. The main characteristic of smart TVs in comparison to regular TVs is the capability to process data and to connect with online services. Thus it is possible to stream videos and other content from the Internet. Unlike mobile operating systems, there is currently no dominant operating systems for TVs. There is, however, a clear trend towards platforms similar to mobile operating systems, including the possibility to extend the systems by installing apps from app stores. In contrast to other smart devices, current smart TVs have no established notification mechanisms. Displaying notifications on smart TVs poses a number of challenges. TVs are primarily used for watching content, including TV series, news, and movies. Displaying notifications on top of the main content can result in distractions and therefore affect the TV experience. Furthermore, unlike smartphones or smartwatches, TVs are shared devices that are used by multiple people, often at the same time. Therefore, the notification mechanisms designed for other smart devices cannot directly be adopted for smart TVs. Instead, it has to be investigated how a pleasant notification experience on all devices can be achieved while respecting the users' attention and privacy.

6.1.1 Related Work

The main characteristic of smart devices is the ability to connect to other smart devices and the Internet. In the past years, existing devices and everyday things got smarter. With mobile data networks, it is possible to access the Internet on the go, and with smartphones it can be carried in the pocket. Smartwatches and smart glasses extend smartphones and are always with the user. Smart TVs are able to stream content from the network, thus transforming the TV from a device that was used mainly for watching television to a large screen that is able to receive content from various sources. The connectivity of smart devices allows pushing messages to the devices, which lays the foundation for notifications. In the previous chapters, we already investigated notifications on smartphones, smartwatches, tablets, and PCs.

Nowadays, multiple devices are often used at the same time. Smartphones are, for example, becoming a second screen for the TV, offering interactivity through social networks [85]. Nathan et al. implemented CollaboraTV, a system for asynchronous interaction with the goal to bring people together even if they do not watch at the same time or place [99]. The results of a field study over the course of one month showed participants valued the system. Alaoui and Lewkowicz proposed a similar system for elderly to cope with loneliness [3].

Holz et al. found in a study that family members joined each other in the living room to be physically together [61]. Courtois found that there are three types of TV watching behavior [27]. One type only focuses on the TV, the second type watches TV with second screens, for example, tablets or laptops, and the third type uses seconds screens and even printed media.

Further work has been done in the field of program recommendation systems for TVs. Chang et al. give a literature overview and, based on the gained insights, propose a recommendation framework [20]. As recommendations are based on the user's interests, this creates challenges for multiple users. One possible solution for these challenges is merging interest profiles from the people in front of the TV, as proposed by Shin and Woo [141]. Lee et al. proposed a system for smart TVs that can authenticate the user using face recognition [81]. This can be used to automatically change the program recommendations depending on the user in front of the TV. Furthermore, the researchers propose using hand detection to control the smart TV with natural hand gestures.

Regan and Todd explored a system that allows multiple users to access their instant messages while watching TV simultaneously [132]. They state that people often use their PC to communicate in addition to watching TV. They looked at the aspects of privacy and distraction caused by such a system when watching TV with multiple people in the same room. To make users aware of incoming messages, they used pop-up alerts in the corner of the screen, similar to ones found on the PC. In a study, they found that for some people access to instant messaging is important even when watching TV. In the study, incoming messages were considered interrupting if they were not meant for the participant.

Hess et al. conducted empirical work on concepts for social TV experiences [59]. They state that through current technology the Web and TV is combined, which enables users to share content and communicate with others over distance. They identified a trend that watching TV is supplemented by other media. Multiple devices are used simultaneously, e.g., for communicating with friends. In a workshop, a group discussed notifications. Messages should be received on the smartphone, but users should be able to decide whether a notification should be displayed on the smartphone, the TV, or both. Neate et al. investigated how to draw attention to companion content on a second screen when watching TV [100]. They implemented several stimuli, including an icon shown in the corner of the TV. In a study conducted by Geerts et al. the need for a "do not disturb" mode was shown [46]. However, the researchers mention that users do not want to enable or disable this mode every time they do (not) want to be disturbed.

In summary, notifications are a core feature of current smart devices. They are used to alert the user through multiple modalities. While there is a corpus of work that investigated the use of smart TVs, no standard notification mechanism for smart TVs has been established. What is missing are design guidelines for the design of notifications on smart TVs.

6.1.2 Focus Groups

We conducted three focus groups to explore the design space of notifications on smart TVs. Each of the focus groups lasted approximately one hour and were held in a meeting room equipped with a whiteboard and projector. We provided post-its and black whiteboard markers, magnets and felt-tip pens (in 3 different colors) as well as printouts of a TV on DIN A4 paper. During the focus groups, we provided snacks and beverages. We compensated the participants for their time with EUR 10. In all groups, one researcher guided the discussion while another researcher took notes and wrote down participants' statements. In the following, we first describe the procedure of the focus group, which is based on Goodman et al. [48]. Afterward, we provide information about the participants and their behavior with respect to smart TVs. Then we present results, followed by a summary and a discussion.

6.1.2.1 Procedure

Each focus group had the same structure and consisted of four parts, an introduction, a round of idea creation, an open discussion, and finally a closing discussion with a summary.

Introduction First, participants were given a short introduction to the topic of the focus group. We prepared slides that explain the current state of notifications on various smart devices, the lack of notifications on smart TVs, and how we want to explore them. Furthermore, we encouraged the participants to speak freely during the session with the request to avoid talking at the same time. Afterward, we asked them to introduce themselves. In the introduction round all participants first stated their names and told the group the kind of devices they own that are able to notify them and the last important notification they can think of. Furthermore, the participants stated whether or not they own a TV and briefly talked about their TV-watching behavior.

Idea creation After the introduction round, we asked the participants to imagine a TV that can notify them about events, like messages, emails, or calendar reminders. We handed out sheets of paper with a TV printed on them and asked participants to sketch ideas on how such a system should look like and how it should behave. We asked them to consider multiple factors, including the content, size, position, and display duration of notifications. After approximately 10 minutes, we asked the participants to discuss their ideas with the person next to them. We instructed them to talk about positive and negative aspects of their ideas and to pick the ideas they like the most.

Open discussion After the idea creation, we collected all sketches that were selected by the participants and pinned them to a whiteboard. Figure 6.1 shows one of the focus groups in the discussion phase. We asked the participants to explain their ideas to the rest of the group. Subsequently, we asked the rest of the group about their thoughts on the idea, including the advantages and disadvantages. If not brought up by any of the participants, we asked them how their ideas would work when watching TV alone compared to watching TV with others.

Closing discussion and wrap-up After discussing the ideas of all participants, we explored with the group how far we can go with notifications on TVs. We asked them what they think about showing advertisements, weather forecasts, reminders, or product recommendations and openly discussed their concerns and suggestions. This discussion concluded the focus group.

Figure 6.1: Participants of one of the focus groups discussing their selected ideas on a whiteboard.

6.1.2.2 Participants

We recruited students from a university campus to participate in the focus groups. In total, 19 students showed interest in participating, and we divided those into three groups. The age of the participants was between 21 and 31 years ($M = 25.7$, $SD = 2.8$). The first group consisted of four female and four male participants and was held in English. The second group consisted of six male participants and was held in German. The third group consisted of one female and four male participants and was again held in English.

All participants owned a smartphone and either a desktop PC or laptop, or both. Nine (47.37%) participants stated that they own a tablet, and ten (52.63%) participants that they own a TV. Streaming was the participants' preferred way to watch movies, series, and news. Consuming those streams was not limited

(a) Toast (b) Ticker (c) Icons

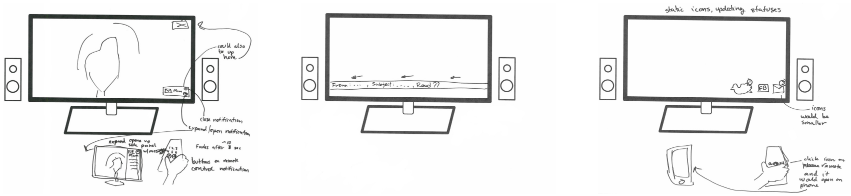


Figure 6.2: Sketches of notification styles created by the participants of the focus groups.

to the TV; instead, participants also watched them on their tablets and laptops. When asked about the last important notification they received, the participants mentioned email, instant messaging, and calendar notifications.

6.1.2.3 Results

In the following sections, we describe the analysis of the idea creation and discussion parts.

Notification styles To analyze the ideas created by the participants, three researchers went through all sketches and derived factors that distinguish them. Afterward, they agreed on one set of factors and described each sketch according to these factors. In total, we collected 46 sheets of paper, with 37 containing sketches of notification styles and 9 containing written comments. The most popular notification style with 19 sketches was the toast notification style known from desktop and mobile operating systems (see Figure 6.2a). Toast notifications overlay parts of the screen and typically consist of a box with an icon and two or more lines of text. On existing operating systems these notifications are typically only shown for a couple of seconds before disappearing again. On some sketches, it is mentioned that after a toast notification disappears, a less intrusive indicator should be shown on the screen, e.g., an app icon. In most sketches, the toast notifications were placed in the top right or bottom right corners of the screen.

The second most popular suggestion was a news ticker style at the top or bottom of the screen. Variants of the ticker style were found on six sketches. Figure 6.2b shows a sketch of a ticker notification at the bottom of the screen that scrolls the content from right to left. While not exactly the same, this style is similar to the notification ticker used in Android prior to version 5.0, which temporarily replaced the status bar at the top of the screen with a ticker that scrolled through the received message content. A concern that came up in the group discussion was that this style would cover subtitles when placed at the bottom of the screen.

Another option that was also suggested six times was to only show icons, similar to the status bar at the top of the screen of Android devices or the system tray area on desktop operating systems. The suggested place for these icons was, similar to the toast notifications, in the top right or bottom right corner. Participants mentioned that the icons could be enhanced by adding a badge to the icons that indicates the number of pending notifications for a certain application. Figure 6.2c shows three icons in the bottom right of the screen, with badges showing the number of notifications.

The fourth category of suggestions was about embedding a LED in frame or base of the TV. This variant was found five times on sketches. Participants suggested that the LED could change the color depending on the app that issued a notification or depending on the importance of the notification. This option would be similar to notification LEDs found on smartphones.

One participant stated that the TV should be used as a smart home hub, showing notifications and other information in full screen when the TV is not in use. Another participant suggested using a screen panel with a wider horizontal resolution that is reserved for notifications. This would allow for a persistent notification stream on the TV without covering content. Independently from the notification style, all participants agreed that sound should be completely optional and configurable. Furthermore, participants agreed that notifications should sync with other devices, thus dismissing them on one device should dismiss them on other devices, too.

Concerns Participants raised a number of concerns regarding notifications on TVs. A concern was occlusion of content. Notifications should be transparent to a degree, so nothing important is hidden. Examples were subtitles and scoreboards of sports broadcasts. Participants were concerned about bright pop-ups in an otherwise dark movie.

Another concern that was brought up in every focus group was the difference in watching TV alone in contrast to watching TV with others. The participants disliked the idea of notifications that show the sender and parts of the message while watching TVs with other people. One participant compared this with the scenario of giving a presentation and stated that he is always cautious about disabling all notifications when giving a presentation. A "family mode" was suggested that hides the content or disables the notifications completely when watching TV with others. Furthermore, participants stated that notifications should be context-aware. First, it should be detected if other people are in front of the TV, so that notifications can be adjusted or disabled automatically. Also, the idea of too many notifications was regarded as annoying, so only important notifications should be shown. Additionally, notifications should not be shown during truly immersive movies, but a summary of missed events after the movie or during slow moments would be acceptable.

In the closing discussion, some participants stated that if the notifications were used to display advertisements, they would disable the notifications. Others mentioned that if advertisements would allow them to watch movies or series for free, they consider them acceptable. Recommendation notifications, for example, that the successor to the movie that is being watched is currently shown in the cinemas, was considered tolerable, as long as it is not overused. The participants agreed that calendar reminders might be useful.

6.1.2.4 Summary and Discussion

In this section, we described the procedure of three focus groups we conducted in order to explore the design space of notifications on smart TVs. The focus groups consisted of four parts, an introduction round, idea creation, open discussion, and a closing discussion. In the idea creation part, participants drew sketches of possible notification mechanisms on smart TVs. Categorizing these sketches resulted in four categories for notification styles. The most popular styles were toast notifications, followed by ticker and icon-based notifications. Further variants include embedding LEDs in the TVs frame or base and using the TV as a hub for smart homes. In addition to these visual cues, sound could be used. However, sound should be optional and configurable.

Participants were concerned about the privacy aspects of showing notifications on the TV when watching with other peoples. It was suggested to adjust the information shown depending on the number of people in front of the TV. Another concern was the occultation of the screen content and distractions caused by notifications. Therefore, notifications should be only used for important events, for example, messages from important contacts or calendar reminders.

6.1.3 Online Survey

Based on the findings from the focus groups, we further investigated how much content should be shown in notifications on smart TVs. To gain results from a wide variety of people, we designed an online survey.

Therefore, we created five notification variants with varying amounts of information. The variants are shown in Figure 6.3. We focused on the amount of information shown rather than the design itself. Because of this, we decided to show all notifications as toast notifications, as this style was the most popular in the focus groups and is common in desktop setups to present notifications. Another preference from the participants of the focus groups was the positioning in the top right or bottom right corner. Accordingly, we displayed all notification variants in the top right corner. Apart from the variants, we decided on one scenario. Therefore, we created videos for the five variants. Each video played back the same video content, and each video was 25 seconds long. While the video was playing three notification popped up, the timing was the same for all variants, namely at 4, 15 and 18 seconds after the start. The displayed notification are an email, an instant message, and second email notification.

In *Variant 1*, a generic notification icon is shown, and a badge on the icon keeps track of pending notifications (Figure 6.3a). *Variant 2* uses app-specific icons instead of the generic icon, and the name of the app that created the notification is briefly shown (Figure 6.3b). *Variant 3* behaves similar to the second variant;

Figure 6.3: The five notification variants with varying amounts of content, as shown at 4, 6, and 23 seconds in the video (from top to bottom).

however, the sender of a message is also shown (Figure 6.3c). Furthermore, in *Variant 4*, an excerpt of the message is shown below the sender, thus showing the most information (Figure 6.3d). These four variants are persistent until dismissed. *Variant 5* also displays the sender and the message excerpt, however, no icon is left behind (Figure 6.3e).

We designed an online survey to receive feedback for the notification variants. The online survey was distributed via mailing lists, social networks, and online communities.

6.1.3.1 Procedure

The online survey was answered by the participants in their web browser and consisted of three parts. First, we asked participants about demographic data, TV watching behavior, and devices they are notified on. In the second part, all notification variants were rated by the participants. The notification variants were counter-balanced (displayed in random order). For every notification variant, a short textual description text was provided along with an embedded YouTube video.

For each condition, the participants were asked to rate the following five statements from "Strongly disagree" to "Strongly agree" on a 5-point Likert scale.

(Q1) With this notification mechanism, I have the feeling that I am not missing a notification anymore.

(Q2) This notification mechanism provides me the information that I want.

(Q3) This notification mechanism disturbs my TV-watching-experience.

(Q4) I'd feel comfortable using this notification mechanism when I am watching TV alone.

(Q5) I'd feel comfortable using this notification mechanism when I am watching TV with others.

Finally, the participants should rate the two statements "It is important for me to know how many notifications from each application do I have." At last the participants could comment our notification variants.

6.1.3.2 Participants

In total, 167 people (50 female, 117 male) completed the survey. They were between 15 and 76 years old ($M = 28.8$, $SD = 10.2$), with 58% being students, 35% employees and 7% others. The online survey was available in English, German, and Spanish. The English version was completed 46 (27.54%) times, the German version 105 (62.87%) times and the Spanish version 16 (9.58%) times. The size of the participants' households had a notable variety. 19.7% participants stated that they live alone, 25.1% with another person, 24.5% in a three-person household, 22.1% in a four-person household and 6.0% live with five or more persons. 2.3% did not state the size of their household.

We asked "How many hours per day on average do you watch TV alone?" and "How many hours per day on average do you watch TV in company with other people?". In Table 6.1, we present the participants' TV usage.

	$0h$	$< 0.5h$	$0.5-1h$	$1-2h$	$2-3h$	$3-4h$	$> 4h$
Alone	19.1%	26.3%	13.7%	23.3%	10.7%	1.1%	5.3%
Others	26.3%	20.3%	20.9%	19.1%	7.7%	2.3%	2.9%

Table 6.1: Hours spent per day watching TV alone and with others.

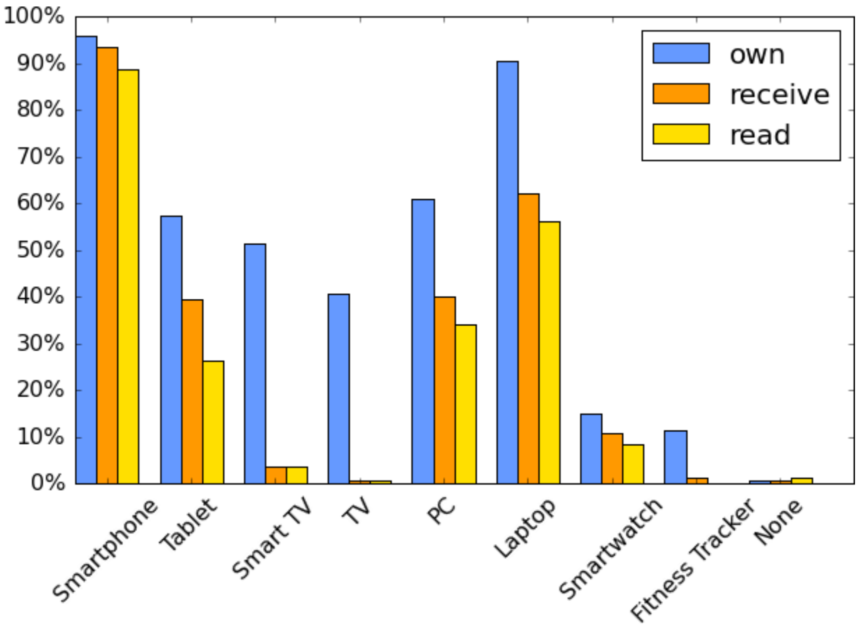


Figure 6.4: Devices which participants of the online survey own, receive notifications on, and read notifications on.

We also asked the participants what kind of devices they own, on which devices they receive notifications and on which devices they actually read notifications. Possible options were smartphone, tablet, Internet-enabled TV, TV without Internet, desktop PC, laptop, smartwatch, fitness tracker, and none. On smartphones, tablets, and PCs notifications are a well-known paradigm to receive the attention of the user. Current smartwatches and fitness trackers often connect to a smartphone. Figure 6.4 shows the responses. 95.81% own a smartphone, 57.49% a tablet, 51.50% a TV with an Internet connection, 40.72% a TV without an Internet connection, 61.08% a desktop PC, 90.42% a laptop, 14.97% a smartwatch and 11.38% a fitness tracker. One participant stated that he does not own any of these devices. Generally, participants receive and read notifications on all smart devices with smart TVs being a notable exception.

6.1.3.3 Results

We analyzed all subjective ratings of the five conditions (Figure 6.5) using a Friedman test. We also analyzed the ratings for each rating using the Friedman test and Wilcoxon signed-rank post-hoc tests with an applied Bonferroni correction, resulting in a significance level of $p < 0.005$.

(Q1) Not missing notifications We found a significant difference for Q1, $\chi^2(4) = 115.020$, $p < .001$. For this statement Variant 3 ($M = 4.40$, $SD = 1.08$) and Variant 4 ($M = 4.40$, $SD = 1.13$) received the highest ratings, followed by Variant 2 ($M = 4.07$, $SD = 1.22$), Variant 5 ($M = 3.78$, $SD = 1.36$) and Variant 1 ($M = 3.38$, $SD = 1.34$). The rating of the variant with the generic icon is significantly lower than all other variants (1vs2 $Z = -6.322$, $p < .001$, 1vs3 $Z = -7.436$, $p < .001$, 1vs4 $Z = -7.436$, $p < .001$, 1vs5 $Z = -2.860$, $p = .004$). Variant 5 is significantly lower rated than Variant 3 ($Z = -5.326$, $p < .001$) and Variant 4 ($Z = -5.464$, $p < .001$).

(Q2) Provides wanted information We found a significant difference for Q2, $\chi^2(4) = 123.015$, $p < .001$. For this statement Variant 3 ($M = 3.99$, $SD = 1.29$) received the highest rating, followed by Variant 4 ($M = 3.96$, $SD = 1.28$), Variant 5 ($M = 3.86$, $SD = 1.32$), Variant 2 ($M = 3.45$, $SD = 1.36$) and Variant 1 ($M = 2.86$, $SD = 1.22$). Again, Variant 1 received a significantly lower rating all other variants (1vs2 $Z = -5.798$, $p < .001$, 1vs3 $Z = -7.922$, $p < .001$, 1vs4 $Z = -7.022$, $p < .001$, 1vs5 $Z = -6.953$, $p < .001$). Also, Variant 2 (app icons, no text) received a significantly lower rating than variants with text, namely Variant 3 ($Z = -4.325$, $p < .001$) and Variant 4 ($Z = -3.409$, $p = .001$).

(Q3) Disturbs TV experience We found a significant difference for Q3, $\chi^2(4) = 17.560$, $p < .001$. For this statement Variant 4 received the highest disturbance rating ($M = 3.74$, $SD = 1.36$), followed by Variant 3 ($M = 3.56$, $SD = 1.35$), Variant 2 ($M = 3.49$, $SD = 1.37$), Variant 5 ($M = 3.42$, $SD = 1.38$) and Variant 1 ($M = 3.38$, $SD = 1.40$). Variant 1, which displays only a generic icon, received the lowest disturbance rating.Variant 4, with sender and message excerpt, was rated significantly more disturbing than all other variants (5vs4 $Z = -3.533$, $p < .001$, 2vs4 $Z = -3.073$, $p = .002$, 3vs4 $Z = -3.018$, $p = .003$, 1vs4 $Z = -3.751$, $p < .001$).

(Q4) Comfort alone We found a significant difference for Q4, $\chi^2(4) = 22.216$, $p < .001$. For this statement Variant 5 received the highest rating ($M = 3.93$, $SD = 1.36$), followed by Variant 3 ($M = 3.81$, $SD = 1.38$), Variant 2 ($M = 3.78$, $SD = 1.35$), Variant 4 ($M = 3.75$, $SD = 1.37$) and Variant 1 ($M = 3.41$, $SD = 1.38$). Variant 2-5 are not significantly different. Variant 1 has a significantly lower rating than Variant 2 ($Z = -3.398$, $p < .001$), Variant 3 ($Z = -3.654$, $p < .001$) and Variant 5 ($Z = -4.014$, $p < .001$).

(Q5) Comfort with others We found a significant difference for Q4, $\chi^2(4) = 60.511$, $p < .001$. For this statement Variant 2 received the highest rating ($M = 3.19$, $SD = 1.36$), followed by Variant 1 ($M = 3.18$, $SD = 1.39$), Variant 3 ($M = 2.89$, $SD = 1.26$), Variant 5 ($M = 2.77$, $SD = 1.23$) and Variant 4 ($M = 2.59$, $SD = 1.10$). Variant 2 is significantly different to all variants except Variant 1 (2vs3 $Z = -3.415$, $p = .001$, 2vs5 $Z = -4.108$, $p < .001$, 2vs4 $Z = -5.236$, $p < .001$). Variant 1 is significantly different to Variant 3 ($Z = -3.059$, $p = .002$), Variant 5 ($Z = -4.127$, $p < .001$) and Variant 4 ($Z = -5.008$, $p < .001$). Also, Variant 3 is significantly different to Variant 4 ($Z = -3.636$, $p < .001$).

Optional comments The last part of the online survey included a free text field that allowed the participants to enter a comment independent of the previous tasks. Two researchers translated comments written in Spanish and German to English and filtered comments without usable feedback. This resulted in 55 comments that were subsequently categorized by their content.

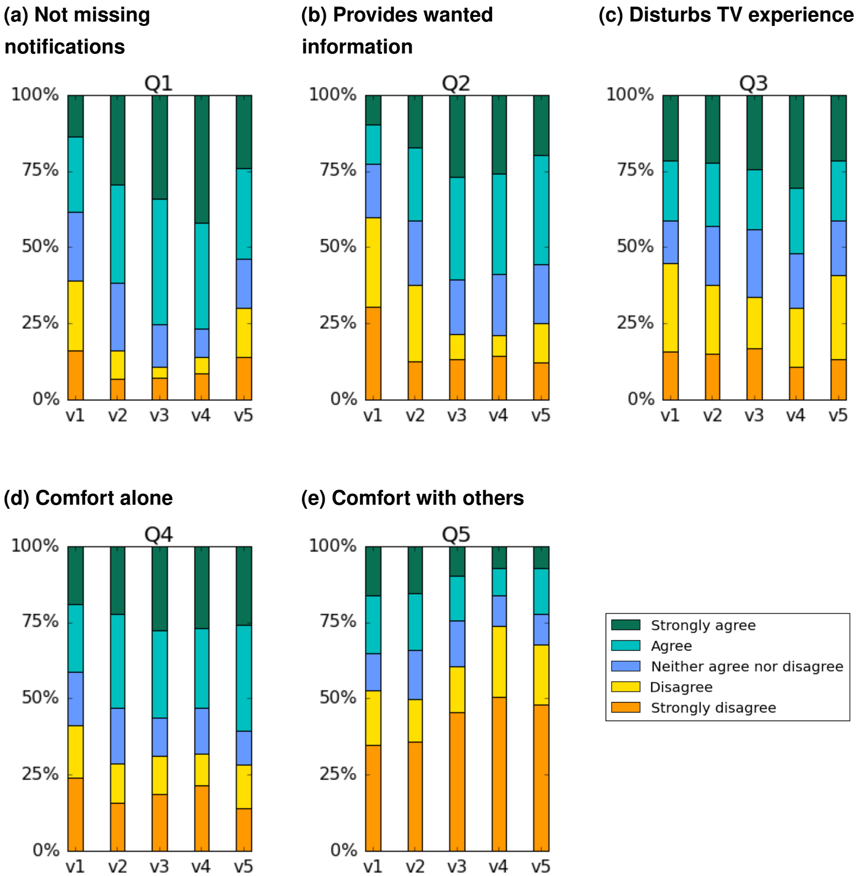


Figure 6.5: Ratings of the five statements (Q1-5) of the online survey for each notification variant (v1-5).

Thirteen participants explicitly stated that they would not use a notification system on their TV under any circumstances. Two participants stated that they do not want to be disturbed when watching TV at all and thus silence their smartphones. Three other participants were not as opposed to receiving notifications on the TV. Instead, they stated that it depends on the importance of the notification, which in return depends on the urgency or person sending the message. An interesting category of comments from 7 participants distinguished between watching a movie and "entertainment programs," for example quiz shows "where you do not have to actively focus on the program to follow it" *(translated from German)*. Two participants suggested displaying notifications after a movie.

In the survey, we asked the participants how comfortable they would feel using this notification style alone compared to using it when other people are around. In the free text field, 5 participants addressed this issue. They suggested multiple modes that can be switched depending on how many people are around. One mode would display notifications without restrictions, whereas the "private" mode would only display notification hints. Customization is another topic that was addressed by 13 participants. They suggested changes to the notification shown in the videos and overall options they would like to see, from the color of the notification to the screen corner that should be used.

6.1.3.4 Summary and Discussion

In this section, we described the online survey, where we evaluated five notification variants with a varying amount of content. For each notification variant, we asked participants to rate their agreement to five statements and asked them what they like and dislike. Furthermore, we asked them in a free text field to give us general feedback to notifications on smart TVs. The participants owned a number of smart devices, on which they receive and read notifications. However, an exception to these were TVs and smart TVs on which most participants did not receive or read notifications.

The results of the online survey indicate participants prefer to see the sender or the sender in addition to a message excerpt in the notification. Participants are concerned about missing notifications if no indicator is left behind and showing only a generic icon is not enough information for the participants. However,

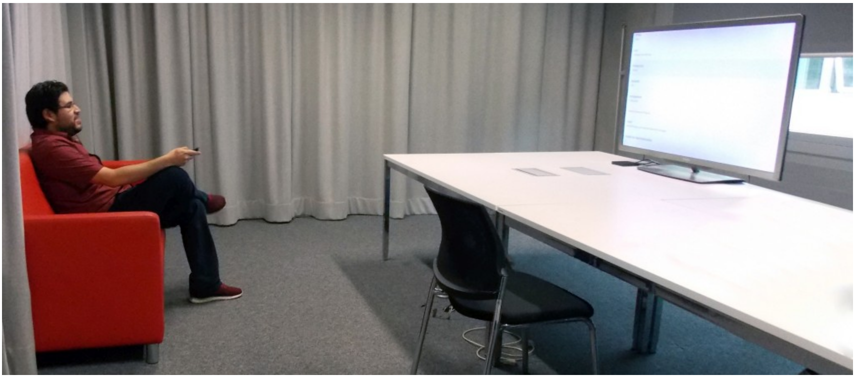


Figure 6.6: The setup of the lab study. A participant is customizing the notification toast on the TV using a remote.

persistent indicators and showing more text in the notification increases the occluded display space. Therefore the participants stated that the variants with text disturb the TV watching experience the most. Four of our tested variants left an icon behind and not doing that could decrease the disturbance created by the text. When watching alone, participants liked all variants except the generic app icon. When watching with others, participants liked the variants that show the sender or message less.

6.1.4 Lab Study

In the online survey, we investigated the amount of content which should be shown in notifications on smart TVs. One major result is that notifications should be customizable by the user. To further investigate in this direction, we conducted a lab study where participants had the task to customize a toast notification. Therefore, we set up a room in our lab with a sofa and a TV (see Figure 6.6). We implemented an application which enables us to push notifications to the TV while a video is playing. Derived by the results from the online survey, there is a need to investigate the customization while watching alone and with others. Therefore, we conducted the lab study with two groups, one group watching alone and the other watching together with a second unknown person. This was done to see if participants choose different settings. In the following, we describe the study as well as the results.

6.1.4.1 Design

To get insights into the differences between watching television alone and with other people, we ran the study with a between-subjects design. The participants of one group (A) sat alone in front of the TV, while the second group (B) watched a video in the presence of a researcher. We used a 55" Philips Full HD TV connected to an Amazon Fire TV box to achieve a realistic TV experience. The Amazon Fire TV enabled us to push notifications on top of a video and also enabled the participant to customize them. Another limitation of the online survey was that we created an exemplary scenario, resulting in notifications that were not meaningful for the participants. Therefore, we used our previously developed notification logging app to log all notifications shown on the participants' smartphones. All notifications shown in the lab study were, therefore, notifications the participants recently received. The notifications were selected randomly from the log files and varied from instant messaging notifications to system messages.

For the lab study itself, we developed an Android application that was installed on the Amazon Fire TV. This app is capable of playing back a video while showing an overlay with a notification. Furthermore, it allows the user to control the representation of the notification with nine different settings. The GUI of the settings menu is shown on the left side in Figure 6.7. These settings are position, size, icon, theme, opacity, duration, content, lines, and sound. The *position* setting controls where notifications appear on the screen, with nine possible options from the top left to the bottom right. The *size* setting allows to scale the notification from small (225dp), medium (300dp) to large (375dp), using Android's density-independent pixels (dp) metric. The *icon* setting allows showing the icon of the app in full color and grayscale, a generic-app icon in color and grayscale, or no icon at all. The *theme* setting allows to set the background of the notification to white (light theme) or black (dark theme). The *opacity* allows setting the opacity to 25%, 50%, 75% or 100%. The *duration* setting controls how long the notification is shown, from 1 second to 25 seconds. The *content* setting controls

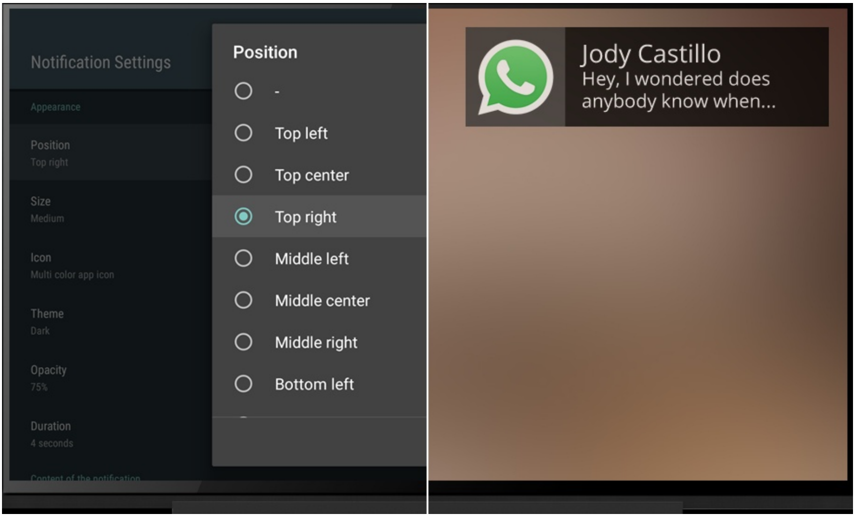


Figure 6.7: The study app on the Amazon Fire TV. The left side shows the settings with the *position* options dialog. The right side shows an exemplary *WhatsApp* notification with the *most popular* settings.

how much of the logged text is shown. Possible options are to only show the name of the app, to include the title/sender, and to show title/sender and message. The *lines* setting depends on the content setting, because it controls how many lines are shown, with possible values being 1-5 or unlimited. The *sound* setting can be either enabled or disabled, and plays a default sound when enabled.

6.1.4.2 Procedure

We invited the participants two times. The first time to sign a consent form and to set up the notification logger. Two days later we invited them the second time to our lab. First, we asked them to fill in a demographic data form and seated them on a sofa in front of the TV ($3\,m$ between screen and participant). Then we explained that we built an application for the TV that would display random notifications from the past two days while an episode of the series "Big Bang Theory" was playing. For group A, it was explicitly stated that they would watch the episode alone, without anyone in the room. For group B, it was stated that the researcher would stay in the room. We opened the settings screen and briefly introduced the participants to the nine available settings. At this point, no setting was configured yet. Therefore, the participants were asked to explore the settings by themselves. After configuring all settings, a preview notification appeared that allowed the participants to make further adjustments. When participants decided that the notification's representation was appropriate, we started the first half of the episode. For group A, the researcher left the room. Ten notifications were shown at predefined times. The predefined times for displaying notifications were randomly chosen by us and were the same for each participant. After the first half finished, the episode was paused, and the settings page was opened automatically. The participants had the opportunity to change their settings for the second half of the episode. In the second half, ten additional notifications were shown. After watching the full episode, the settings page opened again, and participants were asked to adjust the settings one more time. Finally, we asked participants to rate the importance of each setting on a 5-point Likert scale.

6.1.4.3 Participants

In total, 14 participants (5 female) took part in the study. All participants were recruited on the campus of the University of Stuttgart. They were between 22 and 32 years old ($M = 25.86$, $SD = 2.95$). Twelve of the participants were students, one participant was a PhD student, and one participant was a promoter.

6.1.4.4 Results

In Figure 6.8 the agreement to the importance of the settings is shown, highlighting the need for customization of notifications. The three most important settings to customize the notifications were the *position* ($M = 4.79$, $SD = 0.43$), *size* ($M = 4.71$, $SD = 0.47$) and *content* ($M = 4.50$, $SD = 0.65$). Followed by *duration* ($M = 4.29$, $SD = 0.83$), *lines* ($M = 4.01$, $SD = 0.62$), *opacity* ($M = 3.93$, $SD = 1.14$), *icon style* ($M = 3.64$, $SD = 0.84$) and *sound* ($M = 3.50$, $SD = 1.83$). The *theme* setting received neutral ratings ($M = 3.00$, $SD = 1.24$). However, statistics did not reveal any significant difference between people, who watched alone or together with other people.

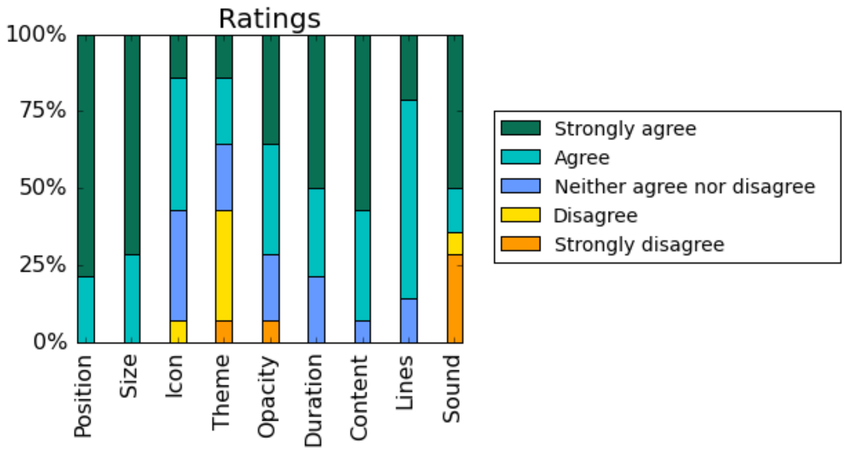


Figure 6.8: The importance ratings for the nine different settings we investigated in the lab study.

Derived from the participants' final settings, the following values are the most popular. For the nominal setting values, we will report the *modus*, and for the duration as a scale we report M and SD. This results in a *most popular* notification style, which is represented as follows: The notification is in a dark-themed box in the upper right corner displayed for $M = 4.93$, $SD = 2.6$ seconds with 75% opacity. Including a colored app icon, the sender and two/three/unlimited lines of the message, with a small font and no sound. The visual representation is shown on the right side of Figure 6.7.

Position Nine participants preferred the position in the upper right corner, two participants chose the bottom left corner, and another two participants chose the bottom right corner. There are no significant differences between both groups. It is important that notifications are positioned in a way that provides visibility, but also does not hide the content or program inserts (P4, P8). Two participants argued that they chose the position because they are used to it from their smartphones and laptops (P11, P12).

Size Ten participants chose a small representation of the notification, four the medium size, and none the large size. There was no significant difference between both groups. The notifications should be big enough to read and small enough not to hide the content (P8). Too big overlays are annoying (P4, P14) and the size should depend on the TV's size and the distance to TV, too (P2).

Icon The selection of the used icon depends on the two groups. Participants, who watched together with a researcher have chosen an icon, which belongs to the incoming notification. The app icon in color was chosen by 5 participants, and 2 participants used the app application icon in grayscale. Participants who watched alone chose dissimilar icons. Only 3 participants chose the app icon in color. Two participants used a generic icon for an incoming notification, and two others decided to hide the icon completely. The usage of an application icon helps to judge the importance of the notification, which generated the notification (P1, P3, P9, P10, P11).

Theme Ten participants set the dark variant and four the light one. Two participants mentioned that the contrast is important (P1, P4) and two other participants think there is not much of a difference between the light and the dark theme (P9, P12).

Opacity Participants who watched alone all chose a high opacity, 6 of them used the 75% opacity, and 1 participant used the 100% opacity. From the participants who watched together with a researcher, one chose 25% opacity, and two participants chose 50%, 75%, and 100% respectively. The notification should not block the TV content (P8, P12) and not be too transparent (P3, P12). This setting is important for minimal distraction (P11). One participant thinks an opacity with 25% or 50% is too transparent (P3), while another participant said the opacity should be between 25% and 50% to not block the TV content (P8).

Duration Participants who watched alone chose longer durations for displaying the notifications. One participant used a duration of $3\,sec$, one participant used a duration of $4\,sec$, 4 participants used a duration of $5\,sec$, and one participant chose a duration of 13 *sec*. However, 2 of the participants who watched together with a researcher chose a duration of 3 *sec*, 2 participants used 4 *sec* for the duration, and 3 participants chose a duration of 5 *sec*. The setting for the duration of displaying the notification is a balance between being long enough to read the message and short enough, so the notification is not a nuisance (P11). The opinions to the duration diverged, too. One participant who watched alone thinks more than 10 *sec* are too much for displaying the notification (P3). However, a participant who watched with a researcher commented $2-3$ *sec* are enough for displaying the notification (P8). Another participant prefers that there should be a standard duration, and user can terminate to read or skip by pressing a button (P2).

Content From the participants who watched alone, 1 participant chose to display the sender only, 6 of them chose to display the sender and the message of the notification. For the participants who watched together with a researcher, 4 participants chose to display only the sender, and 3 of them chose to display the sender and message of the notification. No one of the participants chose the option to display only the name of the application. The participants said that it is important to decide what should be displayed on the screen because of privacy issues (P1, P8, P12). There will be some people who want to read the notification only on their phone (P2), but other people might want to read the notification on the TV (P2). When more text is displayed, longer attention is required, and so you could miss what you are watching (P9) but also affects to what extent you are informed (P10).

Lines From the nine participants who chose to display the message of their notifications, three chose two, three, and unlimited lines of text, respectively. These include participants who watched together with a researcher, one of them chose 2 rows and two others 3 rows for the message. The length of the displayed content is a privacy setting as well and depends on who could see the notification (P8, P10). Another participant suggested a meaningful reduction of the displayed content when full text is too much for a short insert (P4).

Sound All participants but one disabled the sound for an incoming notification. They argue that the sound makes no sense (P1), is not necessary (P2) and distracting (P11). Three participants perceived the sound as annoying (P4, P10, P12). One participant thinks that the sound might bother some people but might help to remember acting on the notification after watching TV (P9).

6.1.4.5 Summary and Discussion

In this section, we described our lab study, where we invited 14 participants to customize notifications while watching TV. The lab study revealed a clear need for customization. Participants rated the importance for all settings on average at least to *neither agree nor disagree*. We also reported qualitative feedback regarding the provided settings. Furthermore, we presented the *most popular* configuration of settings which can be used as an initial setting for further studies. One limitation of the "watching with a researcher" approach is the relationship between the participant and the researcher. In future studies, differences between watching with friends, family, or the partner should be investigated.

6.1.5 Design Guidelines

Based on our findings from the focus groups, the online survey, and the controlled lab study, we derived the following guidelines for notifications on TVs. The guidelines can be used by developers to gain the user's attention on smart TVs in a meaningful way.

6.1.5.1 Evaluate the Importance

Developers should evaluate the importance of notifications instead of creating a stream of notifications as it is currently the case on other smart devices. Related work on smartphone notifications has shown that important notifications are about people and events [137]. Insights gained in the focus groups and the online survey confirmed this. For some people, nothing is important enough to distract them from their immersion when watching TV. Because of this, notifications on smart TVs should always be optional.

6.1.5.2 Privacy Considerations

Privacy aspects on smart TVs differ from other smart devices. TVs are typically shared devices and are used by multiple people, often at the same time. Unlike other smart devices, it is therefore not recommended to simply display message excerpts in notifications. An idea brought up in the focus group was using multiple profiles depending on how many people are in front of the TV. One profile could be used for watching TV alone with no restrictions to the displayed information. Another profile could be used when watching TV with others. In this "private" profile, notifications could show various levels of information. For example, not showing the message excerpts, excluding the sender or using a default application icon. We suggest a system that detects people in front of the TV and uses this knowledge to adjust the amount of information shown in the notifications automatically. If an automated solution is not possible, it should be at least possible to switch between a public and private mode with ease.

6.1.5.3 Time Interruptions

Multiple participants of the online survey mentioned that they like the idea of notifications on the TV. However, the notifications should not be shown during movies, as this was regarded as distractive. Instead, participants suggested showing notifications after a movie. Previous work on timing notifications has shown that notifications are less distractive if they are shown in between tasks [2]. Apart from the end of a movie, we suggest notifying the user during advertisement breaks and, in the case of video-on-demand movies, when the movie is paused.

6.1.5.4 Be Subtle

Notifications on smart TVs should be subtle. Effects and animations should be used with care to avoid distracting the user. Participants of our lab study disliked the idea of playing a sound. The size, opacity, display duration, and text length have to be balanced in order to maximize readability and minimize occlusion of the content.

6.1.5.5 Allow Customization

In all studies, participants agreed that it must be possible to customize how notifications are displayed. As stated above, the amount of information to be displayed should be customizable. Furthermore, the position of the notification and display duration on the screen is something that participants were not in agreement, thus should be configurable.

6.2 Notifications on Public Displays

While smart TVs are large displays typically used at home, we now want to focus on another type of large displays. Public displays are becoming increasingly common in many public or semi-public environments. Currently, most public displays are used to display general information or advertisements, as displaying personal content poses many privacy implications. In prior work, Vogel and Balakrishnan developed design principles and an interaction framework for displaying personal content on public displays [157]. Their framework increases the level of personal content as the user is getting closer to the display. The authors argue that only "harmless" personal content should be shown. Shoemaker and Inkpen investigated interaction techniques to allow private information on shared displays [142]. Langheinrich explored design principles for privacy in ubiquitous computing systems, including the issues of "choice and consent," "proximity," and "pseudonymity" [79]. Alt et al. developed *Digifieds*, a digital public notice area [4].

The lack of public displays showing personal content indicates that further research is needed in this area. We argue that, to learn about personal content on public displays, it is necessary to conduct studies using diverse sets of personal content. Smartphone users are confronted with proactively provided personal content on a daily basis through notifications. Apps provide users with a wide range of content, e.g., instant messages, emails, game invites, personalized news, upcoming appointments, and app updates. The nature of notifications is that for

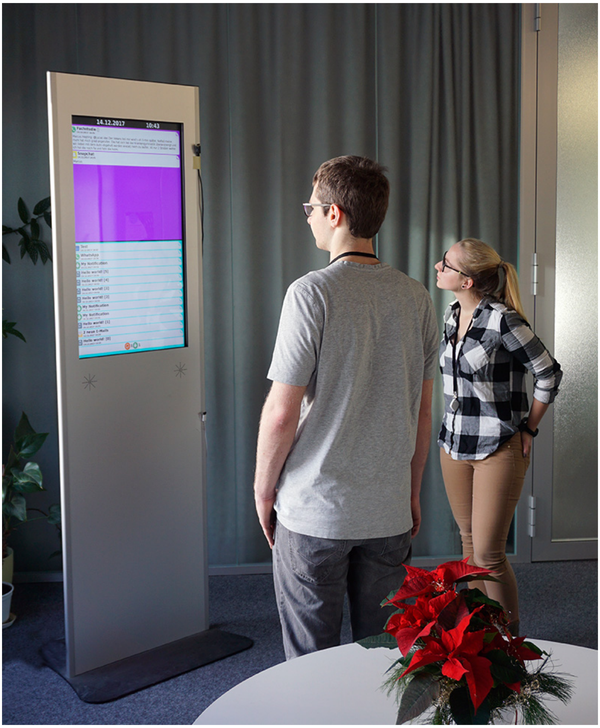


Figure 6.9: Two users are standing in front of a public display that was used in the study. The display is located in a semi-public kitchen environment. The users wear lanyards with Bluetooth Low Energy (BLE) beacons around their necks. The public display detects the beacons and mirrors the users' smartphone notifications.

the most part users do not know when or what they are being notified about. Combined with the diversity of apps and therefore the types of content, smartphone notifications are an excellent source for personal content.

In this second part of the chapter, we introduce *PD Notify*, a system to explore displaying personal content on public displays. *PD Notify* mirrors the user's pending smartphone notifications on nearby public displays (see Figure 6.9). The system consists of public displays that can detect nearby users and a smartphone application that forwards the users' notifications to the displays. Users are in control how much content should be shown on the displays, on a global and an

Figure 6.10: A Bluetooth Low Energy (BLE) beacon as used in the study. All participants carried the same type of beacon, allowing for a reliable detection of nearby users. Ballpoint pen for scale.

app-specific level. We report the system architecture and the first deployment in a semi-public work environment. To evaluate the system, we conducted a three-week-long in-situ study with seven participants. In the study, we logged the participants' behavior with the system and conducted semi-structured interviews. The results show that displaying personal content on public displays is not only feasible but also valued by users. Participants quickly settled for privacy settings that work for all kinds of content. While they liked the system, they did not want to spend time configuring it.

6.2.1 System

PD Notify mirrors users' pending smartphone notifications on nearby public displays (see Figure 6.11). To access the notifications, we implemented an Android

Figure 6.11: An exemplary smartphone notification mirrored to a public display. Depending on the privacy level selected by the user, the level of detail on the public display is reduced. The fourth privacy level is not to mirror notifications from the app at all. The background color (blue) is used as a color-code to provide pseudonymity.

app that listens for added and removed notifications on the user's device. The app forwards the notifications to a central server using a secure connection. The central server can then forward the notifications to connected public displays. The system supports any number of public displays. Privacy settings in the app allow users to control how much content should be sent to the central server and, therefore, should be shown on the public displays. Based on our work on notifications on shared smart TVs in the first part of the chapter, we implemented four

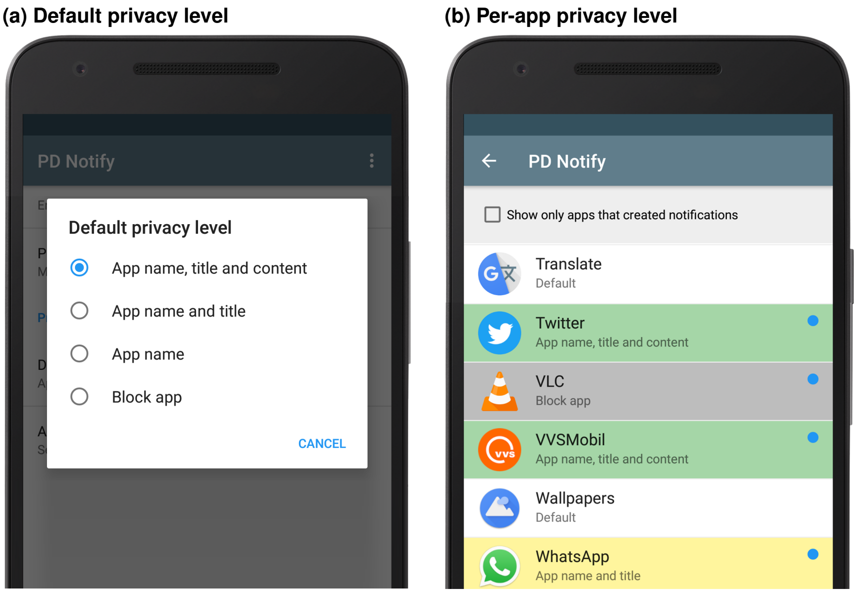


Figure 6.12: Left: Settings dialog for the default privacy level, applying to all apps unless overwritten. Right: Per-app privacy level overview. Blue dots indicate that the app created at least one notification. Overwritten privacy levels are color-coded to provide information at a glance.

privacy levels that correspond to "send everything," "limited content," "app name only," and "nothing." A global privacy level applies to all apps (see Figure 6.12a) and can be overwritten on a per-app basis (see Figure 6.12b). Changing the privacy level updates the content shown on the public displays instantly. Further, a "clear all" button allows users to remove all notifications from all displays at once immediately.

According to prior work, personal content should only be shown on public displays if users are near the display [157]. We explored the idea of using the smartphones' Bluetooth functionality to detect nearby users. In tests, we found that the Bluetooth signal quality varied considerably between different kinds of smartphones. Instead, we opted for using "Gigaset G-tag" Bluetooth Low Energy

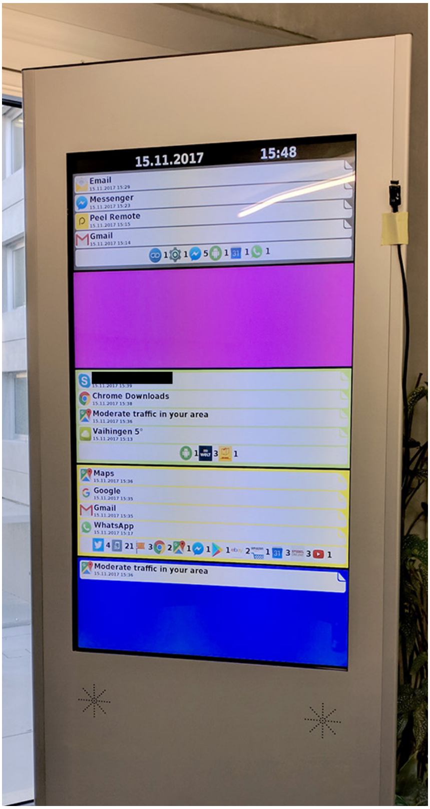


Figure 6.13: State of the public display with five nearby users. Two users allowed reduced content (green, blue), another two users allowed only app names (gray, yellow), and one user did not have pending notifications (pink).

(BLE) beacons attached to lanyards (see Figure 6.10). These beacons broadcast a unique Bluetooth address every two seconds. The public displays continuously scan for the beacons. A user is regarded as near a public display if the number of received BLE broadcasts in a time window and the Received Signal Strength Indicator (RSSI) are above certain thresholds. The thresholds are defined per public display depending on the environment. The BLE scanners inside the public displays continuously send the list of detected users to the central server, which in

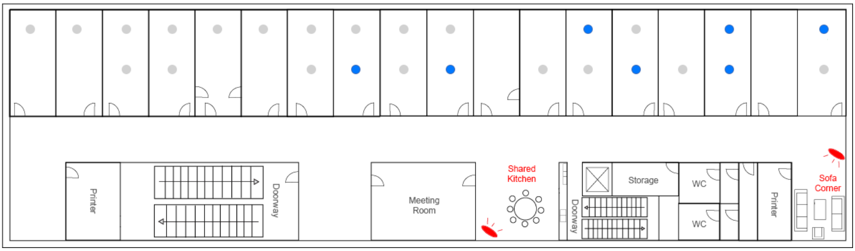


Figure 6.14: Floor plan of the corridor the study was conducted in. The corridor itself is semi-public with students and visitors regularly walking around. A shared kitchen and a sofa corner are popular meeting places in which we set up a public display each (shown in red). Blue dots indicate the offices of the study participants. Gray dots indicate people who did not participate in the study.

return forwards the nearby users' notifications to the display. If multiple users are near a display, the screen space is divided equally. All users are assigned a specific color on the public display that allows them to quickly see which notifications belong to them while providing pseudonymity (see Figures 6.9 and 6.13).

6.2.2 Study

We now report the first deployment of the *PD Notify* system. In a three-week-long in-situ study, seven participants mirrored their personal smartphone notifications on two public displays in a semi-public work environment.

6.2.2.1 Design

We conducted the study in a corridor of a building at the University of Stuttgart. We set up two public displays in popular meeting areas. Both displays featured 40" screens with a resolution of 1080×1920. Two "Raspberry Pi 3 Model B" single-board computers powered the displays and continuously scanned for nearby beacons. The corridor plan with the public displays and the participants' offices is shown in Figure 6.14.

6.2.2.2 Procedure

All participants signed a consent form and filled out a survey about demographic data. We then installed the Android app on the users' personal Android smartphones. We walked them through the app and explained all settings. The app was set to mirror all notification content to the public displays for all participants. Then, we distributed the beacons and assigned a color-code to each participant. We instructed the participants to use the app as they see fit and explicitly stated that disabling the app was allowed. After three weeks, we invited the participants to export the log data, fill out a questionnaire, and conducted semi-structured interviews. All participants participated voluntarily and did not receive a monetary reward.

6.2.2.3 Participants

We recruited participants from the corridor who owned Android smartphones. We excluded one participant due to technical reasons, resulting in seven participants (1 female). They were between 26 and 35 years old ($M = 29.14, SD = 3.24$). All participants were PhD students with a technical background.

6.2.3 Results

We now report the user's interaction and experience with the system, and summarize the semi-structured interviews.

6.2.3.1 Notifications and Privacy Settings

During the study participants received between 135 and 608 notifications per day ($M = 375, SD = 204$). These numbers include updates to existing notifications, like periodically refreshing weather forecasts. Participants had between 15 and 51 apps notifying them ($M = 33, SD = 12$). Categorizing these apps showed that most notifications were from the category *Instant Messaging* (46.96%), followed by *Email & Phone* (21.49%), *Tools* (7.46%), *General Information* (8.84%), *Android System* (8.08%), *Entertainment* (2.78%), *Finance & Shopping* (2.37%), *Social*

Media & Dating (1.71%), and *Health & Fitness* (0.30%). On average, per day participants spent 35 minutes in front of the public display located in the kitchen area and 30 minutes in the sofa corner.

All but one participant initially tested various default privacy levels and settled for one setting that they were comfortable with for all kinds of content within the first day of the study. In the end, no participant chose to display all content, two participants allowed reduced content, and four participants permitted only the app names. One participant initially allowed all content to be shown but changed the setting to block all content after five days. The participant told us that he assumed that no one could speak his language but then noticed that people were, in fact, reading his notifications. This caused him to block all notifications on the public displays. App-specific privacy settings were only used by 3 participants, in all cases to block specific apps completely. One participant blocked 14 of 31 apps (various categories), one blocked 4 of 51 apps (all *Social Media & Dating*), and one 2 of 25 (*General Information* and *Entertainment*).

6.2.3.2 Questionnaire

We asked participants to rate statements on a 7-point Likert scale from 1 (strongly disagree) to 7 (strongly agree). Participants agreed that the system worked as expected ($M = 5.57, SD = 1.40$) and that the visualization of the notifications was appealing ($M = 5.00, SD = 1.73$). They had privacy concerns ($M = 5.43, SD = 1.81$) but found the provided privacy levels to be sufficient ($M = 5.57, SD = 2.15$). However, they disagreed that the color-coding provided privacy ($M = 3.86, SD = 2.61$). Overall, the usefulness of the system was rated as neutral to positive ($M = 4.43, SD = 2.37$).

In free text fields, we asked the participants why they chose their corresponding default privacy level. Participants all agreed that they do not want to share the information in the notifications with others. One participant mentioned that he chose the level because his colleagues chose the level as well. All participants agreed that displaying all content is only appropriate for private displays, e.g., at home. One participant mentioned that showing full content would only work if it can be ensured that no sensitive personal data is being displayed. Displaying reduced content or only app names worked best for all but one participant. They agreed that these options provide them just enough information hints to know whether the notification is important.

6.2.3.3 Semi-Structured Interviews

All but one participant kept their chosen privacy level after the initial setup, as they found that it worked for all kinds of content. P1 and P2 mentioned that they would feel comfortable to show more content in open spaces with more people. P2 stated that over time he learned which color belongs to which user. Although the privacy levels were sufficient, he would be more comfortable with displaying more content if word filters were available. P7 stated that she was concerned because the system makes things written by other people public, which resulted in her initially clearing her notifications more often.

P3 thought more about his notifications when walking towards a public display to avoid embarrassing notifications. Participants liked not having to take their phones out of their pockets. A side effect of the system was that participants left their phones more often in their offices (P2, P3, P7). According to P7, the system made people more accepting of the fact that "you are not responsive all the time."

Participants sometimes forgot to carry their beacon or smartphone (P2, P3, P4, P7). In case of the smartphone, participants told us about their positive and negative experiences. The system allowed them to see their notifications when they forgot their smartphones in their offices. However, without their smartphones, participants were unable to dismiss unwanted notifications. P7 disliked having to carry the beacon. When asked if the detection should work with the smartphone only, she replied that, ideally, she should not have to carry anything.

P3 told us that the notifications were a topic to talk about in the meeting areas. However, P7 found that it led to awkwardness because people asked why she has a certain app or did not finish a task yet (to-do app visible). P5 explained that he once terminated a conversation in the kitchen because he saw a notification on the display.

Nearly all participants were interested in using the system as a pervasive information display at home (P2, P3, P4, P6, P7). P2 suggested being able to share media on the public display when nearby. It could be used as a "public whiteboard" to share messages with colleagues (P2, P5, P7). P1 and P2 wished to interact with the public display directly, e.g., using touch to dismiss notifications. Participants suggested displaying some information persistently, e.g., weather forecasts, public transport information, smartphone battery levels, and upcoming events.

6.3 Conclusion

In the first part of this chapter, we developed guidelines for notifications on smart TVs (RQ5). Through a set of three focus groups, we collected insights about users' attitude towards notifications on TVs. The design space includes the presentation of notifications, the displayed content, the application causing the notification, the number of received notifications, and how long a notification stays on the screen. We further studied selected design alternatives in an online survey to get more information about the displayed content of notifications on smart TVs. With these findings, we implemented an application which enables us to display notifications on the TV while a video is playing and conducted a lab study. In the lab study, we investigated the difference in the settings between watching alone and watching together with other people. From the findings, we have elaborated our design guidelines for displaying notifications on a TV. Only notifications truly important for the user should be shown. Furthermore, users' privacy should be considered, especially if multiple people share the TV. Notifications could mainly be shown during breaks and be presented in a subtle way. Finally, users should be enabled to customize the presentation easily. In the future, further insights could be gained by implementing a system that shows notifications on smart TVs and conducting a field study by installing the system in peoples' living rooms. In particular, it would be interesting to use a system that is able to determine the number of viewers, for example, through the use of depth-sensing cameras. The system could adjust the settings and types of notifications shown according to the viewers. Furthermore, means to interact with notifications shown on smart TVs should be investigated. Important notifications often inform about messages and users, therefore, might expect that they can directly react to them using the smart TV.

A further direction is ambient visualizations that subtly display notifications. A potential approach is to use technologies such as Ambilight and IllumiRoom [68] that allow visualizations in the surrounding of the TV.

In the second part of the chapter, we presented *PD Notify*, a system to investigate personal content on public displays (RQ6). The system mirrors the user's pending smartphone notifications on nearby public displays, enabling us to test a variety of different content from instant messages to calendar appointments. Users can change the level of detail that is shown on the displays using global and app-specific privacy settings. We conducted an in-situ study in a semi-public work environment, where we deployed two public displays in popular meeting areas. Seven co-workers used the system for three weeks, and we conducted subsequent semi-structured interviews with the participants. The results of this first deployment show that displaying personal content on public displays is not only feasible but also valued by users. Participants limited the display of personal content regardless of the content category. They initially tested various privacy settings but quickly settled on one setting that they were comfortable with for all kinds of content, with app-specific settings being an exception. An important finding is that no participant allowed all content to be shown on the public displays. Most participants favored displaying reduced content or only the names of apps. While participants liked the system, they did not want to spend time configuring it. This raises important implications and interesting discussion points regarding personal content on public displays, e.g., the need for reasonable default settings.

7

Notification Logging Framework

A common theme in studies on notifications is the need for accessing users' notifications, often for logging purposes. Based on the work in the previous chapters, we present *Notification Log*, an open-source framework for logging notifications on mobile devices. We implemented *Notification Log* as a modular Android app that can be easily extended. The framework has been used in multiple in-the-wild and in-lab user studies, and has been downloaded by over 400,000 users. By making the framework publicly available, we hope to accelerate the research of notification experiences that are valued by users while respecting their digital well-being. In the following, we introduce *Notification Log*'s architecture and provide an overview of prior application scenarios.

Parts of this chapter are based on the following publications:

D. Weber, A. Voit, and N. Henze. "Notification Log: An Open-Source Framework for Notification Research on Mobile Devices." In: *Proceedings of the 2018 ACM Interna-*

tional Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers. UbiComp '18. Singapore, Singapore: ACM, 2018, pp. 1271–1278. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274118

7.1 Architecture

We implemented the *Notification Log* framework as an Android app with the primary goal of running in the background of personal mobile devices for both in-the-wild and in-lab studies. The requirements for the app are reliable and unobtrusive logging in the background, extensibility, as well as support for most Android versions and devices.

7.1.1 Data Sources

The *Notification Log* framework consolidates multiple data sources and provides an abstraction layer for a wide range of Android versions (see Figure 7.1).

Notification Listener Service The central data source of the framework is the Notification Listener Service API [7]. This service is, after granting permission from the user, permanently running in the background of the device and receives callbacks when a notification is added or removed from the system. Recent versions of Android significantly improved the information provided by this API, e.g., by providing information if a notification was removed by the user or the notifying app itself. The API is available since Android 4.3, which runs on 98.40% of Android smartphones and tablets at the time of writing [6]. In particular, this service provides which apps triggered/removed notifications, the text content, priority/importance levels, vibration patterns, and sound amongst a multitude of additional attributes.

Device Metadata and Context Data *Notification Log* samples device metadata, such as the screen state (on/off), ringer mode (silent, vibration, volume), battery state (current level, if charging), and connectivity state (Wi-Fi, mobile, offline). This information is combined with the notification data to provide insights into the device's context when a notification was received or removed.

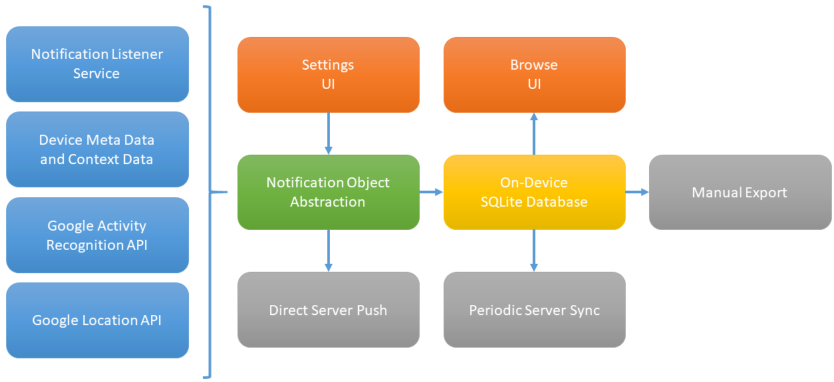


Figure 7.1: The architecture of *Notification Log*. The framework provides an abstraction layer for unifying notification data with device metadata and context data. The logs can be further extended by including additional data sources, such as the Google Activity Recognition API and the Google Location API. The logged data is stored in an on-device database and can be manually exported, synced periodically or directly pushed to a server. Users can control the logging with an extensible settings screen and browse the locally stored data.

Additional Data Sources Other data sources can be easily integrated due to the extensible architecture of the framework. In the past, we used the Google Activity Recognition API to extend the notification data. The API reports probabilities for if the device is still or the user is walking, running, cycling, or driving. Another data source is the Google Location API, which reports location updates including longitude, latitude, estimated accuracy in meters, and the age of the location update. Both data sources can be attached to the Notification Listener Service and enhance the logged notification data. Notably, the framework allows the use of these data sources without affecting the device's performance or battery consumption negatively.

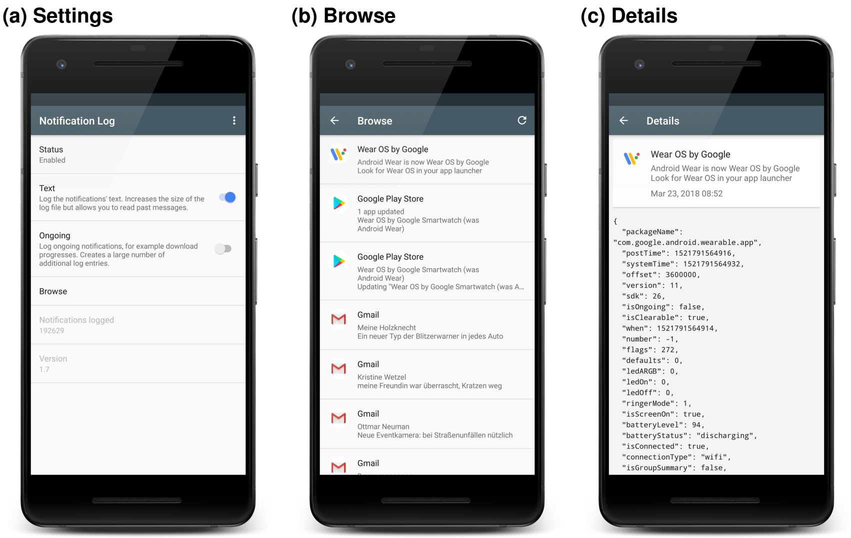


Figure 7.2: Left: The settings screen allows users to control if and how notifications should be logged. The screen is extensible and allows for a simple integration of additional options. Middle: The browse screen allows users to explore the on-device database. It provides a preview of the logged notifications. Right: The details screen shows a preview of a logged notification and the corresponding JSON representation.

7.1.2 Data Consolidation

Notification Log consolidates all data sources in a central Notification Object (highlighted in green in Figure 7.1). This object acts as an abstraction layer that hides differences in the Android SDK. The result of the data consolidation is serialized to a JSON representation.

7.1.3 Data Persistence

Notification Log stores all unified JSON objects in a private on-device SQLite database (highlighted in yellow in Figure 7.1). The database is inaccessible by other apps on the device and therefore enables secure logging of notification and context data.

7.1.4 Data Processing

The data stored in the local database is the core of the framework. The *browse* user interface allows users to see recent additions to the log (see Figure 7.2b). It shows a preview of the recorded notification and the JSON representation of the notification and context data (see Figure 7.2c). The database can also be exported manually in the JSON format for computation by other applications. Events can be sent to a server either immediately when a notification is added or removed from the system or batched for synchronization in specific intervals when pre-defined conditions are met, e.g., when the device is plugged in and connected to a Wi-Fi network.

7.1.5 Extensibility

The framework can be extended in multiple ways, e.g., by adding additional data sources to the notification object abstraction layer. The existing UIs can be extended by further options that control the logging (see Figure 7.2a) or statistics that provide additional insights about the logged data.

7.2 Application Scenarios

The *Notification Log* framework was used in a number of in-the-wild and in-lab studies. For each of these studies, the framework was extended according to the needs of the study. In the following, we report prior applications of the framework and detail the release on the Google Play Store.

7.2.1 Record and Replay of Notifications

Lab studies that involve personal notifications are challenging. When using participants' personal devices, it is not possible to control how many notifications they receive during the study, as this mostly depends on external factors. On the other hand, using fake notifications may be perceived differently by the participants. As a compromise between those approaches, *Notification Log* was used in a lab study setup as described in Chapter 6. The app was installed on the participants' personal smartphones days prior to the lab study. On the day of the study, the recorded notifications were exported, and a set of randomly selected notifications were then used in the lab study. By displaying participants' own recorded notifications, it was possible to create a compromise between meaningful notifications while controlling the number of notifications shown during the study.

7.2.2 Reflection on Mobile Notifications

For the *Notification Dashboard*, described in Chapter 4, the framework was used to enable reflection on mobile notifications. In current mobile operating systems, notifications are ephemeral. To provide insights on how many notifications users receive on a daily basis, *Notification Log* was used to record all notifications on the users' personal smartphones for a specific time span. The log data was then exported and loaded in the web-based dashboard. The dashboard breaks down the number of notifications that were created in the time span, along with the apps that created them, the change over time, and the differences between weekdays. The dashboard allows users to reflect on the notifications they receive and enables them to adjust notification settings to improve their digital well-being. In this specific study, the logs were exported manually, but it is easy to imagine a system in which the notifications are periodically synced with a server to provide an always up-to-date dashboard experience.

7.2.3 Integration in Existing Infrastructures

Notification Log was integrated into an existing infrastructure for Internet of Things (IoT) devices in intelligent living environments [77]. Thanks to the loose coupling of the components, it was possible to quickly integrate the Notification

Listener Service and the Notification Object Abstraction layer into an existing project. The existing IoT infrastructure was, therefore, extended by the possibility to react to notifications received on personal smartphones and tablets.

7.2.4 Novel Experiences

For the *PD Notify* project, described in Chapter 6, the framework was extended to communicate with nearby public displays. Participants of the study carried Bluetooth beacons. When being near a public display, the modified *Notification Log* app would mirror the pending notifications of the participants' personal smartphones on the public display. For this study, the framework was extended by a component that handles the communication with the public display and an additional *privacy* screen that allowed the participants to control the level of detail of the mirrored notifications.

7.3 Open-Source Framework

The base version of *Notification Log* has been available in the Google Play Store since July 2015. At the time of writing, the app has been downloaded over 500,000 times and is installed on over 30,000 Android devices. These downloads originated from 175 countries. Users installed the app on over 360 different Android device models, with Android versions ranging from 4.3 to 12.

Apart from being used by thousands of users over the span of several years, the *Notification Log* framework was used in a number of in-the-wild and in-lab studies. Due to its extensible architecture, it can be quickly customized for many application scenarios. We open-sourced the *Notification Log* framework under the MIT license to provide researchers and developers with a flexible framework for notification-related research and projects. Since the release of the framework on GitHub[1] in August 2018, the project was forked over 40 times. One user changed the framework from a standalone app to a library, which can

[1] `https://github.com/interactionlab/android-notification-log`

be integrated into other apps. Another user replaced the SQLite database with a cloud-hosted database that automatically synchronizes logged notifications with a remote server.

Researchers started using the framework for novel applications as well. Silva et al. developed a system for notification management in multi-device and IoT environments [144, 145]. The researchers based the notification collector component of their system on *Notification Log* and extended it to forward notifications to a decision-making module. Rzayev et al. explored notifications in virtual [135] and augmented [134] reality. The researchers used *Notification Log* to collect "realistic" notifications to display them in virtual and augmented reality focused user studies. Finally, Cho et al. built an app that enables users to share private status messages triggered by specific notifications [22], which is built on top of *Notification Log* as well.

In the future, the framework should be adapted for other platforms and device types as well. This would enable a comprehensive picture of all notifications that users receive on their devices, a foundation for optimizing notifications in ubiquitous computing environments.

8

Conclusion and Future Work

Thirty years after Mark Weiser outlined the vision of ubiquitous computers [186], we are surrounded by a large number of "smart" devices in our daily lives. In his work, Weiser proposed that ubiquitous computers would come in different sizes, such as tabs, pads, and boards. Nowadays, smartphones, smartwatches, tablet computers, laptops, smart TVs, and public displays have become ubiquitous, as have wireless technologies that connect them. This resulted in notifications becoming ubiquitous as well. Devices can provide users proactively with information using multiple modalities. Users generally value notifications, but notifications can also lead to interruptions and adverse effects. As ubiquitous computing environments expand, we have to be careful about not amplifying these adverse effects. In this thesis, we presented our work on the empirical assessment and improvement of ubiquitous notifications.

In Chapter 1, we motivated the thesis and defined the research questions (RQs). We described the research challenges, methodology, and evaluation, as well as the research context. In Chapter 2, we provided background information and a review of related work in the field of notifications. In Chapter 3, we focused on the assessment of mobile notifications on smartphones. In the first part of the chapter, we reported the results of a large-scale in-the-wild study on smartphone notification drawers. We showed how smartphone notifications materialize in the notification drawer throughout the day and proposed three user types for managing notifications in the notification drawer. In the second part of the chapter, we described the architecture of *Annotif*, a privacy-aware annotation tool for notifications. The tool allows participants of user studies to share and annotate their own personal notifications with researchers. Using this tool, we reported the results of a case study that focused on assessing the importance and urgency of smartphone notifications. In Chapter 4, we followed up on the management of smartphone notifications. We first described the *Notification Dashboard*, a tool that helps users reflect on the notifications that they receive on a daily basis. We outlined the implementation as well as a small evaluation study. In the second part of Chapter 4, we investigated new means of interacting with smartphone notifications. We described the development of the *NHistory* app, which allows users to temporarily snooze notifications for a duration or to a point-in-time. Using this app, we conducted a large-scale in-the-wild study to explore how users defer notifications manually. We followed up with a more controlled in-situ study and conducted subsequent semi-structured interviews. In Chapter 5, we expanded the set of considered smart devices by including smartwatches, tablets, and laptops/ PCs. We conducted a quantitative multi-device in-situ study about device usage and the opportunity of showing notifications on different kinds of smart devices. We followed up on this with a qualitative study on multi-device notification environments by conducting semi-structured interviews and considering even more devices. Chapter 6, expanded the set of devices even further by including large and pervasive displays such as smart TVs and public displays. In the first part of the chapter, we reported a detailed exploration of notifications on smart TVs. We conducted focus groups to create a design space, evaluated multiple notification variants in an online survey, and finally conducted a lab study with an implemented prototype. From the learnings throughout this process, we derived design guidelines for notifications on smart TVs. In the second part of the chapter, we introduced *PD Notify*, a system that can mirror smartphone notifications to nearby public displays. We built the system with privacy controls to investigate the feasibility of showing personal notifications on public displays. We conducted an in-situ study in a semi-public work environment and reported our findings. Finally, in Chapter 7, we described the *Notification Log* framework for logging mobile notifications that was used throughout this thesis. We outlined the architecture and exemplary application scenarios. We open-sourced the framework to support notification research and provided a brief overview of the work in which the framework was already used. The work conducted in this thesis can support researchers and developers of future smart notification systems in ubiquitous computing environments.

8.1 Summary of Research Contributions

In the following, we summarize the research contributions of this thesis and answer the research questions. We then provide an outlook on future work.

Assessing Notifications on Mobile Devices

RQ1: *How do notifications materialize on smartphones, and how are users managing them?* In Chapter 3, we reported our findings from periodically sampling notification drawers in a large-scale in-the-wild study. Prior work already investigated how many notifications users receive on a daily basis. However, how these notifications materialize in notification drawers was still an open research question. We collected 8.8 million notification drawer snapshots from almost 4,000 devices. We found that users have, on average, 3.4 notifications in the notification drawer. We saw notifications accumulate overnight as users are sleep, resulting in more notifications in the morning. *SMS & IM* notifications dominate the first position in notification drawers, both due to the large number of communication-related notifications users receive and the notification prioritization in Android. Based on the collected data, we proposed the existence of three user types. *Frequent Cleaners* address notifications quickly and try to keep the notification drawer empty. *Notification Regulators* receive an increased amount of notifications but overall keep them under control. Finally, *Notification Hoarders* accumulate notifications over time and dismiss them all at once. These findings enable a better understanding of how notifications materialize in mobile notification drawers. To complement this research contribution, we open-sourced the collected notification drawer data set.

RQ2: *How can we assess notifications in detail while respecting users' privacy?* In Chapter 3, we introduced *Annotif*, a privacy-aware system for unobtrusively assessing mobile notifications in user studies. The system encrypts notifications on participants' smartphones and allows them to annotate the notifications and censor content before granting access to researchers. We used the system in an in-situ case study to assess the importance and urgency of mobile notifications. The system allowed the participants to rate the importance and urgency of their notification, in addition to providing additional context information. In the case study, we saw an annotation coverage of 93%. Participants rated 39% of the annotated notifications as not important and over half (52%) as non-urgent. We were positively surprised that participants rarely censored all text and instead focused on protecting the names of their contacts. Being able to see the text allowed us to differentiate between 1:1 and group chat notifications. We saw differences with regards to importance and urgency depending on the app that triggered a notification and whether the notification contained the name of the participant or a contact. These detailed insights allowed us to identify four *Notification Clusters*. We saw critical notifications that require the users' immediate attention. Only a few notifications per day are part of this cluster, and they are easy to miss when working with notification data sets with thousands of notifications. We also saw a large number of low priority notifications that were rated as neither important nor urgent. Finally, we saw a large number of medium and high priority notifications whose cluster might change depending on the content of context, which we consider an important research opportunity for future work.

Improving the Management of Mobile Notifications

RQ3: *How can we support users with managing mobile notifications?* We presented two approaches for managing mobile notifications. In the first part of chapter Chapter 4, we introduced the *Notification Dashboard*. The system supports user by allowing them to reflect on the notifications that they receive on a daily basis. We evaluated the system in a small user study and outlined opportunities for improvement. One particularly interesting insight was that participants underestimated the number of notifications they receive daily. In the second part of the chapter, we introduced *NHistory*. The app extends the Android system by allowing users to “snooze” notifications for a user-defined duration or to a point-in-time. Using *NHistory*, we conducted a large-scale in-the-wild and a more controlled in-situ study with subsequent semi-structured interviews. From analyzing the snooze behavior in both studies and the qualitative feedback, we derived design implications for deferring notifications. We saw peaks of snoozed notifications to before and after work hours, as well as lunch breaks. Participants snoozed notifications if they were out-of-context, such as personal notifications at work, if they are unable to attend notifications, or simply were not in the mood. One particular important insight was that participants considered notifications as temporary. Snoozing for more than two days was an exception, a finding that can be used as an upper bound in future notification systems. As in previous studies, we saw the importance of communication notifications. Deferring these notifications should be considered carefully, as well as including the user’s context and daily routines.

Beyond Mobile Notifications

RQ4: *How do various types of personal devices differ in multi-device environments with regards to displaying notifications?* In Chapter 5, we reported a quantitative and a qualitative study on notifications in multi-device environments. Our approach was to use a dedicated device for triggering ESM questionnaires to reduce the study’s impact on device usage. We investigated differences between smartphones, smartwatches, tablets, and desktop PCs/ laptops. The results showed that the smartphone was the most important device to be notified on. Interestingly, smartwatches showed similar results, although none of the participants in the studies owned a smartwatch. Still, the smartwatches were regarded as a suitable device to be notified on, with the exception of users who dislike wearing watches. Both smartphones and smartwatches were considered as suitable for receiving notifications throughout the day. In contrast, desktop PCs and laptops were only regarded as suitable when in active use. Tablet computers were not considered as suitable devices to be notified on. Instead, they were regarded as media-consumption and gaming devices and are often left at home. We saw similar feedback for gaming consoles, ebook readers, and smart TVs. Participants did not consider these devices as relevant for receiving notifications, with the notable exception of system updates and low-battery warnings. The findings of this chapters show that not only the timing of notifications is important but also the device to show the notification on. Being smart about which device to notify users on can reduce device-switching and, therefore, can reduce interruptions and adverse effects.

RQ5: *What are the considerations when displaying notifications on smart TVs?* Smart TVs are similar to other smart devices, such as smartphones, in the sense that they are typically always connected and can be extended by installing apps from app stores. However, notifications on smart TVs are not common yet. In Chapter 6, explored notifications on smart TVs and reported insights gained from focus groups, an online survey, and a lab study. From the study results, we derived design guidelines. Participants clearly stated that notifications on smart TVs should always be optional. Unlike other smart devices, smart TVs are often used as shared devices by multiple people at the same time. This should be considered when showing notifications on smart TVs. One suggestion was to automatically disable notifications if multiple users are detected, or at least making it easy to disable notifications. We discussed opportunities for showing notifications at breakpoints, e.g., during ad breaks, after a series episode or movie has ended, or when pausing content. A particular important aspect is minimizing distractions. The size, opacity, display duration, and text length of notifications should balance readability and the occlusion of content. Finally, we reported the most popular notification settings from the lab study that can be used as a suggestion for default settings for notifications on smart TVs. However, participants stressed that even with reasonable default settings allowing customization is important.

RQ6: *What are the considerations when displaying notifications on public displays?* While public displays are getting more and more ubiquitous, they currently rarely show personal content. In Chapter 6, we introduced *PD Notify*. The system allows users to mirror smartphone notifications to nearby public displays. The amount of content shown on the public displays can be controlled using the *PD Notify* app. Using this system, we conducted an in-situ study in a semi-public work environment. We received positive feedback about the system. Participants valued not having to carry their smartphones and still being able to see their notifications on nearby public displays. However, participants limited the content shown on the public displays. They initially tested various privacy settings and quickly settled on a setting that works for them. Most participants preferred to show reduced content or only the names of apps. While participants liked the system, they did not want to spend time configuring it. This highlights the need for reasonable default settings.

Notification Logging Framework

We developed the *Notification Log* framework for logging mobile notifications as an Android app. The framework described in Chapter 7 was used as the basis for most of the work conducted in this thesis. It provides a compatibility layer for Android versions and combines notifications with device metadata and context information, such as the device state, user activity, and location information. The framework can be easily extended. We open-sourced the framework, and it is already used by other researchers in the field of notification research.

8.2 Future Work

In the following, we discuss opportunities for future work on assessing the expanding set of smart devices, challenges of multi-device-aware notifications, and the need for reasonable defaults and notification middlewares.

Investigating the Expanding Set of Smart Devices

In this thesis, we mostly focused on investigating smart devices that can be described as rectangular screens. However, the field of smart devices is continuously expanding and not limited to devices with traditional screens. While the majority of this work was conducted, smart speakers [160] and smart displays gained popularity in the smart home. Users interact with these devices by talking to digital voice assistants. In the first iterations, these devices were limited to request-response interactions - typically triggered by a wake word. However, we are slowly seeing these devices to evolve to provide proactive information, for example for reminders. Traditional wired headphones and earphones are changing as well. They are losing the wire and instead are connected to smartphones using Bluetooth. In recent works, they are sometimes referred to as "hearables," a subcategory of wearables. Some of these devices gained the ability to announce incoming calls and new text messages using text-to-speech. Coupled with digital voice assistants, these devices allow users to receive and react to notifications without having to look at and interact with a screen.

In terms of wearables, we briefly investigated smartwatches in Chapter 5. However, the field of wearable devices is expanding as well. Next to smartwatches, fitness trackers, often with no or a limited display, are popular devices. Many fitness trackers allow notifying users by using vibration patterns. Further, while head-mounted displays, or smartglasses, did not yet gain mainstream adoption, we are seeing first explorations of such devices for consumers. For instance, one of the core features of the *Google Glass* smartglasses was displaying notifications [103]. Currently, other types of head-mounted displays are gaining popularity as well, specifically devices that enable virtual reality [135] and augmented reality [134].

Finally, the number of Internet of Things (IoT) devices is expanding as well. In the smart home, we see more and more devices that "vanish into the background," as predicted by Mark Weiser [186]. Some of these devices are able to notify users themselves, while others act as sensors that trigger notifications on other devices and surfaces [168].

To conclude, the list of devices investigated in this thesis is not exhaustive and the set of smart devices is continuously expanding. Improving this changing environment is challenging. Future work should focus on creating a taxonomy of smart devices, including the capabilities of the devices, the modalities available for notifying users, and how the devices are used.

Towards Multi-Device-Aware Notifications

A major challenge and future research question is managing the increasing complexity of notifications in ubiquitous computing environments. The source a of notification might be the device itself, a push event from a server, or another device. The trigger of the notification might be self-initiated (e.g., alarm clock or reminder), determined by an algorithm, contextual data (e.g., time or location), an external event, or other users. It might refer to an event that is happening right now (e.g., an incoming call), an event in the past (e.g., a previously received instant message), or in the future (e.g., a reminder for an upcoming calendar event) – all with varying levels of importance and urgency. The notification might be device-related (e.g., low battery warnings) or unrelated to the device (e.g., communication). Notifications might be shown on a single device, targeted to a subset of devices, or broadcasted to all of a users' devices. They might also be shown on a single or subset of devices and then mirrored on other devices. The devices have different properties that might allow for different reactions, such as dismissing, snoozing, or reacting to notifications. Not all devices offer the same modalities and are equally suitable for showing certain notifications. This can also create new interaction patterns, for example, users working on one device, being notified on another device, and attending the notification on a third device. This mesh of devices should also be considered when evaluating what makes a notification disruptive. In Chapter 4, we heard from participants that they sometimes silence their smartphones to avoid interruptions. With multi-device notifications, these notifications might appear on multiple devices, which has implications for interruptions and adverse effects. However, this could also be used to alert users about urgent notifications by intentionally broadcasting notifications to multiple devices. We also have to consider privacy aspects as some devices might be personal; others might be shared or work-issued.

When implementing notifications and developing new notification systems, researchers and developers should keep the complexity of notifications in mind. A fundamental requirement for this is cross-device authentication and reliable cross-device data synchronization. Future work opportunities are creating standards and protocols for multidevice-aware notifications that allow devices to work better together.

Default Settings and Notification Middleware

On many devices, notifications can be configured. The default settings are often set by the operating system or manufacturer. As the set of different devices around us continues to expand, we need means to adapt these settings across devices to avoid inconsistencies. In the interviews conduced as part of this thesis, we heard that many users dislike configuring and tweaking notification settings. This is going to be more challenging as users are surrounded by more and more devices. An option to assist users would be the development of a common notification middleware that adapts notification settings for users across devices. This middleware could take on other tasks as well. Similar to email spam filters, a multi-device-aware middleware could help users with sorting their notifications. The middleware could automatically create summaries or clear outdated notifications based on time or context data. Such as middleware could ensure that devices respect the users' notification preferences while providing the right information at the right time on the right device.

Bibliography

[1] Y. Abdelrahman, P. W. Woźniak, P. Knierim, D. Weber, K. Pfeuffer, N. Henze, A. Schmidt, F. Alt. "Exploring the Domestication of Thermal Imaging." In: *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia*. MUM '19. Pisa, Italy: ACM, 2019, 9:1–9:7. ISBN: 978-1-4503-7624-2. DOI: 10.1145/3365610.3365648 (cit. on p. 24).

[2] P. D. Adamczyk, B. P. Bailey. "If Not Now, when?: The Effects of Interruption at Different Moments Within Task Execution." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '04. Vienna, Austria: ACM, 2004, pp. 271–278. ISBN: 1-58113-702-8. DOI: 10.1145/985692.985727 (cit. on pp. 42, 44, 187).

[3] M. Alaoui, M. Lewkowicz. "A livinglab approach to involve elderly in the design of smart TV applications offering communication services." In: *International Conference on Online Communities and Social Computing*. Springer. 2013, pp. 325–334 (cit. on p. 163).

[4] F. Alt, T. Kubitza, D. Bial, F. Zaidan, M. Ortel, B. Zurmaar, T. Lewen, A. S. Shirazi, A. Schmidt. "Digifieds: Insights into Deploying Digital Public Notice Areas in the Wild." In: *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia*. MUM '11. Beijing, China: ACM, 2011, pp. 165–174. ISBN: 978-1-4503-1096-3. DOI: 10.1145/2107596.2107618 (cit. on p. 188).

[5] C. Anderson, I. Hübener, A.-K. Seipp, S. Ohly, K. David, V. Pejovic. "A Survey of Attention Management Systems in Ubiquitous Computing Environments." In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2.2 (July 2018), 58:1–58:27. ISSN: 2474-9567. DOI: 10.1145/3214261 (cit. on p. 45).

[6] Android Developers. *Android Distribution Dashboard*. https://developer.android.com/about/dashboards/. 2021 (cit. on pp. 55, 58, 202).

[7] Android Developers. *Notification Listener Service API reference*. https://developer.android.com/reference/android/service/notification/NotificationListenerService. 2021 (cit. on pp. 55, 75, 113, 202).

[8] Android Developers. *Notifications UX: What's New for Android O (Google I/O '17)*. https://youtu.be/vwZi56IOMiO?t=1234. 2017 (cit. on pp. 65, 70).

[9] J. H. Aranda, S. Baig. "Toward "JOMO": The Joy of Missing out and the Freedom of Disconnecting." In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '18. Barcelona, Spain: ACM, 2018, 19:1–19:8. ISBN: 978-1-4503-5898-9. DOI: 10.1145/3229434.3229468 (cit. on pp. 42, 94, 160).

[10] J. Auda, D. Weber, A. Voit, S. Schneegass. "Understanding User Preferences Towards Rule-based Notification Deferral." In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI EA '18. Montreal QC, Canada: ACM, 2018, LBW584:1–LBW584:6. ISBN: 978-1-4503-5621-3. DOI: 10.1145/3170427.3188688 (cit. on pp. 22, 24, 43, 93, 135).

[11] D. Avrahami, S. R. Fussell, S. E. Hudson. "IM Waiting: Timing and Responsiveness in Semi-synchronous Communication." In: *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*. CSCW '08. San Diego, CA, USA: ACM, 2008, pp. 285–294. ISBN: 978-1-60558-007-4. DOI: 10.1145/1460563.1460610 (cit. on pp. 36, 37).

[12] B. P. Bailey, J. A. Konstan, J. V. Carlis. "Measuring the effects of interruptions on task performance in the user interface." In: *Smc 2000 conference proceedings. 2000 ieee international conference on systems, man and cybernetics.'cybernetics evolving to systems, humans, organizations, and their complex interactions'(cat. no. 0*. Vol. 2. IEEE. 2000, pp. 757–762 (cit. on p. 42).

[13] A. Battestini, V. Setlur, T. Sohn. "A Large Scale Study of Text-messaging Use." In: *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI '10. Lisbon, Portugal: ACM, 2010, pp. 229–238. ISBN: 978-1-60558-835-3. DOI: 10.1145/1851600.1851638 (cit. on p. 36).

[14] F. Bentley. "Losing the Power of Defaults: Exploring Changing Patterns in Enabling Mobile Notifications." In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. UbiComp '18. Singapore, Singapore: ACM, 2018, pp. 1208–1214. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274109 (cit. on p. 72).

[15] J. Birnholtz, J. Hancock, M. Smith, L. Reynolds. "Understanding Unavailability in a World of Constant Connection." In: *Interactions* 19.5 (Sept. 2012), pp. 32–35. ISSN: 1072-5520. DOI: 10.1145/2334184.2334193 (cit. on p. 42).

[16] M. Böhmer, B. Hecht, J. Schöning, A. Krüger, G. Bauer. "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*. MobileHCI '11. Stockholm, Sweden: ACM, 2011, pp. 47–56. ISBN: 978-1-4503-0541-9. DOI: 10.1145/2037373.2037383 (cit. on p. 62).

[17] Cambridge Dictionary. *Notification*. https://dictionary.cambridge.org/dictionary/english/notification. 2022 (cit. on p. 27).

[18] Cambridge Dictionary. *Push Notification*. https://dictionary.cambridge.org/dictionary/english/push-notification. 2022 (cit. on p. 27).

[19] M. E. Cecchinato, A. L. Cox, J. Bird. "Always On(Line)?: User Experience of Smartwatches and Their Role Within Multi-Device Ecologies." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: ACM, 2017, pp. 3557–3568. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025538 (cit. on p. 39).

[20] N. Chang, M. Irvan, T. Terano. "A TV program recommender framework." In: *Procedia Computer Science* 22 (2013), pp. 561–570 (cit. on p. 164).

[21] Y.-J. Chang, Y.-J. Chung, Y.-H. Shih, H.-C. Chang, T.-H. Lin. "What Do Smartphone Users Do when They Sense Phone Notifications?" In: *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. UbiComp '17. Maui, Hawaii: ACM, 2017, pp. 904–909. ISBN: 978-1-4503-5190-4. DOI: 10.1145/3123024.3124557 (cit. on p. 43).

[22] H. Cho, J. Oh, J. Kim, S.-J. Lee. "I Share, You Care: Private Status Sharing and Sender-Controlled Notifications in Mobile Instant Messaging." In: *Proc. ACM Hum.-Comput. Interact.* 4.CSCW1 (May 2020). DOI: 10.1145/3392839 (cit. on p. 208).

[23] K. Church, R. de Oliveira. "What's Up with WhatsApp?: Comparing Mobile Instant Messaging Behaviors with Traditional SMS." In: *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*. MobileHCI '13. Munich, Germany: ACM, 2013, pp. 352–361. ISBN: 978-1-4503-2273-7. DOI: 10.1145/2493190.2493225 (cit. on pp. 36, 37, 42, 94, 108, 133).

[24] J. Cohen. *Statistical power analysis for the behavioral sciences*. Routledge, 2013 (cit. on p. 148).

[25] S. Consolvo, M. Walker. "Using the experience sampling method to evaluate ubicomp applications." In: *IEEE pervasive computing* 2.2 (2003), pp. 24–31 (cit. on pp. 73, 140).

[26] F. Corno, L. De Russis, T. Montanaro. "A context and user aware smart notification system." In: *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. IEEE. 2015, pp. 645–651 (cit. on p. 109).

[27] C. Courtois, E. D'heer. "Second Screen Applications and Tablet Users: Constellation, Awareness, Experience, and Interest." In: *Proceedings of the 10th European Conference on Interactive TV and Video*. EuroITV '12. Berlin, Germany: ACM, 2012, pp. 153–156. ISBN: 978-1-4503-1107-6. DOI: 10.1145/2325616.2325646 (cit. on p. 164).

[28] M. Csikszentmihalyi, R. Larson. "Validity and reliability of the experience-sampling method." In: *Flow and the foundations of positive psychology*. Springer, 2014, pp. 35–54 (cit. on p. 140).

[29] M. Czerwinski, E. Cutrell, E. Horvitz. "Instant messaging: Effects of relevance and timing." In: *People and computers XIV: Proceedings of HCI*. Vol. 2. 2000, pp. 71–76 (cit. on p. 37).

[30] M. Czerwinski, E. Cutrell, E. Horvitz. "Instant messaging and interruption: Influence of task type on performance." In: *OZCHI 2000 conference proceedings*. Vol. 356. 2000, pp. 361–367 (cit. on pp. 16, 37, 39).

[31] M. Czerwinski, M. Czerwinski, E. Horvitz, S. Wilhite. “A Diary Study of Task Switching and Interruptions.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’04. Vienna, Austria: ACM, 2004, pp. 175–182. ISBN: 1-58113-702-8. DOI: 10.1145/985692.985715 (cit. on pp. 39, 42, 43, 51).

[32] A. K. Dey, K. Wac, D. Ferreira, K. Tassini, J.-H. Hong, J. Ramos. “Getting Closer: An Empirical Investigation of the Proximity of User to Their Smart Phones.” In: *Proceedings of the 13th International Conference on Ubiquitous Computing*. UbiComp ’11. Beijing, China: ACM, 2011, pp. 163–172. ISBN: 978-1-4503-0630-0. DOI: 10.1145/2030112.2030135 (cit. on p. 162).

[33] T. Dingler, M. Pielot. “I’ll Be There for You: Quantifying Attentiveness Towards Mobile Messaging.” In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI ’15. Copenhagen, Denmark: ACM, 2015, pp. 1–5. ISBN: 978-1-4503-3652-9. DOI: 10.1145/2785830.2785840 (cit. on pp. 37, 38, 71).

[34] T. Dingler, D. Weber, M. Pielot, J. Cooper, C.-C. Chang, N. Henze. “Language Learning On-the-go: Opportune Moments and Design of Mobile Microlearning Sessions.” In: *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI ’17. Vienna, Austria: ACM, 2017, 28:1–28:12. ISBN: 978-1-4503-5075-4. DOI: 10.1145/3098279.3098565 (cit. on pp. 24, 37, 45, 53, 110).

[35] Evernote. *Android-Job*. https://github.com/evernote/android-job. 2019 (cit. on p. 55).

[36] A. Exler, M. Braith, A. Schankin, M. Beigl. “Preliminary Investigations About Interruptibility of Smartphone Users at Specific Place Types.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1590–1595. ISBN: 978-1-4503-4462-3. DOI: 10.1145/2968219.2968554 (cit. on p. 43).

[37] A. Exler, M. Budde, E. Pescara, A. Schankin, T. Riedel, M. Beigl. “Challenges and Lessons from Working with Data Collected by Crowdfunding in the Wild.” In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. UbiComp ’18. Singapore, Singapore: ACM, 2018, pp. 618–623. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3267545 (cit. on p. 49).

[38] A. Exler, V. Diener, A. Schankin, M. Beigl. “Detection of a Smartphone User’s Distraction Based on Typing and Touch Gestures.” In: *Proceedings of the 13th EAI International Conference on Pervasive Computing Technologies for Healthcare*. PervasiveHealth’19. Trento, Italy: ACM, 2019, pp. 242–250. ISBN: 978-1-4503-6126-2. DOI: 10.1145/3329189.3329212 (cit. on p. 45).

[39] A. Exler, C. Dinse, Z. Günes, N. Hammoud, S. Mattes, M. Beigl. “Investigating the Perceptibility Different Notification Types on Smartphones Depending on the Smartphone Position.” In: *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. UbiComp ’17. Maui, Hawaii: ACM, 2017, pp. 970–976. ISBN: 978-1-4503-5190-4. DOI: 10.1145/3123024.3124560 (cit. on p. 38).

[40] A. Exler, A. Voit, D. Weber, M. Pielot, N. Goyal, S. Gehring, T. Okoshi, V. Pejovic. “UbiTtention 2019: 4th International Workshop on Smart & Ambient Notification and Attention Management.” In: *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. UbiComp/ISWC ’19. London, United Kingdom: ACM, 2019, pp. 1054–1057. ISBN: 978-4503-6869-8. DOI: 10.1145/3341162.3347766 (cit. on p. 25).

[41] D. Fallman, B. Yttergren. “Meeting in Quiet: Choosing Suitable Notification Modalities for Mobile Phones.” In: *Proceedings of the 2005 Conference on Designing for User eXperience*. DUX ’05. San Francisco, California, USA: AIGA: American Institute of Graphic Arts, 2005. ISBN: 1-59593-250-X (cit. on p. 41).

[42] A. P. Felt, S. Egelman, D. Wagner. “I’ve Got 99 Problems, but Vibration Ain’t One: A Survey of Smartphone Users’ Concerns.” In: *Proceedings of the Second ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*. SPSM ’12. Raleigh, North Carolina, USA: ACM, 2012, pp. 33–44. ISBN: 978-1-4503-1666-8. DOI: 10.1145/2381934.2381943 (cit. on p. 75).

[43] J. E. Fischer, C. Greenhalgh, S. Benford. “Investigating Episodes of Mobile Phone Activity As Indicators of Opportune Moments to Deliver Notifications.” In: *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI ’11. Stockholm, Sweden: ACM, 2011, pp. 181–190. ISBN: 978-1-4503-0541-9. DOI: 10.1145/2037373.2037402 (cit. on pp. 44, 70, 108, 111).

[44] J. E. Fischer, N. Yee, V. Bellotti, N. Good, S. Benford, C. Greenhalgh. “Effects of Content and Time of Delivery on Receptivity to Mobile Interruptions.” In: *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI ’10. Lisbon, Portugal: ACM, 2010, pp. 103–112. ISBN: 978-1-60558-835-3. DOI: 10.1145/1851600.1851620 (cit. on pp. 37, 44, 70).

[45] J. A. Gallud, R. Tesoriero. “Smartphone Notifications: A Study on the Sound to Soundless Tendency.” In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. Mobile-HCI ’15. Copenhagen, Denmark: ACM, 2015, pp. 819–824. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2793706 (cit. on p. 43).

[46] D. Geerts, P. Cesar, D. Bulterman. “The Implications of Program Genres for the Design of Social Television Systems.” In: *Proceedings of the 1st International Conference on Designing Interactive User Experiences for TV and Video*. UXTV ’08. Silicon Valley, California, USA: ACM, 2008, pp. 71–80. ISBN: 978-1-60558-100-2. DOI: 10.1145/1453805.1453822 (cit. on p. 165).

[47] T. Gillie, D. Broadbent. “What makes interruptions disruptive? A study of length, similarity, and complexity.” In: *Psychological Research* 50.4 (1989), pp. 243–250. ISSN: 1430-2772. DOI: 10.1007/BF00309260 (cit. on pp. 16, 42).

[48] E. Goodman, M. Kuniavsky, A. Moed. *Observing the User Experience, Second Edition: A Practitioner’s Guide to User Research*. 2nd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2012. ISBN: 0123848695, 9780123848697 (cit. on p. 165).

[49] S. J. Gould, D. P. Brumby, A. L. Cox. “What does it mean for an interruption to be relevant? An investigation of relevance as a memory effect.” In: *Proceedings of the human factors and ergonomics society annual meeting*. Vol. 57. 1. SAGE Publications Sage CA: Los Angeles, CA. 2013, pp. 149–153 (cit. on pp. 37, 42).

[50] R. Gouveia, E. Karapanos. “Footprint Tracker: Supporting Diary Studies with Lifelogging.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 2921–2930. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2481405 (cit. on p. 73).

[51] N. Goyal, S. R. Fussell. “Intelligent Interruption Management Using Electro Dermal Activity Based Physiological Sensor for Collaborative Sensemaking.” In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1.3 (Sept. 2017), 52:1–52:21. ISSN: 2474-9567. DOI: 10.1145/3130917 (cit. on p. 45).

[52] C. Grevet, D. Choi, D. Kumar, E. Gilbert. "Overload is Overloaded: Email in the Age of Gmail." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 793–802. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557013 (cit. on p. 69).

[53] S. Günther, F. Müller, M. Funk, M. Mühlhäuser. "Slappyfications: Towards Ubiquitous Physical and Embodied Notifications." In: *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI EA '19. Glasgow, Scotland Uk: ACM, 2019, VS10:1–VS10:2. ISBN: 978-1-4503-5971-9. DOI: 10.1145/3290607.3311780 (cit. on p. 41).

[54] R. Hansson, P. Ljungstrand, J. Redström. "Subtle and public notification cues for mobile devices." In: *International Conference on Ubiquitous Computing*. Springer. 2001, pp. 240–246 (cit. on p. 37).

[55] J. M. Hektner, J. A. Schmidt, M. Csikszentmihalyi. *Experience sampling method: Measuring the quality of everyday life*. Sage, 2007 (cit. on pp. 73, 140).

[56] N. Henze, M. Pielot. "App stores: external validity for mobile HCI." In: *interactions* 20.2 (2013), pp. 33–38 (cit. on p. 49).

[57] N. Henze, M. Pielot, B. Poppinga, T. Schinke, S. Boll. "My app is an experiment: Experience from user studies in mobile app stores." In: *International Journal of Mobile Human Computer Interaction (IJMHCI)* 3.4 (2011), pp. 71–91 (cit. on p. 49).

[58] N. Henze, A. Schmidt, M. Pielot, F. Michahelles. "Empirical research through ubiquitous data collection." In: *Computer* (2013), pp. 74–76 (cit. on p. 49).

[59] J. Hess, B. Ley, C. Ogonowski, L. Wan, V. Wulf. "Jumping Between Devices and Services: Towards an Integrated Concept for Social Tv." In: *Proceedings of the 9th European Conference on Interactive TV and Video*. EuroITV '11. Lisbon, Portugal: ACM, 2011, pp. 11–20. ISBN: 978-1-4503-0602-7. DOI: 10.1145/2000119.2000122 (cit. on p. 164).

[60] J. Ho, S. S. Intille. "Using Context-aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. Portland, Oregon, USA: ACM, 2005, pp. 909–918. ISBN: 1-58113-998-5. DOI: 10.1145/1054972.1055100 (cit. on pp. 45, 110, 111).

[61] C. Holz, F. Bentley, K. Church, M. Patel. “"I’m Just on My Phone and They’re Watching TV": Quantifying Mobile Device Use While Watching Television.” In: *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video*. TVX ’15. Brussels, Belgium: ACM, 2015, pp. 93–102. ISBN: 978-1-4503-3526-3. DOI: 10.1145/2745197.2745210 (cit. on p. 164).

[62] M. Hoppe, R. Boldt, J. Strauß, L. Lischke, D. Weber, N. Henze. “Image Browsing on Large High-resolution Displays.” In: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM ’17. Stuttgart, Germany: ACM, 2017, pp. 401–405. ISBN: 978-1-4503-5378-6. DOI: 10.1145/3152832.3156619 (cit. on p. 24).

[63] E. C. M. C. E. Horvitz. “Notification, disruption, and memory: Effects of messaging interruptions on memory and performance.” In: *Proceedings of INTERACT*. 2001, p. 263 (cit. on pp. 42, 51).

[64] E. Horvitz, P. Koch, J. Apacible. “BusyBody: Creating and Fielding Personalized Models of the Cost of Interruption.” In: *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*. CSCW ’04. Chicago, Illinois, USA: ACM, 2004, pp. 507–510. ISBN: 1-58113-810-5. DOI: 10.1145/1031607.1031690 (cit. on p. 110).

[65] S. T. Iqbal, B. P. Bailey. “Effects of Intelligent Notification Management on Users and Their Tasks.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 93–102. ISBN: 978-1-60558-011-1. DOI: 10.1145/1357054.1357070 (cit. on p. 45).

[66] S. T. Iqbal, B. P. Bailey. “Oasis: A Framework for Linking Notification Delivery to the Perceptual Structure of Goal-directed Tasks.” In: *ACM Trans. Comput.-Hum. Interact.* 17.4 (Dec. 2010), 15:1–15:28. ISSN: 1073-0516. DOI: 10.1145/1879831.1879833 (cit. on pp. 16, 36, 37, 53).

[67] S. T. Iqbal, E. Horvitz. “Notifications and Awareness: A Field Study of Alert Usage and Preferences.” In: *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*. CSCW ’10. Savannah, Georgia, USA: ACM, 2010, pp. 27–30. ISBN: 978-1-60558-795-0. DOI: 10.1145/1718918.1718926 (cit. on pp. 16, 39).

[68] B. R. Jones, H. Benko, E. Ofek, A. D. Wilson. “IllumiRoom: Peripheral Projected Illusions for Interactive Experiences.” In: *Proceedings of the SIGCHI Conference*

on Human Factors in Computing Systems. CHI '13. Paris, France: ACM, 2013, pp. 869–878. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2466112 (cit. on p. 199).

[69] D. Kahneman, A. B. Krueger, D. A. Schkade, N. Schwarz, A. A. Stone. "A survey method for characterizing daily life experience: The day reconstruction method." In: *Science* 306.5702 (2004), pp. 1776–1780 (cit. on p. 74).

[70] Y. M. Kalman, G. Ravid, D. R. Raban, S. Rafaeli. "Pauses and response latencies: A chronemic analysis of asynchronous CMC." In: *Journal of Computer-Mediated Communication* 12.1 (2006), pp. 1–23 (cit. on p. 36).

[71] S Kalyanaraman, J Ivory, L Maschmeyer. "Interruptions and online information processing: The role of interruption type, interruption content, and interruption frequency." In: *Proc. of 2005 Annual Meeting of International Communication Association*. Vol. 2005. 2005, pp. 1–32 (cit. on p. 37).

[72] K. Katevas, I. Arapakis, M. Pielot. "Typical Phone Use Habits: Intense Use Does Not Predict Negative Well-being." In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '18. Barcelona, Spain: ACM, 2018, 11:1–11:13. ISBN: 978-1-4503-5898-9. DOI: 10.1145/3229434.3229441 (cit. on p. 42).

[73] N. Kern, B. Schiele. "Context-aware notification for wearable computing." In: *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.* IEEE. 2003, pp. 223–230 (cit. on p. 110).

[74] R. Kettner, P. Bader, T. Kosch, S. Schneegass, A. Schmidt. "Towards Pressure-Based Feedback for Non-Stressful Tactile Notifications." In: *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '17. Vienna, Austria: Association for Computing Machinery, 2017. ISBN: 9781450350754. DOI: 10.1145/3098279.3122132 (cit. on p. 41).

[75] J. Kjeldskov, M. B. Skov. "Was It Worth the Hassle?: Ten Years of Mobile HCI Research Discussions on Lab and Field Evaluations." In: *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services*. MobileHCI '14. Toronto, ON, Canada: ACM, 2014, pp. 43–52. ISBN: 978-1-4503-3004-6. DOI: 10.1145/2628363.2628398 (cit. on p. 48).

[76] J. Kjeldskov, M. B. Skov, B. S. Als, R. T. Høegh. “Is It Worth the Hassle? Exploring the Added Value of Evaluating the Usability of Context-Aware Mobile Systems in the Field.” In: *Mobile Human-Computer Interaction - MobileHCI 2004*. Ed. by S. Brewster, M. Dunlop. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 61–73. ISBN: 978-3-540-28637-0 (cit. on p. 47).

[77] T. Kubitza, A. Voit, D. Weber, A. Schmidt. “An IoT Infrastructure for Ubiquitous Notifications in Intelligent Living Environments.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1536–1541. ISBN: 978-1-4503-4462-3. DOI: `10.1145/2968219.2968545` (cit. on pp. 24, 25, 40, 206).

[78] K. Kushlev, J. Proulx, E. W. Dunn. “"Silence Your Phones": Smartphone Notifications Increase Inattention and Hyperactivity Symptoms.” In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI ’16. San Jose, California, USA: ACM, 2016, pp. 1011–1020. ISBN: 978-1-4503-3362-7. DOI: `10.1145/2858036.2858359` (cit. on pp. 16, 39, 42).

[79] M. Langheinrich. “Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems.” In: *International conference on Ubiquitous Computing*. Springer. 2001, pp. 273–291 (cit. on p. 188).

[80] J. Lee, J. Kwon, H. Kim. “Reducing Distraction of Smartwatch Users with Deep Learning.” In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI ’16. Florence, Italy: ACM, 2016, pp. 948–953. ISBN: 978-1-4503-4413-5. DOI: `10.1145/2957265.2962662` (cit. on p. 40).

[81] S.-H. Lee, M.-K. Sohn, D.-J. Kim, B. Kim, H. Kim. “Smart tv interaction system using face and hand gesture recognition.” In: *2013 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE. 2013, pp. 173–174 (cit. on p. 164).

[82] U. Lee, J. Lee, M. Ko, C. Lee, Y. Kim, S. Yang, K. Yatani, G. Gweon, K.-M. Chung, J. Song. “Hooked on Smartphones: An Exploratory Study on Smartphone Overuse Among College Students.” In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 2327–2336. ISBN: 978-1-4503-2473-1. DOI: `10.1145/2556288.2557366` (cit. on p. 43).

[83] L. Leiva, M. Böhmer, S. Gehring, A. Krüger. "Back to the App: The Costs of Mobile Application Interruptions." In: *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*. MobileHCI '12. San Francisco, California, USA: ACM, 2012, pp. 291–294. ISBN: 978-1-4503-1105-2. DOI: 10.1145/2371574.2371617 (cit. on p. 42).

[84] L. Lischke, D. Weber, S. W. Greenwald. "Pervasive Interaction Across Displays." In: *IEEE Pervasive Computing* 14.4 (2015), pp. 12–15. ISSN: 1536-1268. DOI: 10.1109/MPRV.2015.77 (cit. on p. 24).

[85] M. Lochrie, P. Coulton. "Mobile Phones As Second Screen for TV, Enabling Inter-audience Interaction." In: *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology*. ACE '11. Lisbon, Portugal: ACM, 2011, 73:1–73:2. ISBN: 978-1-4503-0827-4. DOI: 10.1145/2071423.2071513 (cit. on p. 163).

[86] K. Lyons. "What Can a Dumb Watch Teach a Smartwatch?: Informing the Design of Smartwatches." In: *Proceedings of the 2015 ACM International Symposium on Wearable Computers*. ISWC '15. Osaka, Japan: ACM, 2015, pp. 3–10. ISBN: 978-1-4503-3578-2. DOI: 10.1145/2802083.2802084 (cit. on p. 162).

[87] M. Mann. *Inbox zero*. http://sagolla.net/media/jpg/merlinmann_adobetalk_2007-03.pdf. 2007 (cit. on p. 69).

[88] A. Mashhadi, A. Mathur, F. Kawsar. "The Myth of Subtle Notifications." In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. UbiComp '14 Adjunct. Seattle, Washington: ACM, 2014, pp. 111–114. ISBN: 978-1-4503-3047-3. DOI: 10.1145/2638728.2638759 (cit. on p. 37).

[89] S. Mayer, L. Lischke, P. W. Woźniak, N. Henze. "Evaluating the Disruptiveness of Mobile Interactions: A Mixed-Method Approach." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 406:1–406:14. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173980 (cit. on p. 43).

[90] S. Mayer, V. Schwind, H. V. Le, D. Weber, J. Vogelsang, J. Wolf, N. Henze. "Effect of Orientation on Unistroke Touch Gestures." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: ACM, 2019, 698:1–698:9. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300928 (cit. on p. 24).

[91] A. Mehrotra. “A framework for intelligent mobile notifications.” PhD thesis. University of Birmingham, 2017 (cit. on p. 46).

[92] A. Mehrotra, R. Hendley, M. Musolesi. “NotifyMeHere: Intelligent Notification Delivery in Multi-Device Environments.” In: *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval*. CHIIR ’19. Glasgow, Scotland UK: Association for Computing Machinery, 2019, pp. 103–111. ISBN: 9781450360258. DOI: 10.1145/3295750.3298932 (cit. on p. 41).

[93] A. Mehrotra, R. Hendley, M. Musolesi. “PrefMiner: Mining User’s Preferences for Intelligent Mobile Notification Management.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1223–1234. ISBN: 978-1-4503-4461-6. DOI: 10.1145/2971648.2971747 (cit. on pp. 43, 45, 46, 71, 110).

[94] A. Mehrotra, M. Musolesi, R. Hendley, V. Pejovic. “Designing Content-driven Intelligent Notification Mechanisms for Mobile Applications.” In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp ’15. Osaka, Japan: ACM, 2015, pp. 813–824. ISBN: 978-1-4503-3574-4. DOI: 10.1145/2750858.2807544 (cit. on pp. 38, 42, 46, 73, 110).

[95] A. Mehrotra, V. Pejovic, J. Vermeulen, R. Hendley, M. Musolesi. “My Phone and Me: Understanding People’s Receptivity to Mobile Notifications.” In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI ’16. San Jose, California, USA: ACM, 2016, pp. 1021–1032. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858566 (cit. on pp. 37, 38, 42, 46, 73, 109).

[96] A. Mehrotra, J. Vermeulen, V. Pejovic, M. Musolesi. “Ask, but Don’t Interrupt: The Case for Interruptibility-Aware Mobile Experience Sampling.” In: *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*. UbiComp/ISWC’15 Adjunct. Osaka, Japan: Association for Computing Machinery, 2015, pp. 723–732. ISBN: 9781450335751. DOI: 10.1145/2800835.2804397 (cit. on p. 46).

[97] A. Morrison, D. McMillan, S. Reeves, S. Sherwood, M. Chalmers. “A Hybrid Mass Participation Approach to Mobile Software Trials.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, Texas, USA: ACM, 2012, pp. 1311–1320. ISBN: 978-1-4503-1015-4. DOI: 10.1145/2207676.2208588 (cit. on p. 124).

[98] B. A. Nardi, S. Whittaker, E. Bradner. “Interaction and Outeraction: Instant Messaging in Action.” In: *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*. CSCW ’00. Philadelphia, Pennsylvania, USA: ACM, 2000, pp. 79–88. ISBN: 1-58113-222-0. DOI: 10.1145/358916.358975 (cit. on p. 36).

[99] M. Nathan, C. Harrison, S. Yarosh, L. Terveen, L. Stead, B. Amento. “CollaboraTV: Making Television Viewing Social Again.” In: *Proceedings of the 1st International Conference on Designing Interactive User Experiences for TV and Video*. UXTV ’08. Silicon Valley, California, USA: ACM, 2008, pp. 85–94. ISBN: 978-1-60558-100-2. DOI: 10.1145/1453805.1453824 (cit. on p. 163).

[100] T. Neate, M. Jones, M. Evans. “Mediating Attention for Second Screen Companion Content.” In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI ’15. Seoul, Republic of Korea: ACM, 2015, pp. 3103–3106. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702278 (cit. on p. 165).

[101] D. Newtson, G. Engquist. “The perceptual organization of ongoing behavior.” In: *Journal of Experimental Social Psychology* 12.5 (1976), pp. 436–450 (cit. on p. 111).

[102] C. M. Nielsen, M. Overgaard, M. B. Pedersen, J. Stage, S. Stenild. “It’s Worth the Hassle!: The Added Value of Evaluating the Usability of Mobile Systems in the Field.” In: *Proceedings of the 4th Nordic Conference on Human-computer Interaction: Changing Roles*. NordiCHI ’06. Oslo, Norway: ACM, 2006, pp. 272–280. ISBN: 1-59593-325-5. DOI: 10.1145/1182475.1182504 (cit. on p. 47).

[103] L. Norrie, R. Murray-Smith. “Impact of Smartphone Notification Display Choice in a Typing Task.” In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI ’15. Copenhagen, Denmark: Association for Computing Machinery, 2015, pp. 1094–1099. ISBN: 9781450336536. DOI: 10.1145/2786567.2794335 (cit. on p. 216).

[104] M. Obuchi, W. Sasaki, T. Okoshi, J. Nakazawa, H. Tokuda. “Investigating Interruptibility at Activity Breakpoints Using Smartphone Activity Recognition API.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1602–1607. ISBN: 978-1-4503-4462-3. DOI: 10.1145/2968219.2968556 (cit. on p. 46).

[105] T. Okoshi, A. Exler, A. Voit, D. Weber, M. Pielot, B. Poppinga, N. Henze, S. Gehring, M. Böhmer, S. Kim, V. Pejovic. “UbiTtention 2017: 2nd International Workshop on Smart & Ambient Notification and Attention Management.” In: *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. UbiComp ’17. Maui, Hawaii: ACM, 2017, pp. 893–897. ISBN: 978-1-4503-5190-4. DOI: 10.1145/3123024.3124458 (cit. on p. 25).

[106] T. Okoshi, J. Nakazawa, H. Tokuda. “Attelia: Sensing User’s Attention Status on Smart Phones.” In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. UbiComp ’14 Adjunct. Seattle, Washington: ACM, 2014, pp. 139–142. ISBN: 978-1-4503-3047-3. DOI: 10.1145/2638728.2638802 (cit. on pp. 44, 108, 111).

[107] T. Okoshi, J. Nakazawa, H. Tokuda. “Interruptibility Research: Opportunities for Future Flourishment.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1524–1529. ISBN: 978-1-4503-4462-3. DOI: 10.1145/2968219.2968543 (cit. on p. 46).

[108] T. Okoshi, H. Nozaki, J. Nakazawa, H. Tokuda, J. Ramos, A. K. Dey. “Towards attention-aware adaptive notification on smart phones.” In: *Pervasive and Mobile Computing* 26 (2016), pp. 17–34 (cit. on p. 44).

[109] T. Okoshi, J. Ramos, H. Nozaki, J. Nakazawa, A. K. Dey, H. Tokuda. “Attelia: Reducing User’s Cognitive Load due to Interruptive Notifications on Smart Phones.” In: *PerCom’15*. IEEE, 2015, pp. 96–104 (cit. on pp. 44, 70, 71, 108, 111).

[110] T. Okoshi, J. Ramos, H. Nozaki, J. Nakazawa, A. K. Dey, H. Tokuda. “Reducing Users’ Perceived Mental Effort Due to Interruptive Notifications in Multi-device Mobile Environments.” In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp ’15. Osaka, Japan: ACM, 2015, pp. 475–486. ISBN: 978-1-4503-3574-4. DOI: 10.1145/2750858.2807517 (cit. on pp. 41, 44, 108, 111).

[111] T. Okoshi, K. Tsubouchi, M. Taji, T. Ichikawa, H. Tokuda. “Attention and engagement-awareness in the wild: A large-scale study with adaptive notifications.” In: *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE. 2017, pp. 100–110 (cit. on pp. 44, 111).

[112] T. Okoshi, K. Tsubouchi, H. Tokuda. "Real-World Product Deployment of Adaptive Push Notification Scheduling on Smartphones." In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. Anchorage, AK, USA: ACM, 2019, pp. 2792–2800. ISBN: 978-1-4503-6201-6. DOI: 10.1145/3292500.3330732 (cit. on p. 44).

[113] T. Okoshi, K. Tsubouchi, H. Tokuda. "Real-world large-scale study on adaptive notification scheduling on smartphones." In: *Pervasive and Mobile Computing* 50 (2018), pp. 1–24 (cit. on p. 44).

[114] C. Park, J. Lim, J. Kim, S.-J. Lee, D. Lee. "Don't Bother Me. I'm Socializing!: A Breakpoint-Based Smartphone Notification System." In: *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17. Portland, Oregon, USA: ACM, 2017, pp. 541–554. ISBN: 978-1-4503-4335-0. DOI: 10.1145/2998181.2998189 (cit. on pp. 37, 44, 108, 111).

[115] J. Pearson, S. Robinson, M. Jones. "It's About Time: Smartwatches As Public Displays." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 1257–1266. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702247 (cit. on p. 40).

[116] V. Pejovic, M. Musolesi. "InterruptMe: Designing Intelligent Prompting Mechanisms for Pervasive Applications." In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '14. Seattle, Washington: ACM, 2014, pp. 897–908. ISBN: 978-1-4503-2968-2. DOI: 10.1145/2632048.2632062 (cit. on pp. 44, 46, 111).

[117] V. Pejovic, M. Musolesi, A. Mehrotra. "Investigating The Role of Task Engagement in Mobile Interruptibility." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 1100–1105. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2794336 (cit. on p. 42).

[118] M. Pielot. "Large-scale Evaluation of Call-availability Prediction." In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '14. Seattle, Washington: ACM, 2014, pp. 933–937. ISBN: 978-1-4503-2968-2. DOI: 10.1145/2632048.2632060 (cit. on p. 43).

[119] M. Pielot, L. Baltrunas, N. Oliver. "Boredom-Triggered Proactive Recommendations." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 1106–1110. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2794340 (cit. on p. 45).

[120] M. Pielot, B. Cardoso, K. Katevas, J. Serrà, A. Matic, N. Oliver. "Beyond Interruptibility: Predicting Opportune Moments to Engage Mobile Phone Users." In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1.3 (Sept. 2017), 91:1–91:25. ISSN: 2474-9567. DOI: 10.1145/3130956 (cit. on pp. 45, 111).

[121] M. Pielot, K. Church, R. de Oliveira. "An In-situ Study of Mobile Phone Notifications." In: *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services*. MobileHCI '14. Toronto, ON, Canada: ACM, 2014, pp. 233–242. ISBN: 978-1-4503-3004-6. DOI: 10.1145/2628363.2628364 (cit. on pp. 36–38, 51, 60, 71, 108).

[122] M. Pielot, R. de Oliveira, H. Kwak, N. Oliver. "Didn't You See My Message?: Predicting Attentiveness to Mobile Instant Messages." In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 3319–3328. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2556973 (cit. on pp. 37, 45, 71).

[123] M. Pielot, T. Dingler, J. S. Pedro, N. Oliver. "When Attention is Not Scarce - Detecting Boredom from Mobile Phone Usage." In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '15. Osaka, Japan: ACM, 2015, pp. 825–836. ISBN: 978-1-4503-3574-4. DOI: 10.1145/2750858.2804252 (cit. on pp. 37, 45, 110).

[124] M. Pielot, N. Henze, S. Boll. "Experiments in app stores-how to ask users for their consent." In: *Proceedings of the workshop on Ethics, logs & videotape*. 2011 (cit. on p. 55).

[125] M. Pielot, L. Rello. "Productive, Anxious, Lonely: 24 Hours Without Push Notifications." In: *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '17. Vienna, Austria: ACM, 2017, 11:1–11:11. ISBN: 978-1-4503-5075-4. DOI: 10.1145/3098279.3098526 (cit. on pp. 43, 109).

[126] M. Pielot, L. Rello. "The Do Not Disturb Challenge: A Day Without Notifications." In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '15. Seoul, Republic of Korea: ACM, 2015, pp. 1761–1766. ISBN: 978-1-4503-3146-3. DOI: 10.1145/2702613.2732704 (cit. on pp. 37, 43, 108, 109).

[127] M. Pielot, A. Vradi, S. Park. "Dismissed!: A Detailed Exploration of How Mobile Phone Users Handle Push Notifications." In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '18. Barcelona, Spain: ACM, 2018, 3:1–3:11. ISBN: 978-1-4503-5898-9. DOI: 10.1145/3229434.3229445 (cit. on pp. 36, 37, 65, 67, 70–72, 87).

[128] S. Pizza, B. Brown, D. McMillan, A. Lampinen. "Smartwatch in Vivo." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: ACM, 2016, pp. 5456–5469. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858522 (cit. on p. 39).

[129] R. Poguntke, C. Schneegass, L. Van der Vekens, R. Rzayev, J. Auda, S. Schneegass, A. Schmidt. "NotiModes: An Investigation of Notification Delay Modes and Their Effects on Smartphone Users." In: New York, NY, USA: Association for Computing Machinery, 2020, pp. 415–419. ISBN: 9781450375405 (cit. on p. 45).

[130] B. Poppinga, W. Heuten, S. Boll. "Sensor-Based Identification of Opportune Moments for Triggering Notifications." In: *IEEE Pervasive Computing* 13.1 (2014), pp. 22–29. ISSN: 1536-1268 (cit. on p. 44).

[131] B. Poppinga, M. Pielot, N. Henze, N. Oliver, K. Church, A. S. Shirazi. "Smarttention, Please! Intelligent Attention Management on Mobile Devices." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 1066–1069. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2795399 (cit. on pp. 21, 25).

[132] T. Regan, I. Todd. "Media Center Buddies: Instant Messaging Around a Media Center." In: *Proceedings of the Third Nordic Conference on Human-computer Interaction*. NordiCHI '04. Tampere, Finland: ACM, 2004, pp. 141–144. ISBN: 1-58113-857-1. DOI: 10.1145/1028014.1028036 (cit. on p. 164).

[133] Y. Rogers, K. Connelly, L. Tedesco, W. Hazlewood, A. Kurtz, R. E. Hall, J. Hursey, T. Toscos. "Why It's Worth the Hassle: The Value of In-Situ Studies When Designing Ubicomp." In: *UbiComp 2007: Ubiquitous Computing*. Ed. by J. Krumm, G. D. Abowd, A. Seneviratne, T. Strang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 336–353. ISBN: 978-3-540-74853-3 (cit. on pp. 47, 48).

[134] R. Rzayev, S. Korbely, M. Maul, A. Schark, V. Schwind, N. Henze. "Effects of Position and Alignment of Notifications on AR Glasses during Social Interaction." In: *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*. NordiCHI '20. Tallinn, Estonia: Association for Computing Machinery, 2020. ISBN: 9781450375795. DOI: 10.1145/3419249.3420095 (cit. on pp. 208, 216).

[135] R. Rzayev, S. Mayer, C. Krauter, N. Henze. "Notification in VR: The Effect of Notification Placement, Task and Environment." In: *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. CHI PLAY '19. Barcelona, Spain: ACM, 2019, pp. 199–211. ISBN: 978-1-4503-6688-5. DOI: 10.1145/3311350.3347190 (cit. on pp. 208, 216).

[136] A. Sahami Shirazi, N. Henze. "Assessment of Notifications on Smartwatches." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 1111–1116. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2794338 (cit. on pp. 39, 40).

[137] A. Sahami Shirazi, N. Henze, T. Dingler, M. Pielot, D. Weber, A. Schmidt. "Large-scale Assessment of Mobile Notifications." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 3055–3064. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557189 (cit. on pp. 16, 24, 32, 33, 37, 42, 51, 62, 64, 67, 71–73, 80, 108, 119, 133, 162, 186).

[138] A. Sasse, C. Johnson, et al. "Coordinating the interruption of people in human-computer interaction." In: *Human-computer interaction, INTERACT*. Vol. 99. 1999, p. 295 (cit. on p. 42).

[139] A. Schmidt, N. Henze, D. Weber, A. Voit. *DAAN: Schlussbericht des Teilvorhabens der Uni Stuttgart: Projekt: Design Adaptiver Ambienter Notifikationsumgebungen, Vorhabensbezeichnung: DAAN: Laufzeit des Vorhabens: 01.03. 2015-28.02. 2018, Berichtszeitraum: 01.03. 2015-28.02. 2018*. Universität Stuttgart, Institut für Visualisierung und Interaktive Systeme (VIS), 2018 (cit. on p. 24).

[140] S. Schneegass, R. Rzayev. “Embodied Notifications: Implicit Notifications Through Electrical Muscle Stimulation.” In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI ’16. Florence, Italy: ACM, 2016, pp. 954–959. ISBN: 978-1-4503-4413-5. DOI: 10.1145/2957265.2962663 (cit. on p. 40).

[141] C. Shin, W. Woo. “Socially aware tv program recommender for multiple viewers.” In: *IEEE Transactions on Consumer Electronics* 55.2 (2009), pp. 927–932 (cit. on p. 164).

[142] G. B. D. Shoemaker, K. M. Inkpen. “Single Display Privacyware: Augmenting Public Displays with Private Information.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’01. Seattle, Washington, USA: ACM, 2001, pp. 522–529. ISBN: 1-58113-327-8. DOI: 10.1145/365024.365349 (cit. on p. 188).

[143] D. P. Siewiorek, A. Smailagic, J. Furukawa, A. Krause, N. Moraveji, K. Reiger, J. Shaffer, F. L. Wong. “SenSay: A Context-Aware Mobile Phone.” In: *ISWC*. Vol. 3. 2003, p. 248. DOI: 10.1109/ISWC.2003.1241422 (cit. on pp. 44, 108, 110).

[144] L. A. Silva, V. R. Q. Leithardt, C. O. Rolim, G. V. González, C. F. Geyer, J. S. Silva. “PRISER: managing notification in multiples devices with data privacy support.” In: *Sensors* 19.14 (2019), p. 3098 (cit. on p. 208).

[145] L. Silva, G. d. Mello, B. A. d. Silva, G. Villarrubia Gonzalez, J. d. P. Santana, P. Prata, V. Leithardt. “Multi-device Notifications: A Comparison Between MQTT and CoAP.” In: *The Thirteenth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*. ThinkMind, pp. 9–13 (cit. on p. 208).

[146] K. Stawarz, A. L. Cox, A. Blandford. “Don’t Forget Your Pill!: Designing Effective Medication Reminder Apps That Support Users’ Daily Routines.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 2269–2278. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2557079 (cit. on p. 109).

[147] C. Stothart, A. Mitchum, C. Yehnert. “The attentional cost of receiving a cell phone notification.” In: *Journal of experimental psychology: human perception and performance* 41.4 (2015), p. 893 (cit. on p. 37).

[148] J. W. Streefkerk, D. S. McCrickard, M. P. van Esch-Bussemakers, M. A. Neerincx. “Balancing awareness and interruption in mobile patrol using context-aware notification.” In: *International Journal of Mobile Human Computer Interaction (IJMHCI)* 4.3 (2012), pp. 1–27 (cit. on p. 44).

[149] K. Tsubouchi, T. Okoshi. “People’s Interruptibility In-the-wild: Analysis of Breakpoint Detection Model in a Large-scale Study.” In: *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. UbiComp ’17. Maui, Hawaii: ACM, 2017, pp. 922–927. ISBN: 978-1-4503-5190-4. DOI: 10.1145/3123024.3124556 (cit. on p. 44).

[150] L. D. Turner, S. M. Allen, R. M. Whitaker. “Reachable but not receptive: Enhancing smartphone interruptibility prediction by modelling the extent of user engagement with notifications.” In: *Pervasive and Mobile Computing* 40 (2017), pp. 480–494 (cit. on p. 44).

[151] N. van Berkel, D. Ferreira, V. Kostakos. “The Experience Sampling Method on Mobile Devices.” In: *ACM Comput. Surv.* 50.6 (Dec. 2017), 93:1–93:40. ISSN: 0360-0300. DOI: 10.1145/3123988 (cit. on p. 73).

[152] R. Vardhan, A. Sanzgiri, D. Kulkarni, P. Joshi, S. Nalluri. “Notify Assist: Balancing Privacy and Convenience in Delivery of Notifications on Android Smartphones.” In: *Proceedings of the 2017 on Workshop on Privacy in the Electronic Society*. WPES ’17. Dallas, Texas, USA: ACM, 2017, pp. 17–20. ISBN: 978-1-4503-5175-1. DOI: 10.1145/3139550.3139561 (cit. on p. 43).

[153] A. Visuri, Z. Sarsenbayeva, N. van Berkel, J. Goncalves, R. Rawassizadeh, V. Kostakos, D. Ferreira. “Quantifying Sources and Types of Smartwatch Usage Sessions.” In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI ’17. Denver, Colorado, USA: ACM, 2017, pp. 3569–3581. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025817 (cit. on p. 39).

[154] A. Visuri, N. van Berkel. “Attention Computing: Overview of Mobile Sensing Applied to Measuring Attention.” In: *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. UbiComp/ISWC ’19 Adjunct. London, United Kingdom: Association for Computing Machinery, 2019, pp. 1079–1082. ISBN: 9781450368698. DOI: 10.1145/3341162.3344843 (cit. on p. 45).

[155] A. Visuri, N. van Berkel, C. Luo, J. Goncalves, D. Ferreira, V. Kostakos. "Predicting Interruptibility for Manual Data Collection: A Cluster-Based User Model." In: *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '17. Vienna, Austria: Association for Computing Machinery, 2017. ISBN: 9781450350754. DOI: `10.1145/3098279.3098532` (cit. on p. 46).

[156] A. Visuri, N. van Berkel, T. Okoshi, J. Goncalves, V. Kostakos. "Understanding smartphone notifications' user interactions and content importance." In: *International Journal of Human-Computer Studies* 128 (2019), pp. 72–85. ISSN: 1071-5819. DOI: `https://doi.org/10.1016/j.ijhcs.2019.03.001` (cit. on pp. 38, 69).

[157] D. Vogel, R. Balakrishnan. "Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users." In: *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. UIST '04. Santa Fe, NM, USA: ACM, 2004, pp. 137–146. ISBN: 1-58113-957-8. DOI: `10.1145/1029632.1029656` (cit. on pp. 188, 192).

[158] A. Voit, M. Greis, D. Weber, K. Gönner, I. Kutger, T. Mantz, S. Schmidt, L. van der Vekens, A. Schmidt. "My Painting Shows My Stats: Towards a Personal Colorful Activity Display." In: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM '17. Stuttgart, Germany: ACM, 2017, pp. 415–420. ISBN: 978-1-4503-5378-6. DOI: `10.1145/3152832.3156621` (cit. on p. 24).

[159] A. Voit, T. Machulla, D. Weber, V. Schwind, S. Schneegass, N. Henze. "Exploring Notifications in Smart Home Environments." In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '16. Florence, Italy: ACM, 2016, pp. 942–947. ISBN: 978-1-4503-4413-5. DOI: `10.1145/2957265.2962661` (cit. on pp. 24, 25).

[160] A. Voit, J. Niess, C. Eckerth, M. Ernst, H. Weingärtner, P. W. Woźniak. "It's Not a Romantic Relationship: Stories of Adoption and Abandonment of Smart Speakers at Home." In: *19th International Conference on Mobile and Ubiquitous Multimedia*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 71–82. ISBN: 9781450388702 (cit. on p. 215).

[161] A. Voit, B. Poppinga, D. Weber, M. Böhmer, N. Henze, S. Gehring, T. Okoshi, V. Pejovic. "UbiTtention: Smart & Ambient Notification and Attention Management." In: *Proceedings of the 2016 ACM International Joint Conference on*

Pervasive and Ubiquitous Computing: Adjunct. UbiComp '16. Heidelberg, Germany: ACM, 2016, pp. 1520–1523. ISBN: 978-1-4503-4462-3. DOI: 10.1145/2968219.2968542 (cit. on p. 25).

[162] A. Voit, R. Rzayev, D. Weber, M. Müller, N. Henze. "Investigation of an Ambient and Pervasive Smart Wall Calendar with Event Suggestions." In: *Proceedings of the 7th ACM International Symposium on Pervasive Displays*. PerDis '18. Munich, Germany: ACM, 2018, 10:1–10:5. ISBN: 978-1-4503-5765-4. DOI: 10.1145/3205873.3205892 (cit. on p. 24).

[163] A. Voit, M. O. Salm, M. Beljaars, S. Kohn, S. Schneegass. "Demo of a Smart Plant System as an Exemplary Smart Home Application Supporting Non-Urgent Notifications." In: *Proceedings of the 10th Nordic Conference on Human-Computer Interaction*. NordiCHI '18. Oslo, Norway: Association for Computing Machinery, 2018, pp. 936–939. ISBN: 9781450364379. DOI: 10.1145/3240167.3240231 (cit. on p. 40).

[164] A. Voit, E. Stowell, D. Weber, C. Witte, D. Kärcher, N. Henze. "Envisioning an Ambient Smart Calendar to Support Aging in Place." In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp '16. Heidelberg, Germany: ACM, 2016, pp. 1596–1601. ISBN: 978-1-4503-4462-3. DOI: 10.1145/2968219.2968555 (cit. on pp. 24, 25).

[165] A. Voit, D. Weber, Y. Abdelrahman, M. Salm, P. W. Woźniak, K. Wolf, S. Schneegass, N. Henze. "Exploring Non-Urgent Smart Home Notifications Using a Smart Plant System." In: *19th International Conference on Mobile and Ubiquitous Multimedia*. MUM 2020. Essen, Germany: Association for Computing Machinery, 2020, pp. 47–58. ISBN: 9781450388702. DOI: 10.1145/3428361.3428466 (cit. on p. 24).

[166] A. Voit, D. Weber, N. Henze. "Qualitative Investigation of Multi-Device Notifications." In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. UbiComp '18. Singapore, Singapore: ACM, 2018, pp. 1263–1270. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274117 (cit. on pp. 20, 22, 24, 25, 27, 138).

[167] A. Voit, D. Weber, A. Imeri, A. Eidner, A. Tsoulos, D. Koch, K. Chen, M. Rottschäfer, R. Schweiker, S. Söhnel, V. Sabbatino, N. Henze. "Exploration of a Multi-Device Smart Calendar Platform for Smart Homes." In: *Proceedings of the 17th Inter-*

national Conference on Mobile and Ubiquitous Multimedia. MUM 2018. Cairo, Egypt: ACM, 2018, pp. 403–410. ISBN: 978-1-4503-6594-9. DOI: 10.1145/3282894.3289732 (cit. on p. 24).

[168] A. Voit, D. Weber, S. Schneegass. “Towards Notifications in the Era of the Internet of Things.” In: *Proceedings of the 6th International Conference on the Internet of Things*. IoT’16. Stuttgart, Germany: ACM, 2016, pp. 173–174. ISBN: 978-1-4503-4814-0. DOI: 10.1145/2991561.2998472 (cit. on pp. 24, 216).

[169] A. Voit, D. Weber, E. Stowell, N. Henze. “Caloo: An Ambient Pervasive Smart Calendar to Support Aging in Place.” In: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM ’17. Stuttgart, Germany: ACM, 2017, pp. 25–30. ISBN: 978-1-4503-5378-6. DOI: 10.1145/3152832.3152847 (cit. on p. 24).

[170] R. Want, T. Pering, G. Danneels, M. Kumar, M. Sundar, J. Light. “The personal server: Changing the way we think about ubiquitous computing.” In: *International Conference on Ubiquitous Computing*. Springer. 2002, pp. 194–209 (cit. on p. 162).

[171] D. Weber. “Notifications in a multi-device environment.” MA thesis. 2015 (cit. on p. 34).

[172] D. Weber. “Towards Smart Notification Management in Multi-device Environments.” In: *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI ’17. Vienna, Austria: ACM, 2017, 68:1–68:2. ISBN: 978-1-4503-5075-4. DOI: 10.1145/3098279.3119921 (cit. on pp. 20, 24).

[173] D. Weber, A. Exler, A. Voit, V. Pejovic, N. Henze, S. Gehring, T. Okoshi. “UbiTtention 2018: 3rd International Workshop on Smart & Ambient Notification and Attention Management.” In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*. UbiComp ’18. Singapore, Singapore: ACM, 2018, pp. 1190–1193. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274143 (cit. on pp. 22, 23, 25).

[174] D. Weber, S. Mayer, A. Voit, R. Ventura Fierro, N. Henze. “Design Guidelines for Notifications on Smart TVs.” In: *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video*. TVX ’16. Chicago, Illinois, USA: ACM, 2016, pp. 13–24. ISBN: 978-1-4503-4067-0. DOI: 10.1145/2932206.2932212 (cit. on pp. 20, 22, 27, 161).

[175] D. Weber, A. S. Shirazi, S. Gehring, N. Henze, B. Poppinga, M. Pielot, T. Okoshi. "Smarttention, Please!: 2nd Workshop on Intelligent Attention Management on Mobile Devices." In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '16. Florence, Italy: ACM, 2016, pp. 914–917. ISBN: 978-1-4503-4413-5. DOI: 10.1145/2957265.2965025 (cit. on p. 25).

[176] D. Weber, A. S. Shirazi, N. Henze. "Towards Smart Notifications Using Research in the Large." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 1117–1122. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2794334 (cit. on pp. 20, 25, 41, 42, 49, 162).

[177] D. Weber, A. Voit, J. Auda, S. Schneegass, N. Henze. "Snooze!: Investigating the User-defined Deferral of Mobile Notifications." In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. MobileHCI '18. Barcelona, Spain: ACM, 2018, 2:1–2:13. ISBN: 978-1-4503-5898-9. DOI: 10.1145/3229434.3229436 (cit. on pp. 20, 21, 27, 44, 60, 62, 70, 93, 98).

[178] D. Weber, A. Voit, T. Dingler, M. Kallert, N. Henze. "Assessment of an Unobtrusive Persuasive System for Behavior Change in Home Environments." In: *Proceedings of the 5th ACM International Symposium on Pervasive Displays*. PerDis '16. Oulu, Finland: ACM, 2016, pp. 245–246. ISBN: 978-1-4503-4366-4. DOI: 10.1145/2914920.2940340 (cit. on p. 20).

[179] D. Weber, A. Voit, A. Exler, S. Schröder, M. Böhmer, T. Okoshi. "Intelligent Notification and Attention Management on Mobile Devices." In: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM '17. Stuttgart, Germany: ACM, 2017, pp. 561–565. ISBN: 978-1-4503-5378-6. DOI: 10.1145/3152832.3152840 (cit. on p. 25).

[180] D. Weber, A. Voit, N. Henze. "Clear All: A Large-Scale Observational Study on Mobile Notification Drawers." In: *Proceedings of Mensch und Computer 2019*. MuC '19. Hamburg, Germany: ACM, 2019, pp. 361–372. ISBN: 978-1-4503-7198-8. DOI: 10.1145/3340764.3340765 (cit. on pp. 20, 21, 27, 52, 74).

[181] D. Weber, A. Voit, N. Henze. "Notification Log: An Open-Source Framework for Notification Research on Mobile Devices." In: *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive*

and Ubiquitous Computing and Wearable Computers. UbiComp '18. Singapore, Singapore: ACM, 2018, pp. 1271–1278. ISBN: 978-1-4503-5966-5. DOI: 10.1145/3267305.3274118 (cit. on pp. 20, 23, 25, 27, 201).

[182] D. Weber, A. Voit, G. Kollotzek, N. Henze. "Annotif: A System for Annotating Mobile Notifcations in User Studies." In: *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia*. MUM '19. Pisa, Italy: ACM, 2019, 24:1–24:12. ISBN: 978-1-4503-7624-2. DOI: 10.1145/3365610.3365611 (cit. on pp. 20, 21, 27, 52).

[183] D. Weber, A. Voit, G. Kollotzek, L. van der Vekens, M. Hepting, F. Alt, N. Henze. "PD Notify: Investigating Personal Content on Public Displays." In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI EA '18. Montreal QC, Canada: ACM, 2018, LBW014:1–LBW014:6. ISBN: 978-1-4503-5621-3. DOI: 10.1145/3170427.3188475 (cit. on pp. 20, 23, 27, 162).

[184] D. Weber, A. Voit, P. Kratzer, N. Henze. "In-situ Investigation of Notifications in Multi-device Environments." In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '16. Heidelberg, Germany: ACM, 2016, pp. 1259–1264. ISBN: 978-1-4503-4461-6. DOI: 10.1145/2971648.2971732 (cit. on pp. 20, 22, 27, 138).

[185] D. Weber, A. Voit, H. V. Le, N. Henze. "Notification Dashboard: Enabling Reflection on Mobile Notifications." In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '16. Florence, Italy: ACM, 2016, pp. 936–941. ISBN: 978-1-4503-4413-5. DOI: 10.1145/2957265.2962660 (cit. on pp. 20, 21, 25, 27, 42, 98).

[186] M. Weiser. "The Computer for the 21st Century." In: *Scientific american* 265.3 (1991), pp. 94–105 (cit. on pp. 15, 209, 216).

[187] T. Westermann, S. Möller, I. Wechsung. "Assessing the Relationship Between Technical Affinity, Stress and Notifications on Smartphones." In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. Copenhagen, Denmark: ACM, 2015, pp. 652–659. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2793684 (cit. on p. 72).

[188] WhatsApp. *Connecting One Billion Users Every Day*. https://blog.whatsapp.com/10000631/Connecting-One-Billion-Users-Every-Day. 2017 (cit. on p. 72).

[189] S. Whittaker, C. Sidner. “Email Overload: Exploring Personal Information Management of Email.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’96. Vancouver, British Columbia, Canada: ACM, 1996, pp. 276–283. ISBN: 0-89791-777-4. DOI: `10.1145/238386.238530` (cit. on p. 67).

[190] F. Wiehr, A. Voit, D. Weber, S. Gehring, C. Witte, D. Kärcher, N. Henze, A. Krüger. “Challenges in Designing and Implementing Adaptive Ambient Notification Environments.” In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. UbiComp ’16. Heidelberg, Germany: ACM, 2016, pp. 1578–1583. ISBN: 978-1-4503-4462-3. DOI: `10.1145/2968219.2968552` (cit. on pp. 24, 25).

[191] J. Wiese, T. S. Saponas, A. B. Brush. “Phoneprioception: Enabling Mobile Phones to Infer Where They Are Kept.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 2157–2166. ISBN: 978-1-4503-1899-0. DOI: `10.1145/2470654.2481296` (cit. on p. 151).

[192] K. Wolf, S. Schneegass, N. Henze, D. Weber, V. Schwind, P. Knierim, S. Mayer, T. Dingler, Y. Abdelrahman, T. Kubitza, M. Funk, A. Mebus, A. Schmidt. “TUIs in the Large: Using Paper Tangibles with Mobile Devices.” In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’15. Seoul, Republic of Korea: ACM, 2015, pp. 1579–1584. ISBN: 978-1-4503-3146-3. DOI: `10.1145/2702613.2732863` (cit. on p. 24).

[193] S. Yoon, S.-s. Lee, J.-m. Lee, K. Lee. “Understanding Notification Stress of Smartphone Messenger App.” In: *Proceedings of the Extended Abstracts of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI EA ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 1735–1740. ISBN: 978-1-4503-2474-8. DOI: `10.1145/2559206.2581167` (cit. on pp. 16, 39).

[194] F. Yuan, X. Gao, J. Lindqvist. “How Busy Are You?: Predicting the Interruptibility Intensity of Mobile Users.” In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI ’17. Denver, Colorado, USA: ACM, 2017, pp. 5346–5360. ISBN: 978-1-4503-4655-9. DOI: `10.1145/3025453.3025946` (cit. on p. 110).

All URLs cited were checked in October 2023.

List of Figures

List of Tables

List of Acronyms

API	Application Programming Interface
AR	Augmented Reality
BLE	Bluetooth Low Energy
CSS	Cascading Style Sheets
DRM	Day Reconstruction Method
ESM	Experience Sampling Method
EUR	Euro
GPS	Global Positioning System
GUI	Graphical User Interface
HCI	Human-Computer Interaction
HD	High Definition
HTML	Hypertext Markup Language
ID	Identifier
IM	Instant Messenger or Instant Messaging
IoT	Internet of Things
JSON	JavaScript Object Notation
LED	Light-Emitting Diode
M	Mean
Md	Median
OS	Operating System

PBKDF2	Password-Based Key Derivation Function 2
PC	Personal Computer
PD	Public Display
RQ	Research Question
SD	Standard Deviation
SDK	Software Development Kit
SHA	Secure Hash Algorithm
SMS	Short Message Service
SQL	Structured Query Language
TV	Television
UI	User Interface
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
UX	User Experience
VoIP	Voice over Internet Protocol
VR	Virtual Reality