LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
Department "Institut für Informatik"
Lehr- und Forschungseinheit Medieninformatik
Prof. Dr. Heinrich Hußmann

Masterarbeit

Time-constrained access control for mobile devices

Fabian Hartmann
mail@fabian-hartmann.de

Bearbeitungszeitraum:	18.01.2015 bis 03.08.2015
Betreuer:	Daniel Buschek
Verantw. Hochschullehrer:	Prof. Dr. Florian Alt

Zusammenfassung

In dieser Arbeit wurde ein neuartiges Entsperrkonzept entwickelt, welches einen alternativen zeitbeschränkten Zugriff auf Smartphones ermöglicht. In die Entwicklung des Konzepts wurde auch bestehende Literatur zur Nutzung von Smartphones, Entsperrverhalten und anderen alternativen Entsperrmethoden mit einbezogen. Das fertige Konzept wurde als installierbare Android Applikation names SnapApp implementiert. Diese vereint die bereits bekannten Entsperrmethoden PIN und Slide-to-Unlock in einem einzigen Sperrbildschirm. Der Nutzer kann somit entscheiden, ob er uneingeschränkten Zugriff auf das Smartphone durch die Eingabe eines PINs haben möchte, oder ob er lieber eingeschränkten Zugriff durch die Betätigung des Sliders haben möchte. Es wurde eine Langzeit Feldstudie durchgeführt, bei welcher der SnapApp Prototyp auf den Smartphones von 18 Teilnehmer installiert wurde. Diese testeten den neuartigen Sperrbildschirm über einen Zeitraum von 30 Tagen. Die Auswertungen ergaben, dass SnapApp PIN Eingaben insgesamt zu 20% reduzieren konnte, wodurch wertvolle Zeit eingespart wurde. Die Sicherheit wurde dabei nicht beeinträchtigt, da die Mehrheit der Nutzer ihre SnapApp Einstellungen individuell anpasste. Sie konfigurierten die maximale Sitzungsdauer, die Verfallszeit der verfügbaren Kurzzugriffe, sowie die Blacklist, die geschützte Applikationen enthält, welche während des Kurzzugriffs geschützt sein sollen. Das Feedback aus den Umfragen ergab auch, dass die Sicherheit aus Nutzersicht nicht gefährdet war. SnapApp kann an verschiedene Nutzerbedürfnisse angepasst werden und wurde dadurch sowohl von eigentlichen PIN und Entsperrmuster Nutzern, als auch von Anwendern ohne sichere Entsperrmethode akzeptiert. Des Weiteren benötigt SnapApp für seine Funktionalität keine zusätzliche Hardware oder Sensoren und kann dadurch auf jedem beliebigen Android Gerät installiert und genutzt werden.

Abstract

In this thesis, a novel concept to unlock smartphones was elaborated. It enables an alternative time-constrained session on the smartphone for short access. Before the concept was developed, existing research about smartphone usage, unlock behaviors and non-standard unlock methods was explored. The final concept was implemented as an installable Android application afterwards. The prototype application called SnapApp combines the two already known unlock methods PIN and slide-to-unlock in one lockscreen. The user can decide to either get full access by prompting PIN or to get constrained short access by using slide-to-unlock. A longitudinal field study was conducted by installing the prototype on the smartphones of 18 participants, who tested the new lockscreen for a duration of 30 days. Results revealed that SnapApp was able to reduce PIN prompts by 20% in total, which also saved valuable time. The security was not impaired, as the majority of the users has individually configured the maximum session lengths, the expirations of available short sessions and the blacklists, which contain apps protected of usage during short access. This was also confirmed by the feedback questionnaires of the study. SnapApp can be adapted to different user needs and was thereby equally accepted by PIN, pattern and swipe users. Besides, the prototype requires no further hardware or sensors and can thus be installed on any Android smartphone.

Aufgabenstellung

Recent research has shown that users waste a lot of time unlocking their smartphones, although they often only interact for very short time frames after unlocking. Interaction times are often shorter than 20-30 seconds. For example, a user might just briefly check for expected delays in a public transport app or read some headlines in a news app.

Thus, granting access for a very limited amount of time *without unlocking* may present a valuable approach to save users' time during such short interactions. At the same time, many serious attacks may be impossible to perform in less than 20 seconds (e.g. hacking an online banking account).

This thesis will investigate opportunities and challenges of this concept of time-constrained access control on mobile devices.

The task includes:

- Comprehensive survey of related work
- Development and implementation of an Android lockscreen-replacement app, which enables time-constrained access control
- Systematic data collection with a user study or deployment "in the wild"
- Analysis of the collected data

Requirements:

- Interested in the topic of usable privacy and security
- Familiar with Android programming
- Independent scientific work and creative problem solving

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, 3. August 2015

. . . .

Contents

1 Introduction

1.1 Motivation

The modern smartphone is not only a phone, but also a platform running an advanced mobile operating system, which enables the installation and usage of software called applications. With the ongoing growth and development of the internet and the hardware enabling network connectivity built into smartphones, data can be accessed from anywhere. The variety of applications or short apps provides the user several tools to complete different tasks. Depending on the application, private and sensitive data can be accessed by the phone or is stored on it. This may include data like contacts, photos, videos, e-mails, text messages, banking accounts and a great number of credentials for different online accounts. Thereby, the access to the smartphone should be protected by an adequate authentication method [28]. Common unlock methods are PIN, password and pattern, whereas the latter is a graphical password only available on the Android platform [60].

Recent research has shown that smartphones are used for very short interactions throughout the day. With a daily average of 83 activations and 48 unlocks, users waste a lot of time when they employ a secure unlock method like PIN or pattern [21]. As a consequence of the needed effort, a lot of people reject secure unlock methods out of convenience. They use the faster methods swipe or slide-to-unlock instead, which only requires a short gesture to unlock the phone without providing any credentials. Studies have shown that between 28.6% and up to 64.0% of all users do not use a secure lockscreen at all [15, 21, 11], which proves there is an urgent need for action.

This thesis presents a new time-constrained access control concept, its implementation as a prototype called *SnapApp* and its evaluation in a field study. The proposed system tries to increase convenience and save time by reducing explicit authentication. This is accomplished by providing a choice between PIN for unlimited access and a second alternative unlock method, which allows a time limited access to the phone without providing any credentials. The concept is expected to be accepted by both secure unlock method users and their rejectors, as it is a trade-off between convenience and security.

1.2 Research question

The thesis investigates the question, whether it is possible to reduce explicit authentication (in this case PIN) to save time by a new time-constrained access control concept without impairing security.

1.3 Content overview

The thesis is divided into the sections “Related work”, “SnapApp”, “Field study”, “Results”, “Discussion” and “Conclusion”. After this introduction, section 2 gives an overview about smartphone usage, unlock behaviors and other alternative unlock methods and concepts. Section 3 describes the concept development, the resulting concept and the features of the implemented prototype. In section 4, the design, the procedure, the participants and the questionnaires of the field study and its realization is explained, which investigates the usage and unlock behaviors of the implemented prototype. Thereafter, the field study results are presented in section 5, which is structured into data processing, logged data and the different questionnaires. Finally, in section 6 the results are discussed before a final conclusion is drawn in the final section 7.

2 Related Work

This section gives an overview of current data about smartphone unlocking behaviors and provides a collection of concepts and implemented prototypes of alternative unlocking methods, which influenced and helped to develop and implement SnapApp.

2.1 Smartphone usage and (un)locking behaviors

Developing a new unlock method requires to understand the user. What unlock method is used and why? How often does the user unlock the phone and how long does it stay unlocked? What apps are launched and which data is accessed? Does the user want protection for every app or not? The following papers investigated these and more questions and helped to understand what users do and need. The obtained insights influenced the design and the development of SnapApp.

Harbach et al. [21] conducted a online study and asked the users about concerns and motivations for or against locking their smartphones. Furthermore, a longitudinal field study was realized collecting both qualitative and quantitative data about unlocking behavior, risks and user perception.

The online survey (n = 260) revealed that only 42.7% of the participants use a secure unlock method (PIN, password or pattern), while the rest is using insecure unlock methods (no lockscreen or slide-to-unlock). When users with secure unlock methods were asked for their motivations to lock, all answers were within four main topics: general protection goals (e.g. control phone access), protecting information (e.g. accounts, photos), protection from specific scenarios (e.g. lost/stolen phone) and protection from attackers (e.g. unwanted persons, own children). Participants without a secure unlock method indicated inconvenience and absence of threat as their reasons for not using one. As result of the risk evaluation, the most voted worst case scenarios among all users were losing the phone and having to buy a new one (52.7%) and losing all the data on it (20%). While these scenarios could not be prevented by a secure lockscreen, account and data abuse on a lost phone could, as well as app and data abuse on an unattended phone (altogether 26.1%). Analyzing the critical incidents users had experienced (e.g. snooping partners, friends abusing accounts, lost phone) made clear that many of those also could have been averted by using a secure unlock method.

The longitudinal field study (n = 52) was accomplished by installing a logging app on every participant's Android smartphone for the duration of four weeks. The app logged four device states: screen on unlocked / locked and screen off unlocked / locked. Each log entry was combined with the current timestamp. Besides the collected quantitative data, there were also two types of mini-questionnaires randomly popping up on the screen during the field study. The first questionnaire was about rating the unlock process (satisfaction and type of accessed data) and shoulder surfing (the possibility of it, how likely and severe an attack would be, who could be the attacker and the type of environment). The other questionnaire concentrated on the interval since the last unlock. It asked whether a lockscreen was necessary, an unwanted access would have been possible and whether the lockscreen would annoy the user. Further, the type of the environment had to be assigned in that particular situation, too. As results, Harbach et al. measured 47.8 (sd = 26.4, median = 42.1) unlocks and 83.3 (sd = 43.0, median = 83.8) activations per day. These numbers differed as participants did not always unlock their phones when they took a look at it. The unlocking process (time difference from screen on locked to screen on unlocked) took on average 2.67 seconds (sd = 8.46s, median = 1.26s) without a secure unlock method, 3.0 seconds (sd = 13.3s, median = 1.69s) with pattern and 4.7 seconds (sd = 20.72s, median = 2.85s) with a numeric PIN. The average of all sessions (screen on to screen off) lasted 70.3 seconds (sd = 241.5s), while the sessions when the device was unlocked, too, had an average length of 104.1 seconds (sd = 193.9s, median 45.6s). The distribution of users with secure unlock methods was higher (67.3% with PIN or pattern) than in the online survey (42.7%). During 27 days, the phones were used 43.0 hours (sd

= 22.1h, median = 41.2h) on average, whereby 2.9% (range 0.6% to 9.0%) of the time were just unlocking the phone. The evaluation of the mini-questionnaires showed that participants without a secure unlock method were obviously more satisfied with their method than those with secure ones. Many secure unlock users were especially annoyed when applying their method in private environments. Another two outcomes were 74.7% of the accessed data was not sensitive and only in 0.3% of the situations a shoulder surfing attack with a severe or very severe impact would have been possible. The gained data shows that out of 3410 sampled situations, 62.0% have been in a private, 20.2% in a semi-public and 17.7% in a public environment. The questions asked in the debriefing interview about the reasons for or against a secure locking method were matching the previous ones from the online study with only a few exceptions.

This paper gives new insights about people's smartphone unlock behaviors. The results show that a big part of the users is still refusing a secure unlock method due to the caused inconvenience or by seeing no threat not using one. The high number of unlocks and the noticeable time difference in applying a faster, but insecure unlock method compared to a secure one shows the need for improvements. In contrast to the deniers, those applying a secure method show partly high and in general much more dissatisfaction about their used method. The average session time lengths indicate commonly short usages as being standard, which is in favor of deploying short time-constrained access control on devices. A new time-constrained method like SnapApp could use slide-to-unlock and combine it with the short session lengths and mainly access to insensitive data to save time while keeping security and increasing usability. Harbach et al. had implemented mini-questionnaires in their app, which offered great new views during situations in real-time. SnapApp adopted the in situ mini-questionnaires with different questions and uses a more extended, but similar approach for logging the smartphone usage on Android devices. Another interesting finding detected by the study is that shoulder surfing with possible severe impact appeared at a very low risk. The paper showed that many people are completely rejecting secure unlock methods or are unsatisfied while using one.

Egelman et al. [15] also researched the unlocking behavior, risk perception and awareness of sensitive data on smartphones. They conducted preliminary qualitative interviews and confirmed their findings with a follow-up online survey. Finally, an online experiment on sensitive data contained in e-mails completed the study.

The interview participants (n = 28) were asked about their general usage, installed apps and accounts, backup, locking and sharing behavior on their smartphones. A secure unlock method was used by 71.4% (18 PIN, one pattern, one fingerprint) of the users, of which 40% voted yes for ever having been annoyed by it. Egelman et al. found out that the annoyance was not caused by remembering the PIN, but by the unlock procedure itself. Reasons for using a secure unlock method were mainly based on specific scenarios (70%, e.g. snooping friends and family, children using the phone, stolen phone), followed by generic reasons (30%). The remaining 28.6% of the users without a secure unlock method justified their refusal primarily with missing motivation, inconvenience or having no concerns without it. When the participants were asked about perceived threats they listed online identity theft, being impersonated online, privacy invasion (e.g. photos, contacts) or access to apps with payment options and stored financial data.

The online survey (n = 2518) was used to quantify the provided answers from the interviews by summarizing and displaying them as multiple-choice questions. The survey showed 58% in favor of a secure unlock method. One thousand responses were provided for and against secure locks (500 each). The most given answers for using one were preventing access by unknowns (55%), followed by controlling the usage by friends or family (23%) and considering it as a simple procedure (20%). In contrast to that, the reasons inconvenience (34%), no threat by access to the data on the phone (26%) and carelessness (19%) led to the rejection of a secure lockscreen.

Everyone of the interviewed participants had access to their primary e-mail account on the smartphone. As a consequence, an online survey (n = 995) was conducted where people were asked to find sensitive data in their e-mail account. The results showed that 35% were able to find at least one e-mail containing either fully or partly the social security number, bank account number, credit card number, debit card number, passwords, date of birth or the home address. However, more sensitive data was found on the devices locked by a secure method (38%) than on those without one (27%).

The paper of Egelman et al. corroborates Harbach et al. as the results on unlock behaviors correlate. The provided reasons for or against the deployment of a secure unlock method are protection of specific scenarios and inconvenience, no concern or missing motivation. Egelman et al. could also find annoyed secure lock users who were using it anyway, as in the users' opinions its benefits outweighed the effort. The amount of rejectors is also similarly high. The e-mail account being a big threat is a new finding, as it can evidently contain a lot of very sensitive data and is often used for retrieving passwords. A divergent result compared to Harbach et al. is that private environments do need a secure locking mechanism, as almost a quarter of the participants using one stated its purpose would be the regulation of access for family and friends. The interviews revealed a higher risk of intruders in private settings than of shoulder surfing in general. Egelman et al. argue the secure lock decision cannot just be dependent on the context and the environment, but has to take into account the personal privacy preferences and the stored sensitive data on the phone. As a result, SnapApp lets the users alone decide which apps and data are available for short access.

Harbach et al. and Egelman et al. investigated unlock behaviors in general. Kurkovsky et al. [32] concentrated on a special group and asked young *digital natives*[1] (n = 330) aged 18 to 25 and who should know better in the authors opinions, how they use their mobile phones day-to-day and about their attitudes towards security and privacy. Against expectations the target group being cleverer than older users and although the study pointed out that the threats (similar compared to the previous papers) were known to the participants, only 33% used a PIN, the most common secure unlock method in 2010.

Van Brugen et al. [58] did not only examine, but tried to influence user locking behavior during a field study (n = 150) by sending messages themed with the topics morality, deterrence and incentives. After all, the messages made one third of all participants employing the next higher security lock (from no lock to pattern or from pattern to password). Like the previous papers, the participants without a secure unlock method had the percentage of 35% and were the most difficult group to influence.

2.2 Alternative unlock methods and concepts

Authentication on mobile devices is a big research area and can be divided into three different types of authentication - implicit, explicit and mixed as a combination of both [21]. Implicit methods authenticate the user repeatedly by monitoring and analyzing usage patterns or sensor data within certain time spans. As they run in the background, they can either be used additionally to an explicit method to increase security (e.g. [13]) or as replacement of the primary authentication method (e.g. [12, 18, 43, 54]) [27]. However, the time needed for the authentication causes delays, by which a deployment as a direct unlock method alone becomes ineligible for most of them [21]. Explicit methods can be split into biometrics (*What you are*), token-based (*What you have*) and knowledge-based authentication (*What you know*) [52]. The latter is commonly known as passwords, PINs and patterns, which have to face several threats, such as shoulder surfing ([6, 36, 49, 55]) and smudge attacks ([5]). Pattern is an graphical unlock method available on the Android platform (*for details see [60]*).

[1] Young people who grew up with computers, the Internet and other technology [32]

2.2.1 Improving security

The research for improvements of security for knowledge-based authentication methods against these threats is very active. Shoulder surfing attempts try to be tackled by obfuscating the user input. SwiPIN for example divides the PIN pad into two colored areas and assigns the five digits in each half randomly to five gestures (up, down, left, right and tap). Instead of pressing the digit keys directly, the user performs the gesture for the digit anywhere in the associated area [59]. Arif et al. used the same gestures for their approach, but the starting point is exactly the desired digit key. Each digit of the PIN is combined with one user selected gesture and is only accepted when the entry is done accordingly (e.g. the PIN "6 tap, 4 up, 3 down, 7 left" would require the user to tap on digit 6 and perform the directed gestures starting on the other digit keys) [3]. More complicated is the Tactile One-Time Pad, which adds a random number to each PIN digit. First, the random number is passed to user as countable vibrations. After that, modulo 10 of the sum of the random number plus the PIN digit equals the new digit which has to be entered by the user. By this, the PIN being entered changes every time without changing the original PIN [57]. De Luca et al. developed a back-of-device authentication, which hides the pattern input by moving it to a touchscreen on the back of the smartphone. Compared to the normal pattern method, their approach uses three simplified short gestures in a row to authenticate the user [14]. TinyLock is another solution for patterns, which are especially prone to smudge attacks. It downsizes the pattern grid and shows a second one, which displays the drawn user input of the first one. After finishing the pattern, the user has to turn the "virtual wheel" by drawing a circle on the same spot where the grid has been shown before. By this, the authentication procedure is completed and the new circle smudges distort the ones of the previous drawing [33].

2.2.2 Improving usability

Don Norman said once "The more secure you make something, the less secure it becomes. Why? Because when security gets in the way, sensible, well-meaning, dedicated people develop hacks and workarounds that defeat the security." [44]. The previous presented examples improve the security significantly, but every one of them needs to be practiced and clearly slows down the authentication itself. The data analysis in section 2.1 has shown that the problem is not security, but usability - a big group of people is rejecting a standard secure unlock method or using one while being unsatisfied with it. "Solving the problems of passwords doth not a successful authentication mechanism make." claimed Maguire and Renaud [36], who rather suggest to concentrate on an authentication scenario with people, places and purpose. Therefore, the next examples for alternative unlock methods target this group by trying to enhance usability.

Arif et al. tried to encourage users without a secure unlock method to start using one by modifying slide-to-unlock into two new knowledge-based authentication methods: sequential and timed slide-to-unlock. Both methods divide the display vertically into three different zones A, B and C. With the sequential slide-to-unlock method, a passwords is drawn as a stroke starting in a free choosable initial zone and entering other zones by swiping (e.g. the password "BABC" with a length of four would start in the middle zone B, enter A by swiping left and then go back to C over B). By this, 70 unique passwords could be generated with a maximum password length of seven. The timed slide-to-unlock method is a variation of the sequential one and extends each zone with three timeframes. Then the user keeps the finger still, a timer is fired and shows a progress bar above the fingertip indicating the current timeframe. The timeframe changes (from first to second, second to third, third to first) every 200ms and is selected, when the user starts swiping again (e.g. for the password "B 1st, C 3rd", the user would tap in zone B, wait for the progress bar to show up and start swiping to zone C before the progress bar changes into the 2nd timeframe. In zone C, the finger would have to be kept still until the progress bar reaches the 3rd timeframe and then end the touch.). The variation expands the unique passwords to 473,536 possibilities with a maximum passwords length of seven. While the timed method was slower, the sequential slide-to-unlock could compete with a normal four-digit PIN at unlock speed. The authors developed a method which simplifies the change from no password to a at least a semi-secure and easy to use one (sequential slide-to-unlock) [4].

With Waving Authentication by Hong et al., the smartphone identifies its owner by being waved. The matcher of the unlock method uses eight distinguishing features in the acceleration motion to detect whether the person waving the device is an attacker or not. The classifier has to be trained for 30 to 50 trials before Waving Authentication can be used with a tested average true positive (owner is accepted) rate of 92.83% [25]. Srilenkha and Jayakumar presented a similar method which also uses the accelerometer and unlocks the phone by shaking it [50]. As both methods use personal motion features, they belong to biometric authentication.

Face recognition is also a biometric authentication method to determine the device holder. The most modern smartphones are equipped with front cameras and can read the person's face which is normally looking frontally at the screen. The two most important factors for face recognition are alignment and illumination of the face [61, 46]. While the alignment is not a problem due to the head position while holding a smartphone, light conditions are unpredictable and can be extreme. Wilson and Chen recommended a software, which authenticates the user's face and saves every sample in case of a true positive (owner has been detected and is the true owner) for comparison. The security of the system benefits from each new sample, as the facial characteristics can change rapidly (e.g. beards, head hair, glasses or sunglasses) [62]. Face recognition is a good addition to knowledge-based authentication, as it can quickly detect the user without any direct input. However, smartphones usually do not have a flash at the front to compensate bad illumination and the method's detection accuracy is still improving. The possibility of false negatives (owner is rejected) and true negatives (intruder is accepted) can occur any time.

The most convenient biometric way of authentication is using fingerprints. A smartphone user only has to hold the finger onto the fingerprint scanner while the phone compares the scanned biometric characteristics with the samples stored on the device within milliseconds. Scanning fingerprints is supported natively by the major smartphones operating systems [2, 34], but the required sensors for the method are extra hardware [13], which is why they can only be found in high-end smartphones yet. The sensors are well integrated into the home buttons of the phones which are used to activate the phone anyway. With their introduction, many users were concerned about privacy, storage and safeguarding measures of their physiological data on the devices [47]. Fingerprints face the risk of being copied and used as plastic replica [54] - proven in a famous example by the Chaos Computer Club in 2008. The hacker group published one fingerprint of Germany's former Federal Minister of the Interior Wolfgang Schäuble, which was usable to fool fingerprint scanners and was taken from a glass that the minister had used [9]. Nevertheless, it is the most secure biometric method for smartphones as it is used to authorize mobile payments [1, 34, 45].

Another method to authenticate is using physical tokens. While there have been early prototype tokens generating magnetic fields or acoustic signals which got picked up by the smartphone's compass or microphone [8], newer tokens are realized with NFC. Flügge et al. implemented an Android app, which can be unlocked by PIN or by a NFC-tag. Once the unique NFC-tag is bound to the phone in the application setup, users can either use the PIN or hold the token in proximity to the smartphone to unlock it [17, 63]. Passive NFC-tokens are low-cost and a convenient alternative to knowledge-based authentication. On the downside, they are an extra item which can be forgotten or get lost.

Google has introduced On-body detection, which is not an unlock method, but reduces unlock prompts. It uses the smartphone's accelerometer to detect if the phone is being carried in the hand, pocket or a bag. After a initial unlock prompt, the phone stays unlocked as long as it is on the body of the person and does not require the user to authenticate again. When the smartphone is put down, the device automatically locks itself and asks for authentication at the next usage. The disadvantages are that the method might not work in moving vehicles (e.g. cars, boats, planes) and the smartphone stays unlocked when it is handed to or stolen by another person as it cannot differ between owner and other people [16, 19]. Apart from that, On-body detection is a very effective way to reduce unlocks.

On-body detection is a part of Android's SmartLock, which includes several of the previously enumerated authentication methods and other possibilities to reduce unlocks. It supports disabling the lock by face and voice recognition and NFC-tags and keep the phone unlocked by connected bluetooth devices and trusted locations. Bluetooth devices (e.g. a headset or a smartwatch) need to be paired and added to the smartphone's trusted devices list, while locations have to be added to the trusted places list by providing their coordinates. When a trusted device is connected or when the smartphone's turned on GPS detects close proximity to a trusted place, the phone stays unlocked [19].

2.2.3 Against all-or-nothing

As the previous section has shown numerous approaches to enhance usability by simplifying the unlock procedure or reducing the number of prompted unlocks. Nevertheless, none of them has tried to change the binary security model - the phone is either unlocked or fully locked. Users are concerned about their sensitive data on the smartphones, but instant access to apps is more important than taking the effort and giving up convenience [40]. The next papers present alternatives to the all-or-nothing models on smartphones.

Multi-user approaches

In 2006, Stajano [52] discussed the compatibility of usability and security on a PDA[2] - the precursor of nowadays smartphones. In his opinion they were antithetic as the available all-or-nothing approach was users are either having a password or not. The inconvenience of repeatedly entering a password with each access was not accepted by the big majority of the users - *"Whenever security and usability fight, usability wins."* [52]. The amount of sensitive data stored on the device was not as extensive as today, but it started already growing [51]. Consequently, proper security was needed, but the usability had to be improved.

Therefore, he questioned at first the need for authentication at all. The PDA would be the "archetypal single-user machine" [52] with having only one privileged user. This user was needed to access the PDA, regardless of what application or data was used. He suggested introducing "hats", a multiuser concept with each hat having different privileges on the device. There should be a private hat for network access, another one for file system access and so on. The public access alias "no hat" would require no password while the other private hats ask for one when being chosen. Switching hats should be possible any time, although exchanging data between them would be a challenge as each hat should run as a sandbox. Combined with the alternative authentication methods, the new hats system should improve usability while keeping security.

The all-or-nothing authentication status described by Stajano is up to date until now. The ongoing development of the internet and the smartphones increased the amount of sensitive data stored on the devices, e.g. user photos and videos, accounts for social networks, cloud storage, messaging services and banking. Smartphones allow taking photos, showing notifications and answering calls while locked, but besides that no app or other functionality can be fully accessed. A device has to be equally unlocked for playing Tetris, text messaging and for doing online banking - no matter what data is accessed and how long the device is used. The hat system with public and private access was the first idea of escaping the stuck binary concept and dynamically allow access to the device.

[2] Personal Digital Assistant, a handheld device used as dictionary, organizer, address book, personal notes and calculator. Better ones had an e-mail client and a web browser.

Karlson et al. [28] investigated smartphone usage patterns and sharing practices in a user study (n = 12). They found out sharing was was quite common, but users showed security and privacy concerns dependent on the person the device was shared with. Their proposal of a new restricted guest mode was broadly welcomed by the participants as an alternative security concept. Further questions about what guest actions on apps and data the users would allow revealed that not one, but three different access levels for the guest mode would suit the participants needs best. Karlson et al. concentrated on the user perception of a new security model and confirmed Stajano by their study results, who was mainly focussed on implementation challenges. They determined a need for new security models with more than two states and stated that a fast access to restricted guest profiles could make a "valuable contribution toward addressing privacy and data integrity concerns" [28].

Ni et al. [42] were the first to actually implement a lightweight software called DiffUser with a new security model. DiffUser stands for differentiated user access control model and introduces three user types: administrator, normal user and guest user. While the administrator has full rights, a normal user cannot install or uninstall critical apps. The guest user has very limited privileges. They deployed DiffUser on a T-Mobile G1 running one of the first Android versions and replaced the launcher, which is the Android main screen. Changing between the user profiles can quickly be done by the new fast switch button and updates the shown available apps on the home screen depending on the selected profile. Ni et al.'s prototype has proven that a flexible multi-user system can be successfully deployed on smartphones, too.

Context-aware approaches

Smartphones have multiple built in sensors and various connectivity possibilities, which can determine the current context. The next papers use these contexts to regulate authentication.

Gupta et al. [20] implemented a context profiler, which detects familiarity and safety of the current location and decides the needed authentication level based on the measures. The smartphone detects its location by GPS and memorizes visited places. The familiarity of places is established by scanning bluetooth and WiFi and checking for already known recurring devices. Dependent on the determined authenticity level, the device can be unlocked by slide-to-unlock or by PIN. Changing the location can increase, but not decrease the authenticity level - therefor, unlocking the phone by PIN is required as a security precaution. As the context profiler is learning autonomously, the user cannot confirm the familiarity of locations.

Hayashi et al. [22] developed a framework called CASA, context-aware scalable authentication, and conducted three user studies for its investigation. The framework can use multiple passive factors (e.g. location, user behavior) to rule an appropriate explicit authentication method (e.g. PIN, password or slide-to-unlock for none). The first study (n = 36) explored the mobile patterns of the users and revealed they spend most of their time at two places (home and work) where also 60% of all phone activations were counted. Consequently, a second field study (n = 32) was conducted with CASA installed on the devices using location as main passive factor. After authentication, users were prompted in a in-situ questionnaire to categorize their current location (home, work or other). Due to the negative usability results (password entry being a too high burden), a third study (n = 18) with an adapted CASA was conducted. This time, the only unlock methods were slide-to-unlock and PIN. Additionally, nearby known computers where the user is logged in were introduced as another new passive factor. With the new configuration, CASA could cut the number of explicit unlocks down by 68% and strongly interest people to use CASA instead of an insecure unlock method.

Application-centric approaches

Hayashi et al. [23] interviewed (n = 20) users in a lab study about the all-or-nothing device access, who were owning both a smartphone and a tablet and were sharing their devices with others on a regular basis. They suggested a new device access approach, for which each participant had to categorize the most important apps installed on their phone into the three accessibility categories *Always available* (40%), *Available after Unlock* (40%) and *Split* (20%). One example for the split category was using the phone functionality - while without being unlocked only local calls were available, international numbers could be called after authentication. The categorization on tablets showed very similar results. Regardless whether they used a secure unlock method or not, all participants deemed the all-or-nothing access to be a bad solution for both device types and 14 out of 20 favored the alternative new method to the existing ones. The results showed that over half of the apps (Always available and Split together) should be partly or fully accessible without any authentication. Apps with more sensitive data available were likely to be in the categories Available after Unlock and Split (e.g. communication), while those containing no sensitive data (e.g. navigation) could be mostly found in the Always available category. However, the preferences of users differed in the distribution of apps into the categories individually. As the study had also a focus on sharing of devices, it dealt with threats as well. The most concerns were expressed in the private environment - the users' children were named as main threat (e.g. accidentally deleting data or purchasing apps without parental permission).

According to these results, location-based authentication alone cannot meet the users' needs, as it disables secure unlocks at work or at home. It has been proven proven, that several threats occur especially in private environments [41, 15]. Since almost three quarters of the accessed apps and data are not sensitive [21] and users want to have instant access to half of their apps without unlocking [23], the next papers present application-centric authentication solutions.

Riva et al.'s progressive authentication [48] determines the user's confidence level of authenticity by a combination of multiple factors and decides then dependent on the protection level of the requested app if explicit authentication is needed or not. The used factors are biometric signals (face and voice recognition), behavioral signals (check if user behavior differs from the past recurrent behavior, includes location), possession signals (nearby devices which are known to be owned by the user) and PINs and passwords. The apps of the prototype were classified as public, private and confidential. While public apps need no authentication, private and confidential are only accessible with higher authenticity levels. Another important feature of the system is continuity - the unlocked state and the confidence level for authenticity are kept as long no negative signals (e.g. other voices, different location) occur. In a lab study (n = 9), progressive authentication achieved an explicit authentication reduction of 42%, however, it was not tested in the field.

Micallef et al. [38] proposed a similar app-driven mobile authentication model which also uses passive factors and categorizes the installed apps into three levels. Level 1 apps do not access or store sensitive personal data, in level 2 applications do it partially and in level 3 apps have continuous access and are constantly storing sensitive data. While level 1 apps require no passive factors, in level 2 location and orientation are utilized to check if the sensor data reflects usual patterns of the user. For the most critical level 3 applications, more sensors (location, light, noise and orientation) are used to analyze user behavior. If the system detects any irregularities, the user is prompted to enter PIN or password. For their prototype, Micallef et al. used the standard categories from the app market (level 1: e.g. sports, news, games; level 2: e.g. shopping, browser, productivity; level 3: e.g. finance, communication, settings) to classify the installed applications. The future plan was involving users to confirm the classifications. Compared to Riva et al., their app-driven solution automates the app categorization and eases the burden for the user, but it was also not tested in the field.

Kahn et al. [30] took this one step further as they recommended building a library available for every app developer to let the app decide when and how active authentication is needed. They conducted a field study (n = 32) over ten weeks to analyze touch and swipe behavior in apps to test whether the data can be used for implicit authentication of the user. The results showed that a device-centric approach fails, as some apps cannot provide enough distinguishable data (e.g. maps) while others do. They concluded an application-centric approach would be the best solution. Apps know when sensitive data is accessed and how sensitive the data is, which is why the decision of when and how an active authentication is needed should by delegated to applications themselves. The authors noted that the approach causes a development overhead and needs an interface in form of a library for the developers, but on the other hand the authentication overhead for the user can be reduced.

Micallef et al. [37] conducted a field study based on the app and data of Kayacik et al. [29], which also uses location detection combined with multiple other environment sensors. The four goals for the study of their lockscreen application were user perception (annoyance, satisfaction and security), adoption rate (does reducing explicit authentication prompts make participants start using the app?), who would use it (participants with or without secure unlock method or both?) and in which contexts would it be used (at home, at work, other places, on the move, new places). The field study (n = 20) was divided into three phases with a length of one week for each phase. The recruited users were Phase I was needed for the app to construct an environment profile. In phase II after collecting data the app controlled whether an explicit unlock method was needed dependent on the measured sensor data compared to the stored environment profile from phase I. In phase III, participants had the choice to either continue using Micallef et al.'s app or return to their previous unlock method (PIN, pattern or no lock). The results showed that their app increased the needed explicit unlocks for participants previously without lock for 34% in phase II and 24% in phase III. For those using PIN or pattern before, their lockscreen app could achieve a decrease for the needed prompts of 71% in phase II and 74% in phase III. The reduction showed evident changes in the user perceptions. The “no lock” group was not annoyed through all phases although they had sometimes to prompt PINs or patterns, while the annoyance declined for the “lock” group when they started using the app instead. The security perception was consistent for users who had a secure lockscreen before during all phases and did improve for those who had no lockscreen as they were now only asked in insecure situations to provide credentials. No significant difference was found for convenience, whereas results showed a trend of participants with PIN or pattern finding the reduced unlock prompts more convenient compared to their old mechanism. 18 of 20 users adopted the app in phase III (adoption rate: 85%) in one or more contexts. 17 out of those 18 would use a commercially available version - more than five “no lock” participants would use it at *other places*, *on the move* and *new places*, while all nine “lock” users would favor the app with its reduced locks at *home*. There was only one “lock” participant who was not willing to exchange more convenience against a decreased level of security.

Table 2.1: Perception of annoyance (1=most annoying, 5=least annoying), convenience (1=most convenient, 5=least convenient) and security (1=most secure, 5=least secure) for unlock methods in Micallef et al.'s user study [37] - mean, (median) and standard deviation (sd)

	Annoyance	**Convenience**	**Security**
Password	1.84 (1), sd = 1.21	4.47 (5), sd = 0.9	2 (2), sd = 0.94
PIN	2.47 (2), sd = 1.22	3.84 (4), sd = 1.01	2.17 (2), sd = 0.9
Pattern	2.31 (2), sd = 0.82	3.58 (4), sd = 1.07	2.9 (3), sd = 1.1
No Lock	4.42 (5), sd = 1.3	1.55 (1), sd = 1.26	5 (5), sd = 0
Micallef et al. lockscreen	3.95 (4), sd = 0.91	2.42 (2), sd = 1.07	2.79 (3), sd = 1.4

Micallef et al.'s work shows there is no significant difference for “no lock“ and “lock” users to adopt to a new method with reduced authentication prompts. In contrast to previous papers [41, 15], 94% of the users have no problem with private attackers and would like reduced explicit authentication at home, but only 53% at work. Table 2.1 shows user ratings of the study for all unlock methods. Compared to the other unlock methods, the lockscreen app of Micallef et al. was rated with the lowest annoyance and the most convenience after no lock, but was perceived as secure as pattern after PIN and password. This shows, their app was accepted of both groups regardless of the preferred unlock method. Nevertheless, the adoption rate was 85% for one or more contexts, i.e. the alternative was not preferred in all contexts instead of the old unlock method. This proves again that there is no one-for-all solution.

SnapApp uses fractions of many concepts, implementations and ideas from the previously presented papers. Together they are added to a new strategy, which has not been used before. SnapApp is not an invention of a new unlock method, but a combination of the two existing methods PIN and slide-to-unlock in one lockscreen. PIN offers as usual unlimited access to the smartphone, whereas slide-to-unlock restricts it to a short access within a user-defined timeframe. Proven by Harbach et al., smartphones are frequently used with short session times, mostly access insensitive data (74.7%) and need on average 2.9% of the total usage time to be unlocked, which is converted over one hour per month [21]. Instead of applying implicit authentication like [22, 48, 38, 37], SnapApp relies on the user to chose short access when needed to reduce explicit authentication prompts and to save time spent for unlocking the device. Studies have shown that many people are rejecting a secure unlock method mainly due to inconvenience or using a secure one while being often annoyed by it [21, 15, 37]. PIN is the fastest secure standard unlock method [60] and slide-to-unlock is the most convenient and at least annoying one [37], which is why both have been chosen for SnapApp. For security reasons, the short access by slide-to-unlock is not only time-constrained, but also limited by a user-defined blacklist of apps, by which SnapApp becomes an application-centric approach. Against all-or-nothing, SnapApp provides with its short and unlimited access a dynamic configurable model comparable to Stanjano's public and private hats [52].

3 SnapApp

The main task of this thesis was the development and implementation of an application on the Android platform. The app should replace the Android default lockscreen, enable time-constrained access control and collect usage data during a field study. This section explains the features of SnapApp and the design decisions which led to the final application.

3.1 Concept overview

SnapApp provides two unlock methods in one lockscreen (*see figure 3.1*). While the first method PIN requires the user to prompt credentials to get full access, the second method only needs a fast slide-to-unlock gesture before the smartphone is unlocked and can be used for a limited amount of time. SnapApp replaces the Android system lockscreen by overlaying it and can be installed as an app on any regular smartphone running Android OS between the versions 4.1 (Jelly Bean) and 5.X (Lollipop). The SnapApp lockscreen pops up every time the smartphone is turned on or when it is locked again by an event while the screen is on. The short access by slide-to-unlock is called a *Snap* and limits the access to a certain preset amount of time (alias *Snap time*). Apps can be optionally excluded from the usage during a *Snap* by a user-defined blacklist. Apart from that, the lockscreen is built from the example of ordinary lockscreens and shows the current time, weekday and date and can be personalized with a custom background image (*see figure 3.3*). The system notification bar at the top is not occluded by the SnapApp lockscreen as it shows notifications, connectivity status, battery life and other valid informations. In previous studies, different unlock concepts have successfully reduced explicit authentication by using implicit methods [22, 48, 38, 37]. Due to the reduction, users who had rejected a secure unlock method before were accepting it then as it was only shown when necessary. On the other hand, the group in favor of security appreciated having more convenience by less active unlocks. With the trade-off between convenience and security, SnapApp aims for the same effect of the reduction by introducing *Snaps*. The assumption is that a *Snap* is more convenient, faster and thus time-saving than prompting the PIN when it is used for the various short sessions during the day as determined by Harbach et al. [21].

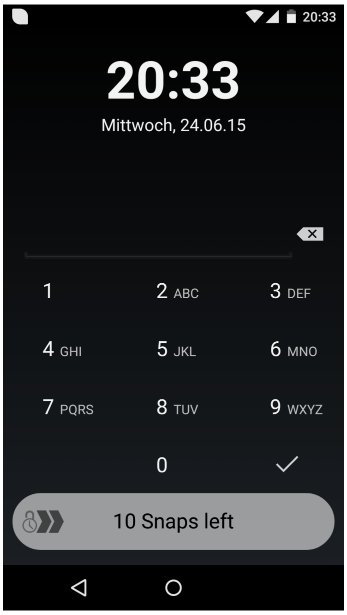


Figure 3.1: SnapApp lockscreen with PIN and 10 Snaps (slide-to-unlock) left

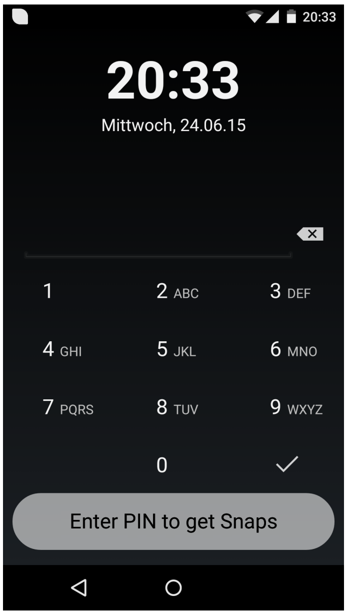


Figure 3.2: SnapApp lockscreen with PIN and no Snaps left

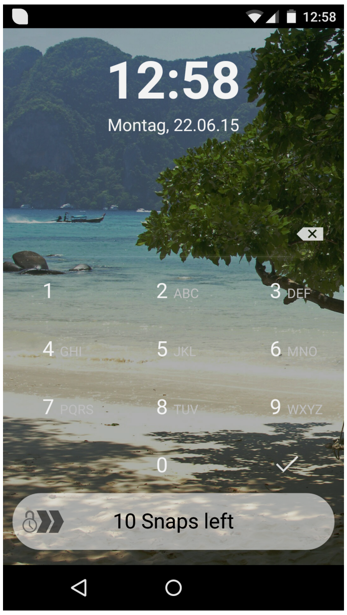


Figure 3.3: Personalized SnapApp lockscreen with a custom background image

3.2 Snap

A *Snap* starts when the smartphone is unlocked by slide-to-unlock and ends after the the *Snap time* has elapsed. In the field study of Harbach et al. an user session (from screen on to screen off) lasted on average 70.3 seconds and 104.1 seconds for the sessions where the device was actually unlocked and the home screen was viewed by the user, too [21]. The default *Snap time* is set to half or respectively to one third of these values with 35 seconds, as the measured averages included both short and long user sessions. SnapApp provides a haptic warning by a threefold vibration alarm of the phone five seconds before the *Snap time* is up. Thereby, the intention is to prepare the device holder to the upcoming end of the *Snap* without the distraction of a visual notification which might disturb the user's workflow. At the end of the *Snap*, the lockscreen pops up and the user can again decide between full access and short access.

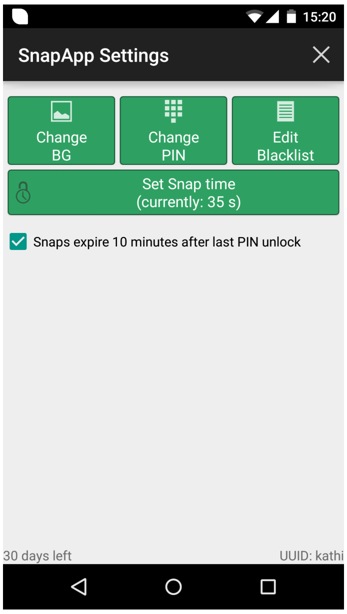
Until now, the user would have full access to the phone during a *Snap* - without any authentication. As this functionality alone would clearly be a big security vulnerability, SnapApp has integrated multiple security features on this reason. As shown in figure 3.1, there is a *Snap* counter which can reach a maximum of ten *Snaps*. Each time the user unlocks the phone by a *Snap*, the counter gets reduced by one until it hits zero. After that, there is no alternative to unlock the phone except by PIN (*see figure 3.2*). A successful PIN entry "recharges" the *Snap* counter to its maximum. The counter gets also reset to zero at the system boot-up, when the PIN validation fails, after the default expiration of ten minutes since the last PIN unlock or when an app of the blacklist is opened during a *Snap*.

3.3 Blacklist

The blacklist of SnapApp is the most important security feature as it regulates the app access during *Snaps*. It is a list of all installed apps of the smartphone which can be launched by the user. When an app is active on the blacklist, it is excluded from the usage during a *Snap*, i.e. the device gets instantly locked and the remaining *Snaps* expire immediately if a user tries to launch a blacklisted app anyway. All applications in the list are inactive per default and can be separately activated by the user tapping once on the desired app. Tapping a second time on the same list item deactivates the according app again. There are also the two options to select or deselect all list items at once in the action bar overflow menu (*see figure 3.6*). Hence, the blacklist can be also used as a whitelist by selecting all apps first and then deactivate the particular desired apps which may be used during *Snaps*. The SnapApp settings and blacklist and the Android system settings are permanently activated on the blacklist, not shown in the user-editable list and can as a consequence not be deactivated or be removed. That measure is a security precaution, because SnapApp needs device administrator rights to work properly which can be withdrawn in the Android system settings. The SnapApp settings and blacklist are also safety-critical as the deactivation of apps on the blacklist or SnapApp specific preference changes in the settings could lead to misuse of the unlock method.

3.4 Settings

The settings panel provides several individual adjustments for SnapApp. Users can choose an individual image from their own photo gallery, which is then set as background image of the SnapApp lockscreen for personalization (*see figure 3.3*). Changing the PIN requires the user to provide the old one and prompt a new one twice (*see figure 3.5*). When the old PIN is prompted wrongly three times in a row, the device gets completely locked to prevent potential attackers from editing other settings. The expiration of the maximal ten *Snaps* comes per default ten minutes after the last successful PIN unlock and can be disabled to never expire, too. Changing the standard *Snap time* of 35 seconds to another time value is allowed as long as the new one equals or exceeds ten seconds. The *Snap time* picker dialog for it has no predefined time steps or limits to avoid



15:20

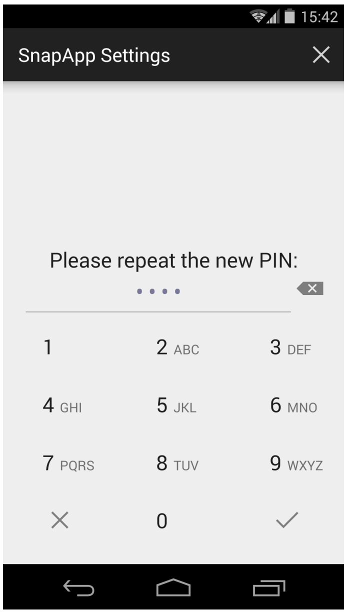
SnapApp Settings

☑ Snaps expire 10 minutes after last PIN unlock

30 days left

UUID: kathi

Figure 3.4: SnapApp settings screen



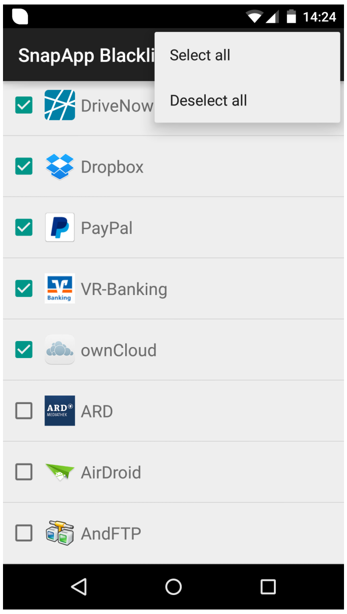
15:42

SnapApp Settings

Please repeat the new PIN:

••••

Figure 3.5: SnapApp settings: providing a new PIN



14:24

SnapApp Blackli

☑ DriveNow

☑ Dropbox

☑ PayPal

☑ VR-Banking

☑ ownCloud

□ ARD

□ AirDroid

□ AndFTP

Figure 3.6: SnapApp settings: editing the blacklist

biased adjustments by the user. It takes any provided value in hours, minutes and seconds. Editing the blacklist is only possible by using shortcut to it, which is also placed on the settings panel. Additionally, participants of the SnapApp field study can check the amount of the remaining days until the end of the study and their anonymous unique user ID at the bottom of the screen.

3.5 Concept development

The requirements for SnapApp were the replacement of the Android lockscreen, enabling time-constrained access control and logging the usage data for further analysis. Apart from that, there were no further specifications. The lockscreen of Android can only be replaced by having root access to the phone. On normal commercial smartphones, which should be used during the field study, root access is denied to the user by default and cannot be obtained without severe changes in the operating system. Additionally, lockscreens are specific on each device, dependent on the manufacturer, the device model and the Android version. On these grounds the software was implemented as an installable app instead.

Security was a big concern right from the beginning. There have been other prototypes of alternative lockscreen apps, which were fullscreen applications but could be circumvented by the home button or the button for displaying recently used apps [17]. SnapApp works as an overlay on top of the system lockscreen. When SnapApp gets installed, it requests the permissions and device administrator rights to disable the system lockscreen, change the screen-unlock password and lock the screen. Additionally, the user gets prompted to provide a PIN which is stored inside the app (*see figure 3.5*). Thereby, SnapApp can disable the system lockscreen in the background by setting the system password to blank and unlock the device before it hides the overlaying SnapApp lockscreen. In the best case, the user does not even notice the system lockscreen behind the SnapApp lockscreen. After the user is done with all interactions, it reacts on the screen off event and sets the system password to the user's stored PIN. When the SnapApp lockscreen is now dismissed by the back button, the home button or any other way, the device is still locked and the system lockscreen asks for the same PIN as SnapApp.

The layout of the first app draft (*see figure 3.7*) dismissed the action bar (title of the app), was not in fullscreen and kept the status bar at the top. This bar provides useful system informations to the user, e.g. battery level, status of wifi and mobile data connection, current sound profile and other notifications. Although being a prototype, the app should look very close to a conventional lockscreen. The SnapApp lockscreen shows the current date, day of the week and time formatted by the user's region settings of the phone. A pin-pad and a *Snap* button below were placed at the bottom. The *Snap* button should be easy accessible at all times, especially when the phone is used single-handed.

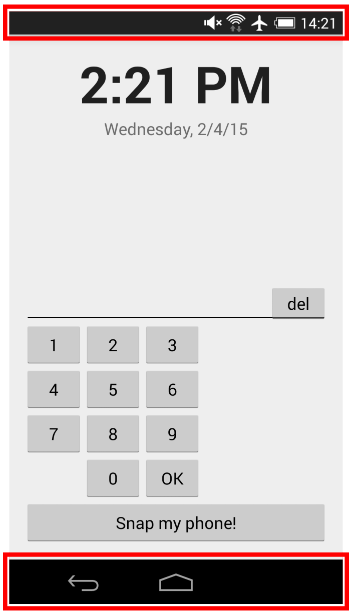


Figure 3.7: The first SnapApp lockscreen draft with shown status bar could be dismissed by the back or home button.

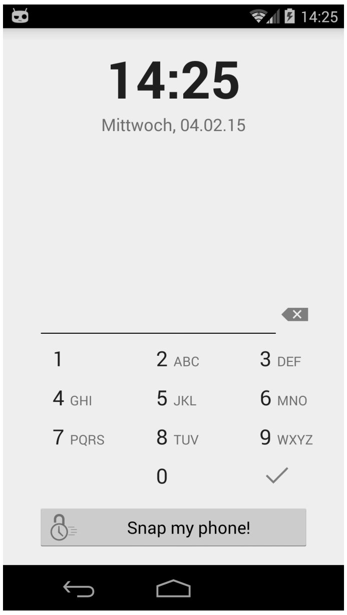


Figure 3.8: With the second SnapApp draft, the pin-pad was changed to alphanumerical and an icon was added to the *Snap* button.

With the second app draft (*see figure 3.8*), the pin-pad got a facelift by alphanumerical buttons. Some people memorize PINs as words [53] which is why the according characters should be displayed next to the digits. An icon showing a moving "lock-clock" was added to the *Snap* button to illustrate the functionality behind it.

Tests exposed that the *Snap* button was often accidentally pressed while hitting the zero or the confirm button of the pin-pad above. Consequently, the smartphone started a time-constrained session although an unlimited one was wanted. A first solution using a bigger push time threshold for the button was dropped as the unlock process felt artificially long. Using slide-to-unlock instead solved the problem of unwanted pushes in another way. The movable icon has to be dragged completely from the left side to the other end of the screen before it triggers a *Snap*. As a result, misplaced touches on the slide-to-unlock bar are ignored. The duration of one slide-to-unlock is approximately the same as one long button press, however, due to the movement of the gesture it feels faster.

Another issue regarding security was caused by the *Snap* concept itself. A hypothetical intruder would have had access to the phone for a short amount of time. After that, the phone would have locked itself and displayed the SnapApp lockscreen again. The invader could just indefinitely repeat the *Snap* unlock and do anything. A regulating feature had to be included. Therefore, the *Snap* counter was introduced, limiting the *Snaps* in a row to a maximum of ten and displayed within the slide-to-unlock bar (*see figure 3.1*). After all *Snaps* are used up, users can only unlock the smartphone by PIN, which also tops up the available *Snaps* back to ten. Extra security was established by letting all available *Snaps* automatically expire after ten minutes since the last PIN unlock. Hayashi et al. found out users want to have 50% of their apps accessible without unlocking the smartphone [23]. On the other hand, this means 50% should be excluded from the easy access or adapted to SnapApp from the short access by a *Snap*. Studies have shown very diverging user preferences in changing contexts of which apps are sensitive and which are not [23, 37]. The easiest way to meet all needs was a blacklist concept. The Android package manager API allows the extraction of a list with all apps which can be manually launched by the user. Implemented and displayed as editable list within SnapApp, the user can activate and deactivate apps with the according list item checkbox (*see figure 3.6*).

3.6 Lifecycle

SnapApp is implemented as an application with an always running background service which monitors and controls every state (*see figure 3.9*). The service utilizes receivers to react on several Android system events like system boot-up and shut-down, screen activations (on and off), user presence (device is unlocked and in use) and call state changes (idle, ringing, active). At the system boot-up, the SnapApp service is started, resets the *Snap* counter to zero, locks the device and shows the SnapApp lockscreen. If the service registers an initiated system shut-down, it sets the stored user password before the device is powered off. When the smartphone is now switched on again, the system lockscreen demands the user PIN until SnapApp is started and overlays it with its own lockscreen.

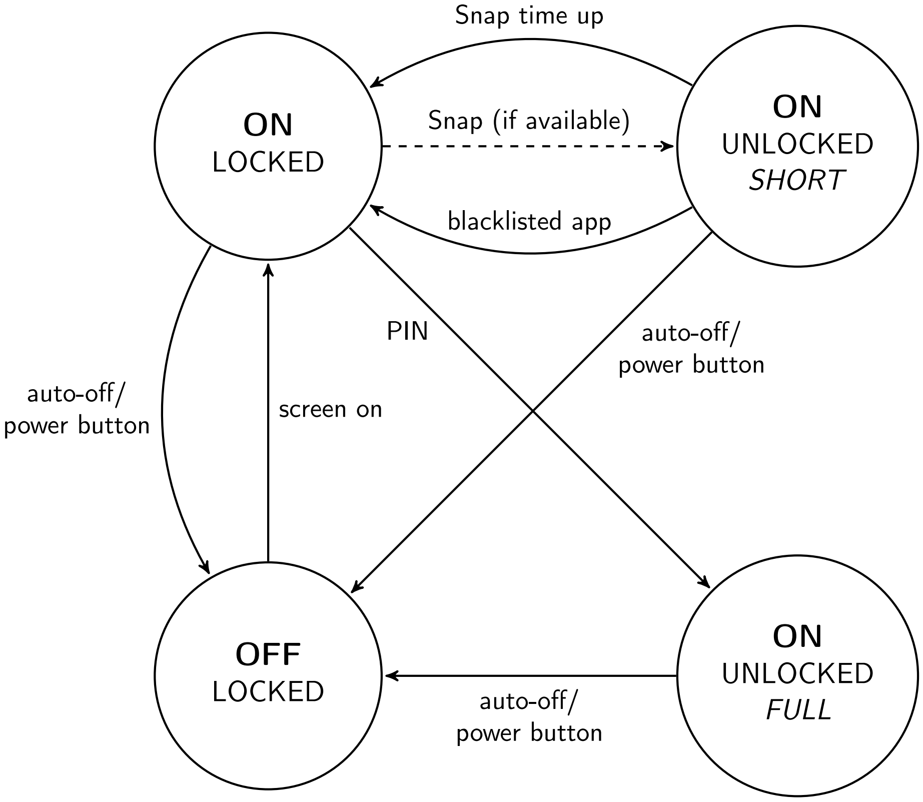


Figure 3.9: Simplified SnapApp lifecycle with device states and transitions

The screen on event can be manually triggered by the user or automatically by an event like an incoming call, a new notification or an alarm. Alarms and calls overlay the system lockscreen, but they would be occluded by the SnapApp lockscreen, which has already happened to another lockscreen prototype [17]. When the device is locked and there is a screen on event (state: ON LOCKED), the service repeatedly checks every 300ms for active calls or alarms and shows the SnapApp lockscreen only if none have been found. The user can then decide between full unlock by prompting the PIN (state: ON UNLOCKED FULL) and *Snap* by using slide-to-unlock if available (state: ON UNLOCKED SHORT). During a *Snap*, the service monitors the usage with a sampling rate of 1Hz. Simply put, SnapApp checks every 1000ms if the currently used app is active on the blacklist until the *Snap time* is up and the device is locked again (return to state: ON LOCKED). A blacklist violation causes an instant lock with immediate expiration of all remaining *Snaps* before the *Snap time* has ended. Both, full and short unlock can be ended by biding the screen off timeout or pressing the power button. The SnapApp service registers the screen off event, sets the user password and locks the device (state: OFF LOCKED).

4 Field study

A user study was conducted to gain insights about the unlock and the usage behavior of people using SnapApp. As a lockscreen plays a central role in the everyday smartphone usage in different situations and environments, the user study was carried out in the field by installing and testing the SnapApp prototype on users' regular commercial smartphones.

4.1 Design

The user study was designed to evaluate the general acceptance of a time-constrained access control concept implemented in form of the previously presented SnapApp prototype (*see section 3 on page 13*). For a broad acceptance and usage of a new unlock concept, the most important requirements are a technical bug-free implementation and a control concept which is simple and convenient to use. The main task for the participants of the field study was the usage of SnapApp as lockscreen replacement on their own phone for 30 days in a row. For evaluation, qualitative data was aggregated by two big online questionnaires at the beginning and at the end of the 30 days period and also by multiple in-situ mini questionnaires on the phone during the field study. Additionally, quantitative data about the SnapApp usage was continuously recorded throughout the field study by extending the prototype with log mechanisms. Used dependent variables were the user-defined *Snap time*, the *Snap* expiration setting, session count and various lengths, app categories and usage times, the chosen unlock method per session, the blacklist usage and the feedback from the questionnaires.

The field study was focussed on the following research questions:

(a) How well was the time-constrained alternative unlock method accepted compared to the PIN?

(b) Did the users adapt the *Snap time* and the *Snap* expiration to their needs?

(c) Were apps used differently during *Snaps* and full access?

(d) Was the blacklist feature used?

(e) Do the user-defined settings and configurations mirror their privacy and security concerns?

4.2 Apparatus

As previously presented (*see section 3*), the SnapApp prototype enabling time-constrained access control on a smartphone by *Snaps* and full access by PIN as alternative unlock method was used for the longitudinal field study. The prototype was implemented for Android 4.1 or higher and was delivered as an Android application package file (APK file). The APK file was distributed on the SnapApp microsite and had to be downloaded by the participant to the own Android smartphone first, before it was installed.

For the data aggregation of the unlock behavior and the smartphone usage in the field study, the SnapApp prototype needed to be adapted. The logging was realized by a similar approach as [7, 21], which also used the Android OS APIs to collect data. With its various events, the SnapApp lifecycle (*see section 3.6*) allowed an easy integration of simple log mechanisms. Logging is activated after the initial setup, which requires the user to grant SnapApp device administrator rights and provide a PIN. All logs are then stored locally on the device into a SQLite3 database, which is automatically transferred to a secure server every 24 hours. The database is divided into the two tables "blacklist" and "logs".

Each single added and removed app of the blacklist gets a log entry into the "blacklist" table. As soon as the user is done editing the blacklist, an overview entry with a concatenated list of all active blacklist items is saved. Additionally, the overview log entry has been triggered automatically every 12 hours by the background service since a SnapApp update during the study. This was necessary as some participants had edited their blacklist before they have finished the setup, by which their blacklist changes were not logged. The update helped to reconstruct the blacklist usage retrospectively. In both database tables, each log entry contains the current weekday, date, time and the current unix timestamp in milliseconds, which allows accurate calculations of session times and other durations afterwards if needed.

The "log" table is additionally supplemented with the current *Snap* counter reading, the custom *Snap time* and the *Snap* expiration status (on or off). Its entries are based on the SnapApp lifecycle (*see figure 3.9*). Each screen on and off event is captured and optionally combined with an extra parameter indicating a boot event before or after ("DEVICE_STARTUP" for the first screen on after the system boot-up and "DEVICE_SHUTDOWN" for the last screen off before system power off). An unlock log entry includes the unlock method (*Snap* or full access) chosen by the user. The triggering event is captured after a successful PIN prompt has been submitted by the confirm button or after a completed slide-to-unlock. The SnapApp service had to be extended for the full unlock phase by using the application checks every 1000ms which were already implemented for *Snaps*. With the repeating checks being available for both unlock methods (logging only for full access), they are used to determine the currently used application. Each newly detected active app is logged with the additional parameters name of the app, its package name and the usage duration in milliseconds. By switching to another application, the previously active app's duration is calculated and updated in the database. When the display shuts off due to the screen timeout or the user pressing the power button, two log entries are made. The first one is the lock entry which marks the end of an unlock session and contains a "SCREEN_OFF" parameter. It is directly followed by the screen off log entry with the same timestamp which ends the current screen session. While full unlock sessions can only be ended by a screen off event, a short unlock session can be terminated in two more ways. Firstly, the service could detect the usage of an app prohibited by the blacklist, which causes an immediate lock with the according log entry flagged with a "BLACKLISTED_APP" parameter. Secondly, the *Snap time* might run up and the device gets locked before the user finishes the current interaction, resulting in a lock log entry with the parameter "TIME_UP". When a lock is caused by the usage of a blacklisted app or wrongly prompted PINs either on the settings panel or on the SnapApp lockscreen, it is called a full lock, is logged with the additional log entry parameter "PIN_FAIL" and causes immediate expiration of the remaining *Snaps*. Occasionally, SnapApp shows mini questionnaires (*see section 4.7 for more details*) to the user, which also can be skipped. Both the user provided feedback data or the skipping of the feedback are logged into the "logs" database table. User changes of every SnapApp settings parameter are exceptional loggings, as they occur sporadically and are not part of the SnapApp lifecycle. The logged changing events are editing the blacklist, the *Snap time* or the PIN. Additionally, toggling the ten minute automatic *Snap* expiration to on or off is also recorded as a log entry.

4.3 Procedure

The study realization was entirely online from the recruitment to the closing questionnaire. At first, anyone interested had to complete a preliminary short survey. Based on the results, users were categorized and selected by the Android version of their smartphone and their used unlock method. A random chosen set of participants using PIN, swipe or no unlock method and having a compatible smartphone were invited by e-mail to take part in the paid user study. A part of the users was informed that their smartphone did not meet the minimum requirements for the SnapApp prototype. The others received a message that they have not been chosen for the paid study and were offered to participate voluntarily.

The e-mail for the chosen users and the volunteers provided a link to a microsite, which contained the download link to the SnapApp application and the briefing. The participants were advised to install the app on their smartphone, read the frequently asked questions to understand the functionalities of SnapApp and complete a consent form and the first of two questionnaires. The pre-study questionnaire collected demographic data about the user, informations about the smartphone itself and the usage by the participant, ratings of the previously used unlock method and the user perception of risks and privacy.

In the following 30 days of the field study, two different in-situ mini questionnaires popped up occasionally while using SnapApp. The first questionnaire showed up right after an unlock and asked the user about the current whereabouts. Additionally, the criticality of the current environment and the sensitivity of the data which would be accessed had to be rated. The second mini questionnaire occurred after the *Snap time* ran up during a short access and questioned the user how much longer the time span should have been.

After the end of the field study, the participants were requested by e-mail to complete another questionnaire to finish the whole user study. The after-study questionnaire started with the smartphone usage, ratings of SnapApp as used lockscreen and *Snaps* as alternative unlock method. Again, privacy and risk perceptions were gathered like before do detect any changes due to SnapApp. Finally, specific questions about the usage of SnapApp, problems with it and its features were raised before the user could finish with optional praise, criticism and open comments.

4.4 Participants

The devices of the participants had to match several requirements. With SnapApp being implemented as an Android application, interested parties needed to have a smartphone with Android version 4.1 or above. Furthermore, the used unlock method was crucial. SnapApp utilizes PIN and slide-to-unlock for Snaps which is why people who were already familiar with using one of both unlock methods were preferred. This was based on the assumption that an extensive amount of pattern or password users could have a distorting impact on the study results.

A message was sent to the mailing list (*see appendix A*) for scientific studies of the University of Munich with 5133 recipients. Additionally, online links with descriptions were placed on social media platforms. The title of the recruitment message was “Test a novel unlock concept for Android”. It further asked whether people would be annoyed of permanently unlocking their phones even for short interactions. The message stated that there might be a solution for that problem which could be tested on the own smartphone during a period of four weeks. Neither the functionality of SnapApp was explained nor was the desired target user group mentioned in order to keep as many interested as possible. If they wanted to give it a try and get a 20 € Amazon voucher for their participation, they just should follow the link to a preliminary short online survey. Students of the Media Informatics department of the University of Munich could choose between the Amazon voucher and study participation points for their studies.

The preliminary short survey was completed by 240 users, of whom 45 were dismissed due to their smartphones with incompatible Android versions. From the 195 remaining participants, 54 PIN and swipe users were approved to the paid user study while the remaining 141 were invited to take part voluntarily. Altogether, 101 participants used pattern, 38 PIN and four password, while 52 users were using swipe or no unlock method at all. That is, 26.6% of all participants were rejecting to use a secure unlock method. Both, the installation of SnapApp and the completion of the pre-study questionnaire was accomplished by 50 users, of whom 40 were paid and ten volunteered. This results in a drop-out rate of 74.35% from the initial interest to the actual study. During the 30 days runtime, 21 participants left the field study by uninstalling SnapApp from their device, which means the drop-out rate was at 42.0%. Both, the SnapApp usage for the full study duration and the after-study questionnaire was completed by 29 participants. After a log file analysis, eleven participants had to be dismissed leading to the final amount of 18 users with valid datasets. The average age of those 18 participants was 24 years (median = 21, range 19 -

64). Seven users were female, while eleven were male. 72.2% of them used with PIN or pattern a secure unlock method, 27.8% preferred an insecure one with swipe. An overview of the statistical data at all study stages can be found at table 4.1.

Table 4.1: Number of all, paid and voluntary participants, drop-out rates and distributions of the usual used unlock method for each stage of the study

	users	**paid / voluntary**	**drop-out rate**	**usual unlock method**
preliminary short survey	195	54 / 141	74.4%	PIN (38), pattern (101), password (4), swipe / none (52)
pre-study questionnaire	50	40 / 10	42.0%	PIN (25), pattern (15), swipe / none (10)
after-study questionnaire	29	26 / 3	37.9%	PIN (14), pattern (9), swipe / none (6)
final selection	18	17 / 1		PIN (9), pattern (4), swipe / none (5)

4.5 Preliminary short survey

At the beginning, the short survey asked for standard demographic data, such as age, gender and current occupation (*see appendix B*). After that, participants had to provide the smartphone make and model, the version number of the installed Android operating system and the mainly used unlock method. The amount of usages of the smartphone per day had to be guessed and specified as a number. Finally they had to provide their e-mail address. At the end of the survey, participants were promised to get contacted soon with further information.

Out of 240 submissions, 45 users had to be dismissed as their Android version did not match the minimal system requirements of SnapApp. The remaining 195 were split in 38 PIN, 52 swipe or no unlock method, 101 pattern and four password users. Concentrating on the first two mentioned groups, a list randomizer [35] helped at each group to shuffle it and choose randomly 36 PIN and 18 swipe or no unlock method users.

4.6 Pre-study questionnaire and briefing

After the random choice of the 54 paid participants, the rest of the 195 users were informed about an overflow of interested parties for the paid study. They were encouraged to take part in the study without a fee, but with the incentive having a chance to win one of three 20 € Amazon vouchers in the lottery among all volunteers.

The e-mails to both paid and voluntarily users confirmed their admission to the study and its conditions. Attached as a link, the SnapApp microsite explained the whole study process. Participants were informed that a completion of the study implies two surveys, the installation of SnapApp on their own smartphones and the usage of SnapApp for 30 days, including repeatedly occurring in-app mini questionnaires.

As SnapApp had not been introduced until then, its functionality and features were shortly described. It was noted that SnapApp would log the usage of the phone in several dimensions: unlocks with timestamp and used unlock method, durations between screen on and off, usage period of apps and their names and any changes to the SnapApp blacklist. Users were assured not any other data would be logged (e.g. PIN or phone numbers) and transfers of the collected logs to the server would be anonymous. The microsite listed then the next steps to take.

At first, SnapApp had to be downloaded and installed onto the Android device. Secondly, the pre-study questionnaire had to be completed. Step three advised to read the frequently asked questions on the SnapApp microsite - the users should understand what a Snap is and how it works. It was also noted that the usage of the blacklist feature can improve the security of their data. As last step, users were asked to download a consent form, fill it in and send it back by e-mail or as postal item.

The pre-study questionnaire of step two started with required field “UUID” (Universal Unique Identifier). When SnapApp starts for the first time it requests a new UUID from the SnapApp web server and stores it within the application. The UUIDs for the study were randomly assigned from a list of 1970 generic first names. The names should gain a better readability compared to cryptic computer-generated values and prevent write errors while typing or writing them (e.g. on the consent form or on questionnaires). The questionnaire data and application usage logs of the participants were saved and assigned anonymously only to their UUID throughout the whole study. Due to this, users had to provide some informations twice which had already been asked before, as the preliminary short survey data contained their e-mail addresses and was thus not used for the later data analysis. The pre-study questionnaire was structured as followed (*see appendix C*):

1. **UUID:** Users had to provide their UUID, which could be found at the bottom of the settings screen of SnapApp on their Android device (*see figure 3.4 on page 15*).

2. **Personal data:** Participants were asked again for their age, gender and current occupation.

3. **Smartphone usage:** Providing informations about their smartphones like make, model, installed Android version number and mainly used unlock method. Once again, participants had to guess how often they use their device per day. After that, the questionnaire continued with new questions. Users were asked to rate on Likert scales (5-point Likert scale; 5 = strongly agree, 1= strongly disagree) whether they use their smartphone for short actions (below 1 minute) in the following categories: phone calls, photography, messaging, social networks, games, e-mail, web browsing, news/magazines, maps/navigation, music, videos, journey planner, shopping and others. If *others* was rated differently than *strongly disagree*, a mandatory text field required user input for specification.

4. **Your smartphone:** At first, the current unlock method (before SnapApp) had to be rated (5-point Likert scale; 5 = strongly agree, 1= strongly disagree), whether it provided data security, unlock speed and convenience. Afterwards the same method was applied to rate their worries about two incidents for different groups being involved. The incidents were: being watched during authentication by someone and another person having short access on the phone. Each scene had to be evaluated for the involved other person being a stranger, a colleague, a friend, the partner and a family member.

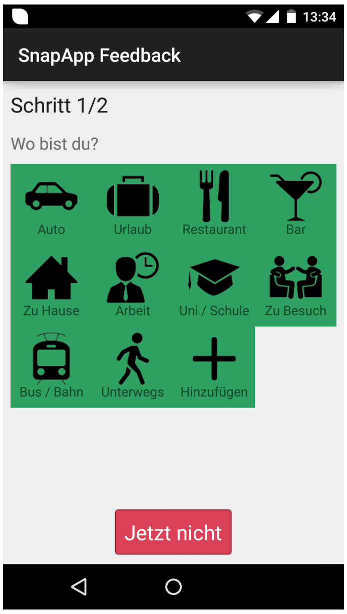
5. **Privacy:** The last part of the survey were privacy questions based on the “dimensions of privacy” table by Taylor [56]. Users had to rate on Likert scales (3-point Likert scale; 3 = extremely important, 2 = somewhat important, 1 = not very/not important at all) the importance of various privacy aspects: having control of who gets information about you, being able to share confidential matters with someone you trust, not having someone watch you or listen to you without your permission, having control of what information is collected about you, not being disturbed at home, being able to be completely alone, not being asked highly personal things in social and work settings, being always identified in public and not being monitored at work.

Both, the SnapApp installation and the completion of the pre-study questionnaire were accomplished by 50 participants, among whom nine were volunteers and 41 were in the paid study (voucher or study participation points).

4.7 In-app mini questionnaires during the study

During the 30 days, users were requested to participate in two repeatedly occurring mini questionnaires right after an unlock or when the *Snap time* ran up. Showing questionnaires in-situ is called the Experience Sampling Method (ESM) and has been successfully used before in previous studies [21, 10, 24, 26]. Subsequent reporting by the users like the Day Reconstruction Method (DRM) tends to overestimation and forgetfulness [31]. Möller et al. have made similar observations including inaccuracy of the users who saw belated self-reporting during a long term study as a burden, too [39]. Consequently, using ESM was the best solution for SnapApp as the questionnaires refer to the current environment or to the usage of the smartphone in a particular situation.

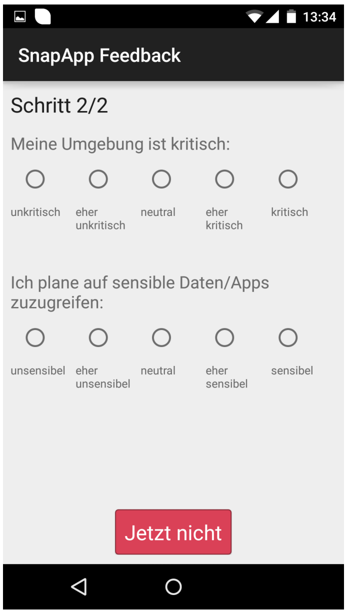
The first questionnaire showed up by a chance of 20% when the device was unlocked in either of both ways. It asked about the current environment and the sensitivity of the data which would be accessed. On the first screen of the two-step feedback the device holder had to classify the own whereabouts. Besides the choice of ten predefined common locations illustrated with matching icons, users could also add their own custom set of places (*see figure 4.1*). On the following second screen participants could rate the criticality of their environment (5-point Likert scale; 5 = critical, 1= uncritical) and the sensitivity of the data (5-point Likert scale; 5 = sensitive, 1= insensitive) which they were going to access (*see figure 4.2*).



Schritt 1/2

Wo bist du?

Figure 4.1: Unlock questionnaire: asking about user whereabouts



Schritt 2/2

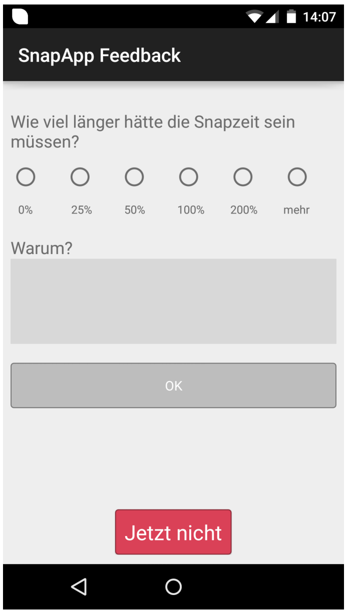
Meine Umgebung ist kritisch:

○ unkritisch ○ eher unkritisch ○ neutral ○ eher kritisch ○ kritisch

Ich plane auf sensible Daten/Apps zuzugreifen:

○ unsensibel ○ eher unsensibel ○ neutral ○ eher sensibel ○ sensibel

Figure 4.2: Unlock questionnaire: rating environment and data



Wie viel länger hätte die Snapzeit sein müssen?

○ 0% ○ 25% ○ 50% ○ 100% ○ 200% ○ mehr

Warum?

Figure 4.3: After *Snap* questionnaire: estimating needed extra time

The second kind of mini questionnaire showed up afterwards when the *Snap time* has run up also by a chance of 20%. Participants could choose whether they needed extra time or not and if yes, how much more. Optionally, users could state why they needed more time in an extra text input field (see figure 4.3).

Both mini questionnaires were each suspended for the duration of one hour before they could pop up again. Combined with the 20% chance for showing up oversampling tried to be avoided.

Besides, users could always bypass the questionnaires by pushing a big red “Not now” button at the bottom of each feedback screen.

4.8 After-study questionnaire

With the automatic upload of the last database file containing the data of all 30 study days, users received an e-mail which asked them to participate in a final questionnaire to complete the study. The mapping of the UUID to the according name and e-mail address was possible by the sent in consent forms. Each participant was allowed to revoke the consent, which results in the deletion of the user’s data and was needed once during the study. Another reason why the mapping was necessary was the delivery of the payments to participants who had finished both surveys and the 30 days SnapApp usage. The payments in form of Amazon vouchers were also sent by e-mail. Neither the mapping in form of a list nor the consent forms were ever combined or stored with actual logged SnapApp data or questionnaire data on any server.

The after-study questionnaire partly repeated the same questions from the pre-study questionnaire to compare before and after. New added questions targeted the participant’s perception, experience and opinion of SnapApp during the study. The questionnaire contained the following questions (*see appendix D*):

1. **UUID:** The users were asked to provide their UUID from the settings screen (*see figure 3.4 on page 15*) again to enable a clear assignment to the their pre-study questionnaire and their SnapApp log database file.

2. **Smartphone usage:** Participants were asked again to guess their phone usages per day and rate what apps they use now for durations below one minute. The categories which had to be rated on Likert scales (5-point Likert scale; 5 = strongly agree, 1= strongly disagree) were the same as in the pre-study: phone calls, photography, messaging, social networks, games, e-mail, web browsing, news/magazines, maps/navigation, music, videos, journey planner, shopping and others. As before, the *others* category demanded a mandatory user input for any other rating than *strongly disagree*.

3. **Your smartphone:** SnapApp had to be rated for its provision of data security, unlock speed and convenience (5-point Likert scale; 5 = strongly agree, 1 = strongly disagree) instead of the previously used unlock method from the pre-study. Afterwards, the incident ratings from the pre-study were repeated, but this time with SnapApp enabling the short access. For each of the two incidents - being watched during authentication by someone and another person having short access to the smartphone enabled by SnapApp - users had to rate their concerns for the person being a stranger, a colleague, a friend, the partner or a family member.

4. **Privacy:** The users were asked privacy questions for a second time based again on the “dimensions of privacy” table by Taylor [56]. The rating on Likert scales (3-point Likert scale; 3 = extremely important, 2 = somewhat important, 1 = not very/not important at all) was about the importance of various privacy aspects: having control of who gets information about you, being able to share confidential matters with someone you trust, not having someone watch you or listen to you without your permission, having control of what information is collected about you, not being disturbed at home, being able to be completely alone, not being asked highly personal things in social and work settings, being always identified in public and not being monitored at work. The reasons for asking privacy questions twice was testing whether SnapApp had any influence on the users’ attitudes.

5. **SnapApp:** The last big part of the questionnaire concentrated on SnapApp only. At first, participants had to answer yes or no whether they had to face the two main problems which could occur in the SnapApp prototype. One problem was the system PIN prompt showing

up on some devices after the unlock with SnapApp, which was an unfixable Android bug and referenced as double PIN problem. The other one caused unexpected locks by SnapApp during the usage and could be fixed in the first half of the study by the deployment of an update. These crucial questions determined whether the user's experience of SnapApp was impaired by unintended malfunctions.

Apart from that, participants had to rate on Likert scales (5-point Likert scale; 5 = strongly agree, 1= strongly disagree) how multiple statements fit to their smartphone usage during the study. They were asked whether they lock their phone before they put it away to assess how correct the measured session times are. Further, the questionnaire tried to figure out the impact of *Snaps* - whether SnapApp saved time for the user, if *Snaps* were stressful to use, if another *Snap* was used when they could not finish their action in the first one and whether they were annoyed when the *Snap time* ran up before they could complete their task. They also had to rate whether the vibration-alarm warning was helpful to successfully complete their current task and if using a *Snap* was easier and faster than PIN. To rule out PIN usage by force of habit, users were asked if they have sometimes thought "I could have used a *Snap* for that, too.". The next ratings questioned if the in-situ mini questionnaires had influenced their user behavior or if they were annoying to find out if the collected data reflects the participants' views. The last two statements to be rated were whether the blacklist feature was reasonable and whether they would use SnapApp without the blacklist feature, too, for a better understanding of their security perceptions.

Finally, after 30 days of usage, participants had to choose what to do now between uninstalling SnapApp and return to their old unlock method or keep on using SnapApp without data logging. In the case of the uninstalling choice, a free text box demanded a short explanation for their decision.

6. **Praise, criticism and miscellaneous:** To gain as much feedback as possible, users were invited to freely describe in big free text boxes what they had liked about SnapApp and what could be improved. Additionally, another text box was placed in the questionnaire for any other comments.

7. **Interview:** At the end, the participants were asked if they could be contacted for an optional interview. Those who affirmed had to specify their e-mail address after being warned about the dismissal of their anonymity of their data by providing it to the questionnaire.

The whole study with all questionnaires has beed completed by 29 participants, among whom 26 received the promised Amazon vouchers or study participation points. The three remaining volunteers could also be paid as the lottery contained the same amount of Amazon vouchers.

5 Results

This section presents the evaluation of the aggregated data from 19 participants of the user study. Quantitative data was collected by the integrated logging mechanisms of the SnapApp prototype. Additionally, qualitative data was gathered by the two main online questionnaires before and after the field study and by the various mini questionnaires on the smartphones during the 30 days field study.

5.1 Data preprocessing

The SQLite3 database files generated by SnapApp containing the usage logs were transferred to a secure web server every 24 hours. During the field study, intermediate checks exposed that the logs were partially incomplete due to app crashes. This was shown by testing the lock-to-unlock ratio of each file in the "logs" table. The ideal case would have been a 1:1 ratio or 100%, i.e. each unlock would have had a completing lock afterwards. The examinations revealed mostly ratios within the 90%-100% range, but also low cases with 60% and below. An update was distributed to all participants during the field study fixing the crash problem. However, after the end of the study eleven of 29 finished participants had to be dismissed. Despite the update, the lock-to-unlock ratio of their datasets had not improved enough to surpass 90%, which was defined as the minimum requirement for a valid dataset.

The remaining 18 valid datasets were processed by deleting every incomplete unlock session without a finishing lock in the "logs" table to retrieve an unlock-to-lock ratio of 100%. Overall, 247,919 log entries have been recorded of which 12,187 (4.92%) entries were deleted by the cleanup process. The amount of log entries per user ranged from 653 to 33,915 and the deletions from 10 (0.16%) to 6,219 (18.34%). There were two participants where the deletion rate was higher than 3.61%, who were both power users with more than 19,000 log entries each. Due to their extensive usage, uncompleted sessions and their removal in the logged data caused a higher deletion rate.

5.2 Logged data

Depending on the usage behavior, the amount of the collected data by SnapApp differs for each user. To prevent over-representation of power users, the data is aggregated per user first, where appropriate, before the average of all users is built.

5.2.1 Unlock durations

The unlock durations were measured by the time difference of the logged screen on and the unlock event. With no touches being recorded, it is important to know that the unlock durations are worst-case estimations. Beside the durations needed to unlock the phone by slide-to-unlock or by PIN, it also includes the times spent for deciding which unlock method should be used and for viewing the clock or notifications. To be able to compare, the data was separated into groups depending on the chosen unlock method. The unlocks which were taken into account had to occur directly after the screen on event. Unlocks after a expired *Snap* were ignored, as well as those where a mini questionnaire was shown which delayed the unlock event. Outliers within the user datasets have been removed by filtering the times per user which exceeded the user's durations mean plus three times the standard deviation. Additionally, seven participants who used *Snaps* for less than 10% of all unlocks have been dismissed for the measurement (*see table 5.2*).

A two-sided paired t-test showed that the authentication is significantly faster for *Snaps* than for full access by PIN (t(10) = 2.45, p = 0.034). Figure 5.1 shows the distribution of the unlock durations for both unlock methods among the eleven users.

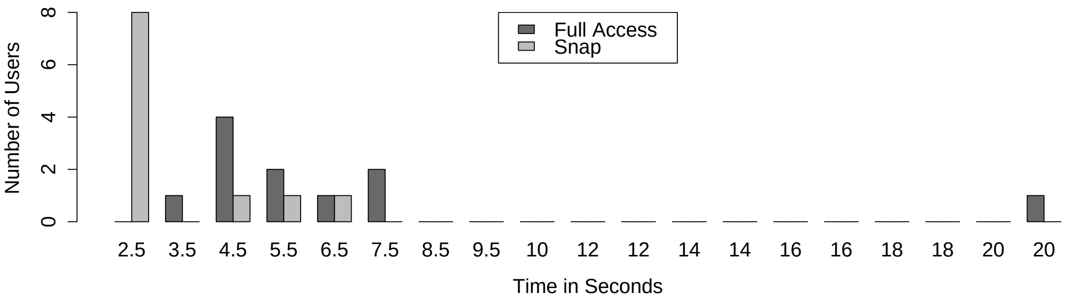


Figure 5.1: Distribution of needed times to unlock the smartphone by PIN (full access) and slide-to-unlock (*Snap*) (n = 11)

On average, 3.46 seconds (sd = 1.45 s, min = 2.53s, max = 6.93s) were needed to unlock the smartphone by *Snap* and 6.71 seconds (sd = 4.82s, min = 3.91s, max = 20.78s) with the PIN approach.

5.2.2 Snap time and expiration

The *Snap time* could be configured by the user in the SnapApp settings panel and was set to 35 seconds per default. Out of 18 participants, four never changed the value, five changed it once, another five twice and three users adjusted it four times (*see table 5.1*).

Table 5.1: Overview of *Snap time* changes during the study (n = 18)

Snap time changes	Number of users
none	4
1	5
2	5
3	4

In total, the participants chose 15 different *Snap time* values, ranging from 10 to 3635 seconds (1 hour 35 seconds). Figure 5.2 shows the distribution of the 15 *Snap time* values among all 4,236 logged *Snap* sessions of 17 users. One user did not use short access at all and provided thus no data. The percentage values were calculated for each user first, before they were converted to total. With the quota of 30.37%, 35 seconds was the most used session *Snap time* as 13 participants started using *Snaps* with the default value, of whom four kept it for the whole duration of the study. The following three best-liked values were adjacent to each other: 120 seconds with 12.30%, 30 seconds with 11.50% and 60 seconds with 10.42%. After that, 7.01% of the participants preferred 300 seconds and 5.23% liked 45 seconds as *Snap time* values. The remaining times did not surpass the 5% threshold and were in descending popularity order: 600s, 150s, 3635s, 155s, 90s, 1200s, 160s, 180s and 10s.

The other *Snap* preference which could be changed by the user was the *Snap expiration*. Per default, available *Snaps* expired automatically after ten minutes since the last full unlock by PIN. In the SnapApp settings panel, users could switch the expiration off and on. Four of 18 users disabled the automatic expiration, whereas the switching off was mostly done within the first half of the study.

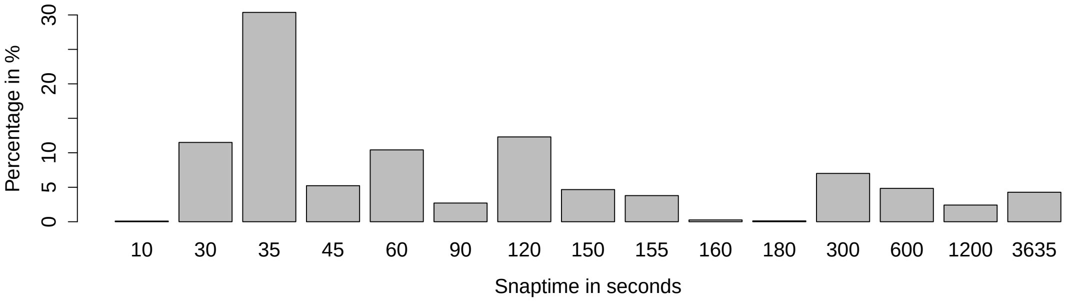


Figure 5.2: Distribution of used *Snap times* during *Snap* sessions (n = 17)

5.2.3 Sessions

During the 30 days field study, 36,520 smartphone activations and 20,701 unlock sessions of 18 users have been logged. That is a mean of 71.31 activations (sd = 34.85, median = 63.57, min = 4.17, max = 148.70) and 40.35 sessions (sd = 20.11, median = 41.27, min = 2.10, max = 83.73) per day and user. With a total of 4,236 registered *Snap* sessions, the new unlock method was used on average with a share of 20.82% (sd = 19.23%, median = 12.46%), ranging from 0.00% to 70.00% per user. Although SnapApp enables multiple sessions in a row during one activation by using *Snaps*, no user had a higher amount of sessions than of activations. That is, the higher number of activations is due to participants viewing notifications or the lockscreen clock without unlocking the device. Table 5.2 presents an overview about the number of full and short sessions, resulting ratios and the normally used unlock method per user. Only three participants who normally use PIN were above the median *Snap* usage level, which indicates the *SnapApp* concept might be better accepted among people who are in favor of the pattern, swipe or no unlock method at all.

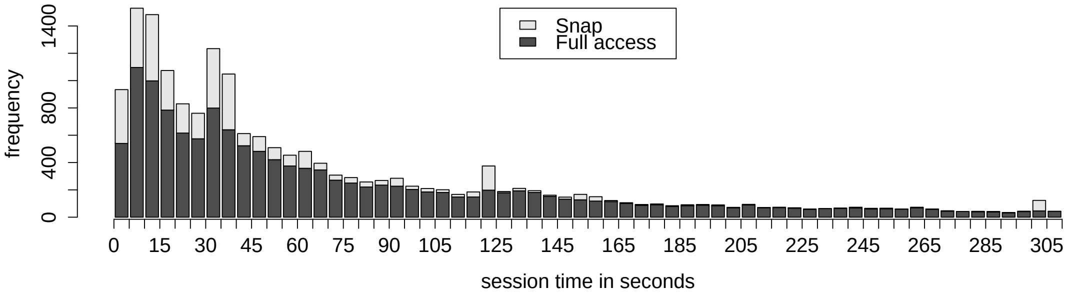


Figure 5.3: Distribution of used unlock method on different session times

On average, a *Snap* session lasted 50.49 seconds (sd = 70.09s, median = 30.48, min = 0.11s, max = 1174.00s), while a full access session endured 198.00 seconds (sd = 429.70s, median = 65.30s, min = 0.62s, max = 9116.00s). Figure 5.3 shows the distribution of both used unlock methods on session durations between 0 and 310 seconds, which includes 88.46% of the user configured *Snap time* values (*see figure 5.2*). The figure shows a high frequency of visible peaks for session times correlating with the most chosen *Snap times*, e.g. at 35s, 60s, 120s and 300s session time. They are caused when the *Snap time* runs up before the user ends the short session manually.

Table 5.2: The number of unlocks with full access and *Snaps*, the resulting used *Snap* ratio, shares of locks caused by blacklist violations or *Snap time* lapse, amount of blacklisted apps and the normally used unlock method per participant (n = 18) and in total. The upper dashed line indicates the average of the total used *Snap* ratios and the lower dashed line the median of it.

participant number	full unlocks	Snaps	Snaps used	Snap time up locks	Blacklist locks	Blacklist items	usual unlock method
#1	447	1,043	70.00%	7.57%	0.77%	0	pattern
#2	751	646	46.24%	31.26%	10.22%	0	PIN
#3	599	389	39.37%	2.06%	5.91%	0	swipe
#4	392	228	36.77%	21.05%	0.44%	0	PIN
#5	41	22	34.92%	27.27%	45.45%	0	pattern
#6	1,010	525	34.20%	14.10%	9.52%	14	swipe
#7	670	296	30.64%	13.51%	2.36%	0	swipe
#8	1,641	250	13.22%	46.40%	2.40%	1	pattern
#9	1,081	157	12.68%	8.28%	12.10%	0	PIN
#10	1,555	217	12.25%	27.65%	2.76%	0	pattern
#11	554	68	10.93%	33.82%	35.29%	2	PIN
#12	1,626	142	8.03%	56.33%	1.41%	0	PIN
#13	1,041	74	6.64%	54.05%	2.70%	0	PIN
#14	468	33	6.59%	6.06%	36.36%	10	PIN
#15	1,237	75	5.72%	33.33%	1.33%	0	PIN
#16	743	43	5.47%	30.23%	44.19%	64	swipe
#17	2,484	28	1.11%	67.86%	0.00%	0	PIN
#18	125	0	0.00%	-	-	0	swipe
Total	**16,465**	**4,236**	**20.82%**	**28.29%**	**12.54%**		

As presented in section 3.6, a *Snap* can be ended in three ways. The presumed ideal case was the user powers off the display by pressing the power button or by biding the screen off timeout before the *Snap time* is up. Otherwise, either the *Snap time* has elapsed and SnapApp locks the phone or a blacklisted app has been used which ends the short session immediately. The regular *Snap* ending occurred on average in 58.76% (sd = 23.24%, median = 58.51%, min = 22.73%, max = 92.03%) of all cases. Blacklist violations happened at least and caused on average 12.54% (sd = 16.43%, median = 27.65%, min = 0.00%, max = 45.45%) of the short session ends. The lapse of the *Snap time* was the most irregular *Snap* termination and appeared with an average share of 28.29% (sd = 19.20%, median = 27.65%, min = 2.06%, max = 67.86%).

5.2.4 Blacklist configurations

The blacklist is the most security feature of SnapApp which always excludes the Android system settings and the SnapApp settings from usage during short access. Although just five of 18 participants had configured their blacklist, 16 faced blacklist unlocks due to the two per default excluded settings apps (*see table 5.2*).

The five users who had configured their blacklists did this in very different ways. Participant #8 had just WhatsApp as messaging application on the list and has changed the *Snap time* twice to 10 seconds at last. Participant #11 changed the *Snap time* once to 45 seconds and excluded the e-mail and the contacts application from *Snap* access. Participant #6 kept the default *Snap time* for the whole study and protected ten apps of different categories, from payment applications to games, by the blacklist. The apps had in common that they were all connected to some kind of user account (e.g. PayPal, Netflix, Runtastic, Quizduell). Participant #14 chose to interpret the

Snaps concept individually by increasing the *Snap time* to 2 hours and 35 seconds. The blacklist contained messaging apps like WhatsApp, Hangouts and Threema, one e-mail client, a firewall app, another settings app and some developer tools which have extended system access privileges. The last blacklist using participant #16 also kept the default *Snap time* and utilized the whitelist approach by adding all available apps to the list first. In the next step, four apps were removed from the blacklist to enable them during *Snaps*: the camera and Chrome as browser app, Adobe Acrobat for viewing PDF documents and Hangouts for messaging. For the participants #11, #14 and #16, their configuration caused evidently more forced *Snap* session ends by a blacklist violation with the shares 35.29%, 36.36% and 44.19% compared to the total average of 12.54%.

5.2.5 Application usage

While the smartphone was unlocked, the currently active app was also logged by its package name and its title. As the chosen unlock method was recorded with the start of each session, all apps could be assigned to be used either during a *Snap* or during full access. In total, 18 users have launched 629 different apps 11,028 times during short sessions and 99,174 times while having full access in die field study. In order to get a better understanding of the data, all apps were categorized. At first, the Play Store category name was assigned to each logged app by crawling the Play Store with the app's package name. The Play Store categories were given to the apps by their developers when they published it. Some applications could not be found by their unique package name in the Play Store[3]. That is, they were either test applications of developers that occurred rarely, default Android apps (e.g. settings or phone), apps pre-installed by the device manufacturers (e.g. clock or keyboard) or apps which were distributed via other channels. In the next step, all apps which were unknown and could not be manually assigned to a category were dismissed. The same was done to applications which were not launched actively by the user, such as launcher apps, background services and UI system apps. From the remaining applications, the top 20 most frequently used apps were accumulated per user. When the next apps in the ordered application list had the same amount of launches like the twentieth, each of them was additionally included until the number of launches decreased again. On average, 20.44 apps (sd = 1.42, min = 18, max = 25) were determined as the most frequently used applications for full access, while 16.88 apps (sd = 6.71, min = 7, max = 27) were selected for short access. The lower number is due to the fact that the participants used less applications during *Snaps*. Next, the usage frequency of all apps per user was converted to percentage shares. Afterwards, the shares were divided by the number of participants to transform them into total shares among all users. The number of participants was 18 for the full and 17 for the short access group, as one user did never unlock by a *Snap* (*see table 5.2*). All apps were then manually assigned to the same custom categories, which were also used for the pre- and the after-study questionnaires (*see section 4.6 and 4.8*). As final step, the total shares of apps within each category were summed up and the categories were sorted by value.

Table 5.3 and 5.4 present the most frequently used app categories separated by unlock method. All categories are self-explanatory except *Other*, which contains all apps which could not be assigned to one of the custom categories. The most used apps of the *Other* category during *Snaps* were the Google App (7.70%), the clock (3.07%), the calendar (1.86%) and the Google Play Store (1.29%). On the contrary, the Google App (9.57%), the system settings (3.80%), the Google Play Store (2.61%), the clock (2.20%) and the calendar (0.84%) were the most widely used *Other* apps while having full access. For both unlock methods, *Messaging* was in the first two most frequently used categories, whereas it was clearly on top during *Snaps* with 34.87% and a gap of 15.55% to the next category. Assigned to *Other*, the Google App was equally utilized for searching and individual informations. After it, the clock and calendar were more often used during short access to view appointments or alarms. In contrast, full access let the participants launch the Android system

[3] The Play Store was crawled on June 6th, 2015

Table 5.3: Categories of used apps during Snap sessions

Category	Percentage
Messaging	34.87%
Other	19.32%
Phone calls	12.76%
Web browsing	7.66%
E-mail	4.51%
Music	4.49%
Games	3.94%
Photography	3.06%
News / magazines	3.04%
Social networks	2.84%
Journey planner	1.27%
Shopping	1.26%
Videos	0.51%
Maps / navigation	0.47%

Table 5.4: Categories of used apps during full unlock sessions

Category	Percentage
Other	26.08%
Messaging	25.88%
Phone calls	11.19%
Web browsing	7.71%
E-mail	7.12%
Games	4.84%
News / magazines	4.36%
Photography	3.59%
Social networks	3.08%
Music	2.03%
Videos	1.70%
Shopping	1.14%
Maps / navigation	0.75%
Journey planner	0.53%

settings in the first place, which were blacklisted during *Snaps*, followed by browsing the Google Play Store. *Phone calls* (*Snap*: 12.76%, full access: 11.19%) and *Web browsing* (*Snap*: 7.66%, full access: 7.71%) were almost used with an identical frequency, *E-mail* apps were less often launched during short access (*Snap*: 4.51%, full access: 7.12%). While *Games* (*Snap*: 3.94%, full access: 4.84%), *Photography* (*Snap*: 3.06%, full access: 3.59%) and *Shopping* (*Snap*: 1.26%, full access: 1.14%) are ranking almost at the same for both unlock session types, the positions of the remaining categories differ within the two ranking lists.

Apparently similar usage frequency can differentiate in the unlock methods within the same category regarding the apps, such as in *Photography*. During *Snaps*, the camera (2.05%) to take photos was more often used than the gallery (1.01%) to view them. When the phone was unlocked by full access, it was the other way round and the gallery (2.01%) was more times launched than the camera (1.59%).

5.3 In-situ mini questionnaires

All participants were prompted to complete two types of in-situ mini questionnaires on their smartphones during the study. In total, 1,455 questionnaires were filled out, while 1,509 were skipped by the user pressing the “Not now” button. 2,841 feedbacks popped up by random after an unlock for all 18 participants, of which 1,369 were not dismissed. On average, each user completed 76.06 questionnaires (sd = 45.66, median = 72.50, min = 3, max = 179) and gave answers about the current whereabouts, the criticality of the environment and the sensitivity of the data which was going to be accessed. On the contrary, 123 questionnaires were displayed to only 15 users after *Snaps* which ended by the lapse of the *Snap time*. With 86 completions, one participant replied on average 4.78 (sd = 6.15, median = 3.5, min = 0, max = 27) feedbacks.

5.3.1 After Snap questionnaires

The questionnaires after *Snaps* asked the users how much longer they would have liked the *Snap time* to be. Table 5.5 gives an overview about the user selections. With a share of 67.44%, the majority of the *Snaps* were voted 36 times as satisfying (0%), 13 times as slightly too short with an desired extension of a quarter *Snap time* length (25%) and 9 times too short with a wanted overtime of 50% more. With a percentage of 32.56%, 28 short sessions ended with the user stating 100% (7 times), 200% (14 times) or even more (7 times) as desired *Snap time* extension. It is important to note that three users were accountable for 19 of these 28 votes (five times 100%, eleven times 200% and three times more). All three participants have not voted for lower values in other feedbacks, which could indicate a too low configured *Snap time* in their cases. The answers of the other users were well balanced. In the optional comment field which asked for the reason of the desired extension, participants mostly stated that they were still typing, waiting for loading request or deciding spontaneously to do more on the phone. The latter brought further reading, watching videos or playing games longer.

Table 5.5: After *Snap* Mini questionnaire answers on the question how much longer the *Snap time* should have been

	user selection					
	0%	**25%**	**50%**	**100%**	**200%**	**more**
frequency	36	13	9	7	14	7

5.3.2 After unlock questionnaires

The two step unlock questionnaire asked the user at first about the current whereabouts. Figure 5.4 shows the distribution of unlocks at different locations for both unlock methods. The place with the most registered unlocks by far is home with 741 occurrences. The next two locations work (149) and uni (143) are close together, as participants spent obviously most of their time there when they are not at home. After that, the most common situation was on the way with 124 unlocks, followed by on a visit (74) and on public transport (59). The remaining locations were not day-to-day places, such as the car (23), other (20), a restaurant (18), on vacation (15) and a bar (3). Locations of the other category were not predefined by SnapApp and included custom places, e.g. the movies or a theater. The shares of short accesses per location varied from 4.35% (car) to 33.33% (bar) with an average of 17.13%.

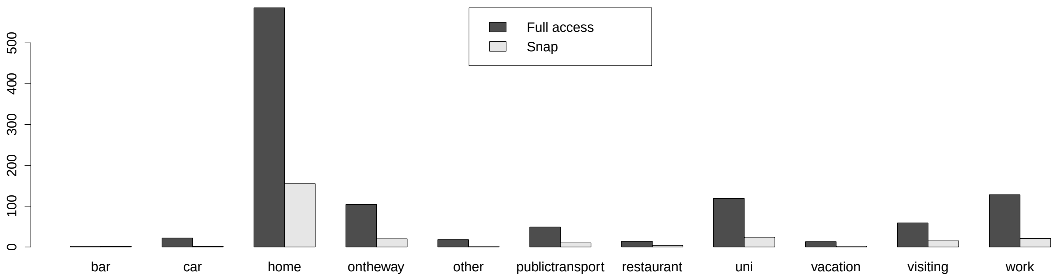


Figure 5.4: Distribution of unlocks by *Snap* or full access in different locations by frequency

The second step of the unlock questionnaire concentrated on the data sensitivity within the upcoming session and the criticality of the current environment. Table 5.6 summarizes all given answers for both the environment and the associated data ratings. The data reveals that in 63.99% of all unlock sessions insensitive to neutral data is accessed within uncritical to neutral environments. Sensitive and rather sensitive data was called up in rather critical to critical areas only with a share of 9.06%, which indicates that participants were probably well aware of their surroundings when they handled sensitive information. That is, the location is only a small factor for data security. The user ratings of the current environment for all unlock sessions in descending order of shares were: 50% uncritical (695), 15.56% rather uncritical (213), 13.73% rather critical (188), 13.15% neutral (180), and 6.79% critical (93). Accordingly, the data sensitivity of each session was also evaluated: 35.14% chose neutral (481), 32.51% insensitive (445), 19.28% rather sensitive (264), 7.82% rather insensitive (107) and 5.25% sensitive (72).

Table 5.6: User ratings of the data sensitivity for the upcoming short accesses mapped on the according ratings of the environment

data sensitivity	**environment rating by user**				
	uncritical	**rather uncritical**	**neutral**	**rather critical**	**critical**
insensitive	389	6	16	7	27
rather insensitive	34	40	16	11	6
neutral	155	108	112	83	23
rather sensitive	78	53	26	79	28
sensitive	39	6	10	8	9

Table 5.7: User ratings of the environment criticality on different locations

location	**environment rating by user**				
	uncritical	**rather uncritical**	**neutral**	**rather critical**	**critical**
bar	0	0	0	3	0
car	7	2	12	2	0
home	552	127	48	12	2
on the way	40	8	29	36	11
public transport	7	2	7	28	15
restaurant	1	5	1	9	2
university	45	12	20	31	35
on vacation	3	0	10	1	1
on a visit	20	28	17	8	1
at work	19	29	31	56	14

Another aspect shown by the data are the participants' diverging perceptions of the criticality of their surroundings. Table 5.7 gives an overview about how critical the participants rated the different places. For better comparison, the following shares of the ratings are summarized to three values by combining the two non-neutral rating counts of each location. The results show that rather private environments were rated more uncritical: home (92% uncritical, 6% neutral, 2% critical), on a visit (65% uncritical, 23% neutral, 12% critical) and in the car (39% uncritical, 52% neutral, 9% critical). Further, there were environments where the rating could be neutral and drift in both directions, such as on vacation (20% uncritical, 67% neutral, 13% critical) or when the user was on the way (39% uncritical, 23% neutral, 38% critical). Interestingly, the users disagreed at the static locations work (32% uncritical, 21% neutral, 47% critical) and at the university (40% uncritical, 14% neutral, 46% critical) where the environment was both perceived as hostile or harmless. Evidently public places were rated as to expect as more critical: at restaurants (33% uncritical, 6% neutral, 61% critical), bars (0% uncritical, 0% neutral, 100% critical) or on public transport (15% uncritical, 12% neutral, 73% critical).

The unlock mini questionnaire results also provide the information which unlock method is used for which data. Table 5.8 reveals that in 54.90% of the *Snap* sessions, insensitive data is clearly the most accessed. While neutral (15.69%), rather sensitive (14.51%) and rather insensitive (12.55%) data is obtained in similar extents, the processing of sensitive (2.35%) data is rare. During full access sessions, the most users rated the to be accessed data as neutral (39.59%). This may be the result of the longer full unlock sessions times, during which more apps can be launched and needed to be rated. For the rest, insensitive (27.38%) and rather sensitive (20.38%) data was called up more. Finally, the participants were interested alike in insensitive (6.73%) and sensitive (5.92%) informations.

Table 5.8: Distribution of access data sensitivity for both unlock methods

data sensitivity	Snaps	full unlock
insensitive	54.90%	27.38%
rather insensitive	12.55%	6.73%
neutral	15.69%	39.59%
rather sensitive	14.51%	20.38%
sensitive	2.35%	5.92%

5.4 Pre- and after-study questionnaires

Beside the SnapApp usage for 30 days, the field study was accompanied by two main questionnaires. The pre-study version was the first step on the to-do list for every participant in the study. It collected general and demographic data, which is presented in the first part of this section. Further, both questionnaires had overlapping contents, as some questions were asked twice - once before and once after using SnapApp. These results are compared further down. Finally, the after-study questionnaire has additionally gathered specific data regarding the experience of the users with the SnapApp prototype on their smartphones during the field study. The evaluation of that data and the free comments of the users complete the results chapter.

5.4.1 Demographics

After the data preprocessing, 18 valid datasets remained. The users were between 19 and 64 years old with a median of 21 years. Seven participants were female, eleven male. The distribution of the installed Android version was split almost half in eight Android 4 and ten Android 5 devices. Out of 18 participants, nine normally used PIN, four pattern and five swipe as unlock method. All major smartphone makes were available with HTC, LG, Samsung, Sony and Google Nexus devices. With the differently aged models, the devices sample is representative. Table 5.9 gives an overview about the previous presented demographics.

5.4.2 Usages per day

All participants have been asked twice to guess how often a day they would use their smartphone. Figure 5.5 shows their guesses mapped with the actual measured activations per day from the logged data. The blue triangles show the pre-study guesses, the green circles the after study ones. Symbol positions above the black diagonal are underestimations by the user, whereas positions below stand for overestimation. In both guesses, most of the user have underestimated their phone usage.

5.4.3 Short usages

The participants had to rate also twice whether they use apps of certain categories for short usages (below one minute) on their smartphone. Both ratings can be compared with each other in the figures 5.6 and 5.7 and with the results of the logged data of section 5.2.5.

Table 5.9: Field study participant demographics

n	18	
age	19	- 64 years, median 21 years
gender	7	female
	11	male
Android version	8	Android 4 - Jelly Bean / KitKat
	10	Android 5 - Lollipop
unlock method	9	PIN
	4	pattern
	5	swipe
devices	3	Samsung Galaxy S4
	2	HTC One, Nexus 5, Samsung Galaxy S4 Active
	1	Fairphone, Huawei, LG G3, Nexus 4, Samsung Galaxy Note 3, Samsung Galaxy S3 Mini, Samsung Galaxy S4 Mini, Sony Xperia Z2, Sony Xperia Z3 Compact

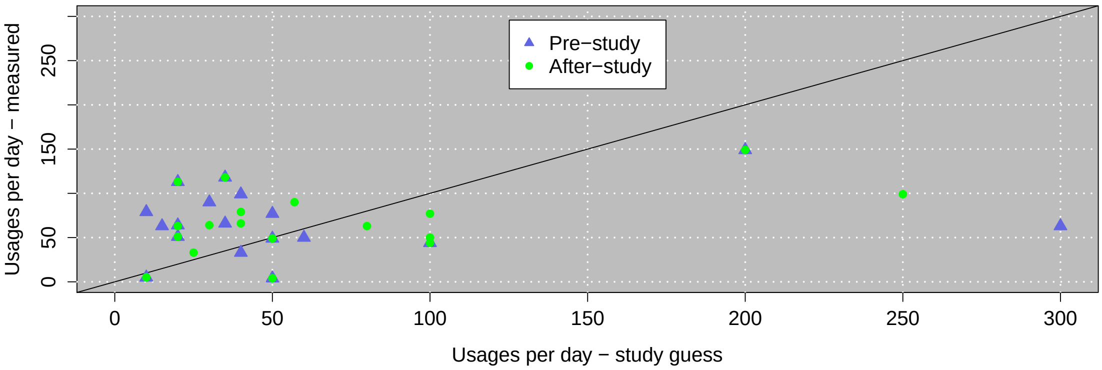


Figure 5.5: Usages per day guessed before and after the field study by participants and measured values from the logs (n = 18)

In the pre-study rating, the participants rated the most used app categories as following in descending order: messaging, journey planner, e-mail, web browsing, maps and navigation, photography, phone calls, news and magazines, music, social networks, shopping, games, videos and at last others.

After the field study, the ratings had changed. The new chronology was still headed by messaging, but e-mail has switched with journey planner to second place. Photography climbed two places up to fourth, pushing web browsing one step down to fifth. After that, the order reads as follows: music, social networks, maps and navigation, phone calls, others, games, news and magazines, videos and shopping.

Table 5.10 shows both rankings next to the results of the logged data. In all three rankings, messaging is on top. E-mail was higher rated than actually used, whereas the estimated web browsing rank was right in the pre- and close in the after-study. Music scored the same ranks in the after-study rating and in the logged data. Consistently, videos was at the seconds to last place in all lists. Against the user ratings, games were used more often, while maps and navigation and journey planners were used much less.

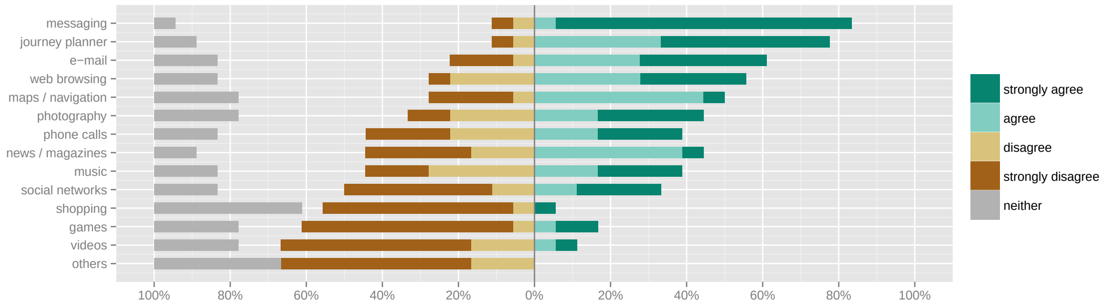


Figure 5.6: Pre-study short usage ratings (n = 18)

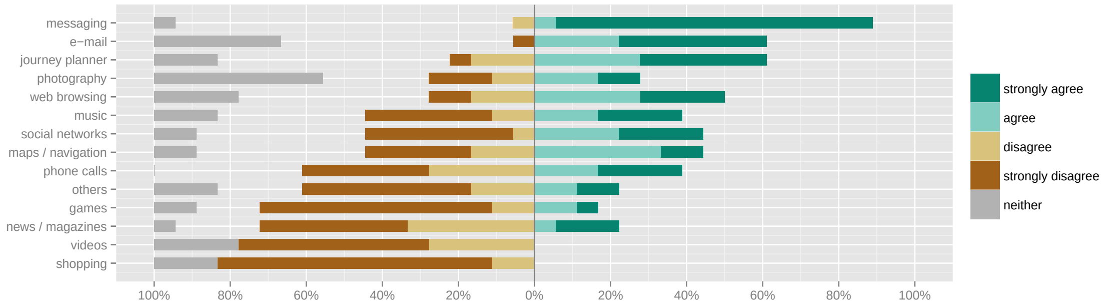


Figure 5.7: After-study short usage ratings (n = 18)

5.4.4 Privacy aspects

The privacy perceptions of the users were captured twice in the pre- and in the after-study questionnaire. Figure 5.8 shows the the participant ratings of the single aspects after the field study under the impression of 30 days of SnapApp. For all users, the ability to share confidential informations with someone they trust was extremely important. The next three top rated aspects were also extremely important to at least 11 participants and somewhat important to the rest: users wanted neither to be monitored at work, nor to be watched or listened to without permission and they desired the control of who was getting their informations. The latter is especially interesting in regard to the previously presented usage of the blacklist, as its configuration controls the availability of the apps to anyone holding the phone. Controlling the kind of information which is collected about themselves was somewhat or extremely important to all users, while being able to be completely alone and not being asked highly personal questions troubled them less. Finally, over 70% of the participants did not mind or somewhat mind when they would be disturbed at home or emphasize to be identified in public.

5.4.5 Security perceptions

To make sure SnapApp is perceived as secure as the preceding used unlock method of the participant, they have been compared in two incidents. The previous used unlock method was rated in the pre-study questionnaire, while it was the turn of SnapApp in the after-study questionnaire. Incident one implied the user was being watched during authentication by five different individuals: by an unknown person, by a colleague, by a friend, by the partner or by a family member. Figure 5.9 shows the ratings of the first incident.

As to expect, being watched by an unknown worries the most, followed be the colleague. The friend lies in between the colleague and the partner regarding concerns. The worries are marginally higher for SnapApp compared to the preceding unlock method, except for the family member, where the difference is more clear out of favor for SnapApp. This might be an indication that

Table 5.10: Short usage app category rankings based on the pre- and after-study questionnaire results and the logged SnapApp data. Matching elements are written in Italics.

pre-study rating	after-study rating	logged data
messaging	*messaging*	*messaging*
journey planner	e-mail	others
e-mail	journey planner	phone calls
web browsing	photography	*web browsing*
maps / navigation	web browsing	e-mail
photography	*music*	*music*
phone calls	social networks	games
news / magazines	maps / navigation	photography
music	phone calls	news / magazines
social networks	others	*social networks*
shopping	games	journey planner
games	news / magazines	shopping
videos	*videos*	*videos*
others	shopping	maps / navigation

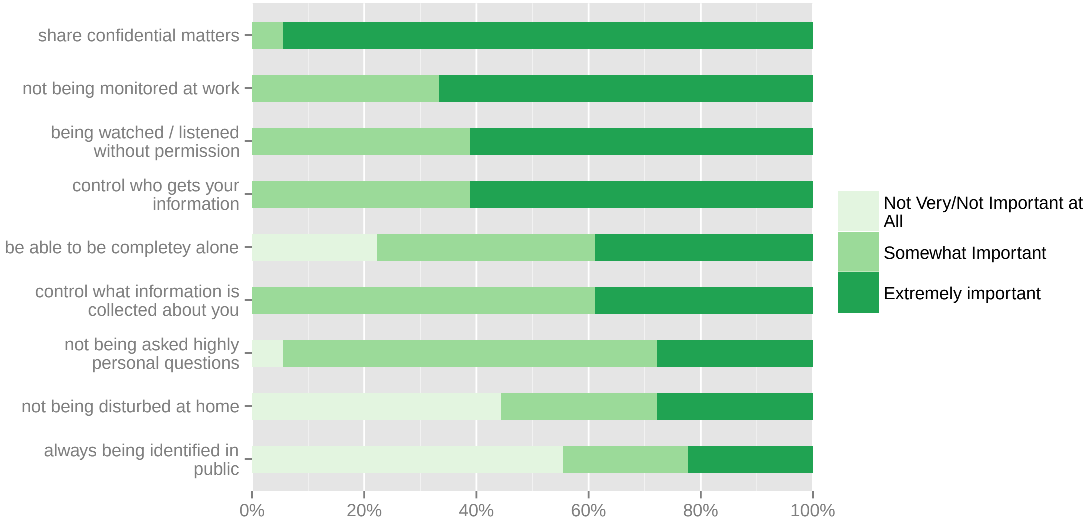


Figure 5.8: User ratings of the importance of different privacy aspects (n = 18)

either PIN and pattern users were worried others could see the possibility to use the smartphone without an explicit authentication, or swipe users feared to be observed when they prompt the PIN for SnapApp.

The second incident was having short access to the phone in the pre-study questionnaire and a *Snap* usage in the after-study one. The persons accessing the device were the same as in incident one. Figure 5.10 presents the results for the short access incident.

Again, the participants were equally worried when an unknown person would have had short access and were more calm when the partner or a family member would have had it. On the other hand, the concerns about a colleague or a friend accessing the smartphone by a *Snap* are greater than about the preceding unlock method. As the majority of the participants normally uses a secure unlock method, SnapApp reduces the security partly for a trade-off with convenience.

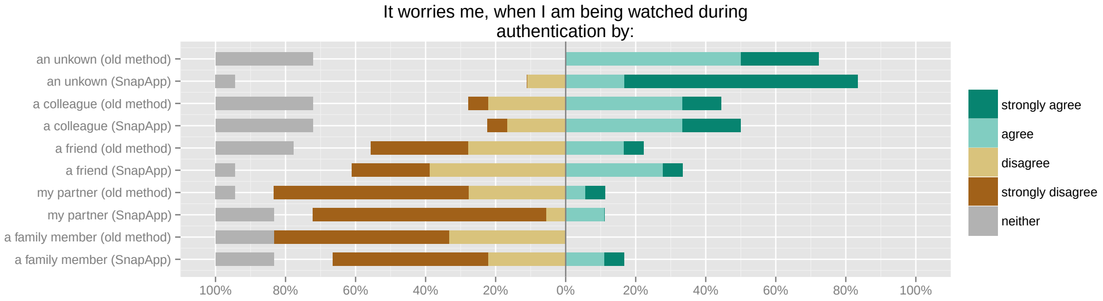


Figure 5.9: User ratings for incident one with the preceding unlock method and with SnapApp: worrying about being watched during authentication by an unknown, a colleague, a friend, the partner or a family member

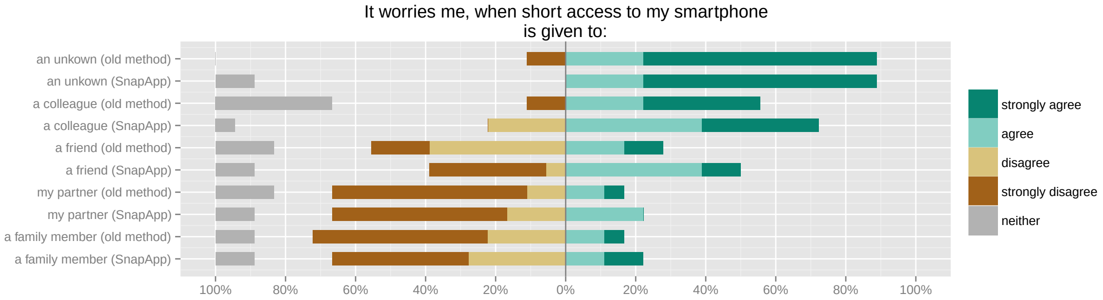


Figure 5.10: User ratings for incident two with the previous unlock method and SnapApp: Short access to the smartphone given to an unknown, a colleague, a friend, the partner or a family member

5.4.6 SnapApp experience evaluation

In both main questionnaires, participants have been asked to rate the used unlock method ("With [my current unlock method / SnapApp as unlock method]...") in the categories speed ("the smartphone access is fast"), convenience ("the smartphone access is simple") and data security ("my data is safe"). Figure 5.11 shows the results of the ratings for both unlock methods. While most of the users strongly agreed that the old unlock method was fast, about 10% less agreed that SnapApp allows quick access, too. While PIN and pattern users had a fast alternative by using *Snaps*, swipe users needed to unlock the phone by PIN, which could have been a bigger burden to them. The simplicity of the access by *Snaps* was rated with over 30% less agreement and even received about 15% more disagreement compared to the preceding unlock method. The reasons for the experienced inconvenience may be on the one hand the changeover from swipe or pattern to SnapApp with PIN as secure unlock method, which could have been apprehended as uncomfortable. On the other hand, users may have experienced one of the two main problems of the SnapApp prototype, which was the occurrence of the system PIN prompt after the phone had been unlocked using SnapApp, referenced in this thesis as double PIN problem. The data safety rating showed that SnapApp was rated safer than the preceding unlock method. For one thing, the result could be due to the swipe users applying now a secure unlock method to protect their smartphone with SnapApp, as almost 40% of the participants disagreed to the previously used unlock method being safe. Otherwise, PIN and pattern users could have liked the possibility to lend their phones to trusted persons without sharing the unlock secret or the opportunity to use a *Snap* instead of PIN in insecure environments.

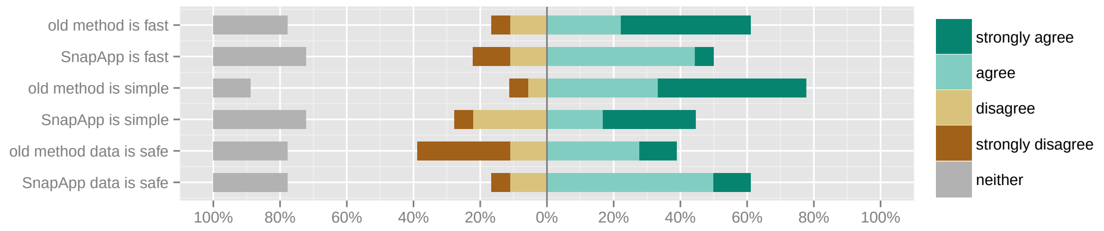


Figure 5.11: User ratings of the speed, the convenience and the security of the preceding unlock method and SnapApp

The next questions were asked in the after-study questionnaire only. They were either used to ensure that the logged data actually mirrors the usage behaviors or to check the popularity of single SnapApp features. At first, the questionnaire wanted to know if participants lock their smartphones before they put it away to assure that the logged session lengths were realistic. Over 80% of all 18 users agreed or strongly agreed to do so. About 60% stated that SnapApp was no time saver for them, while about 25% were neutral. This might be an indication for either the *Snap* concept not working out as expected or for delays during the unlock phase due to the mentioned double PIN problem. A little over 20% of the users felt stressed during *Snaps*, whereas over 60% stated to use another *Snap* when the previous one has ended. Two thirds of the answers proved that the end of *Snaps* disrupting the current action was annoying to the participants. The vibration alarm warning for the upcoming *Snap* end polarized the users' opinions, as it was found helpful by about 45% and simultaneously perceived as unhelpful by 50% of all users. To the statement of having sometimes thought that they could have used a *Snap* instead of a full access, over half of the participants agreed while almost 40% disagreed. This indicates that either users were sometimes not thinking about how long their unlock session would last or that PIN users may have automatically chosen PIN instead of a *Snap* out of habit.

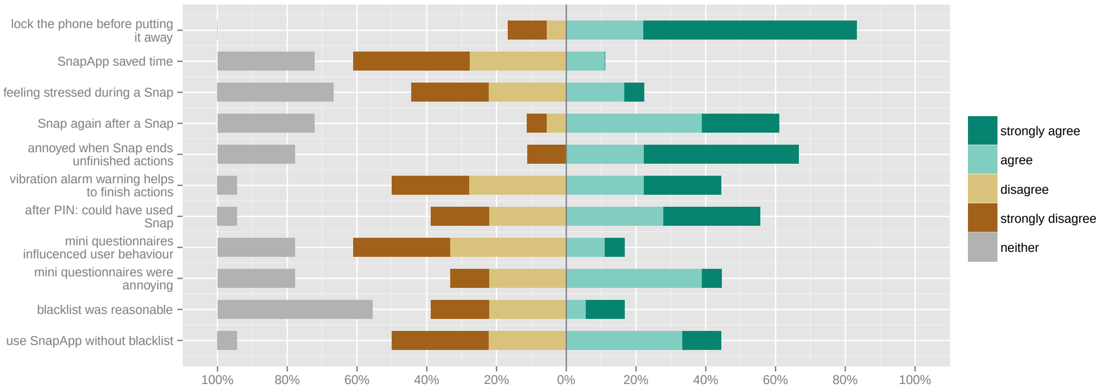


Figure 5.12: User ratings of various questions regarding SnapApp features and usage during the field study

Next, the in-situ mini questionnaires were evaluated. The users had to rate, whether the occurrences of the questionnaires influenced their user behavior. With over 60% disagreeing and about 20% neutral answers, the majority has confirmed that their normal usage behavior was not affected. Approximately 40% of all participants both agreed and disagreed of having been annoyed by the mini questionnaires. In this case, power users might have been sampled too much, which should be taken into account for further similar studies. Next, the reasonability of the blacklist feature was rated neutrally with over 40% and not reasonable with less than 40%. Accordingly, half of the users would not use SnapApp without the blacklist feature, while about 40% of them would do it. The blacklist results stand in contrast with the privacy perceptions and mirror the logged blacklist usage.

To rule out negative influencing factors, the participants were asked whether they have experienced the two main problems of the SnapApp prototype. The first problem were unexpected locks during the smartphone usage caused by the crashed background service of the prototype. An app update during the field study fixed the problem and was installed by all 18 users. The second main problem was the system PIN prompt popping up after the smartphone has already been unlocked by SnapApp. As presented in table 5.11, on twelve devices participants had to face both problems. Moreover, two more users experienced the unfixable double PIN problem. That is, 77.77% of the whole group were not able to use SnapApp without limitations. Three users had the unexpected locks problem which was eliminated during the study, while only one participant had no problems at all.

Table 5.11: Overview about the occurrences of both main problems on the participants' smartphones (n = 18)

		Unexpected locks	
		yes	**no**
Double PIN	**yes**	12	2
	no	3	1

Based on the gathered data of both the pre- and the after-study questionnaires, four different SnapApp unlock flows emerged. Both, the Android version installed on the device and the occurrence of the double PIN problem were the determining factors. Figure 5.13 presents four different screens labeled with the letters A to D.

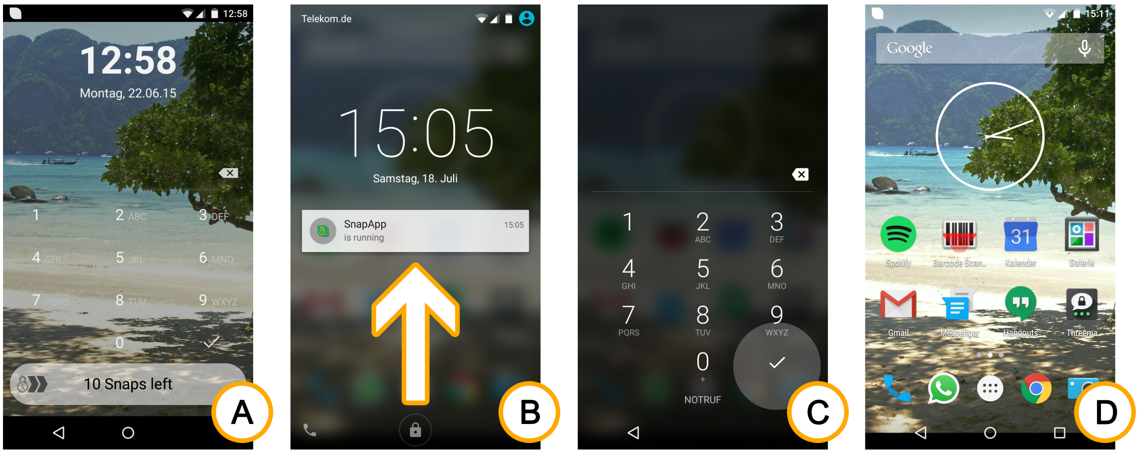


Figure 5.13: Different SnapApp unlock flows: ideal (A + D), Android 5 (A + B + D), double PIN (A + C + D), double PIN and Android 5 (A + B + C + D)

Letter A represents the SnapApp unlock screen. After the user decision for a *Snap* or a full access unlock by PIN, the next screen depends on the two previously mentioned factors. When the installed Android version number is 5, screen B is shown next. The Android 5 lockscreen underneath SnapApp had to be dismissed by doing an upwards slide starting at the bottom of the touchscreen display (*see arrow indication at screen B in figure 5.13*). If the double PIN problem was present, screen C was next. For participants with Android version 4 smartphones, screen C followed directly on screen A, provided the double PIN problem occurred. After screen C was dismissed by confirming an empty PIN prompt, screen D was shown next. Android version 4 users without the double PIN problem went straight from the SnapApp lockscreen (screen A) to the home screen (screen D), whereas Android version 5 participants had to stop at the Android 5 lockscreen (screen B) in between.

Summarizing, the four unlock flows were: the *ideal case* (A + D) directly from the SnapApp lockscreen to the home screen, the *Android 5* case (A + B + D) with the Android 5 lockscreen in the middle, the *double PIN* case (A + C + D) with the system PIN prompt in the middle and the *double PIN and Android 5* case (A + B + C + D).

At last, the participants were asked after the field study what they like to do with SnapApp: keep it and use it further without being logged or uninstall it and return to the preceding unlock method. Table 5.12 provides an overview about the user decisions and shows their dependencies to the occurrences of both main problems, the normally used unlock methods and the unlock flows of SnapApp on the participants' phones.

Table 5.12: After the field study: user decisions to keep or uninstall SnapApp, presence of the two main problems (system PIN prompt after SnapApp unlock and unexpected locks during *Snaps*), usually chosen unlock method and resulting SnapApp unlock flow (*see figure 5.13 for explanation*) (n = 18)

decision	**double PIN**	**unexpected locks**	**usual unlock method (SnapApp unlock flow)**
keep	No	No	PIN (AD)
SnapApp (3)	Yes	Yes	2x PIN (ACD)
uninstall	No	Yes	Swipe (AD), PIN (AD), Swipe (ABD)
SnapApp (15)	Yes	No	Pattern (ACD), Swipe (ABCD)
	Yes	Yes	Swipe (ACD), PIN (ACD), Swipe (ABCD), 3x Pattern (ABCD), 4x PIN (ABCD)

Three users stated that they would keep on using SnapApp. While the only participant without any problems (unlock flow AD) was one of them, the two others were PIN users who had to face the double PIN problem (ACD). Out of the 15 users who chose to uninstall SnapApp, three participants did not have the double PIN problem (AD and ABD), which means they abandoned the new unlock method although it has worked after the update fix. As two of the three had swipe as usual unlock method, SnapApp could have caused too much inconvenience. The remaining twelve users had all to face the double PIN problem, while another ten had experienced unexpected unlocks during the first half of the field study. It it also important to know, that no Android version 5 user (letter B in the unlock flow) has decided to use SnapApp any further.

5.4.7 Uninstalling reasons, praise, criticism and miscellaneous

The last part of the after-study questionnaire allowed users to describe freely in their own words what their reasons were to remove SnapApp, what they liked about it, what they would improve of the prototype and one comment box for anything else. All user comments are translated from German into English.

All 15 users who chose uninstalling left a comment to justify their decision. Their answers were categorized and are presented in order with most mentions first. Seven persons said they removed SnapApp due to the the double PIN problem and three stated they would do it because of the unexpected locks. Beside the two main problems, three swipe users stated that they do not want to decide which unlock method they should take with each usage. Another two mentioned that the delay before the SnapApp lockscreen showed up was to big. All of the following reasons were just mentioned once per user: one participant did not like the design, one wanted pattern instead of PIN as alternative unlock method, another one missed widgets on the lockscreen. For one PIN user, SnapApp was too insecure, while another PIN user was too accustomed to using PIN instead of a *Snap*. Apart from that, one user did not like that *Snaps* were expired all the time, whereas another one said the *Snap time* was too short. The authors of the last two statements have obviously not found the SnapApp settings panel, which allowed the configuration of both *Snap* expiration and *Snap time*.

17 participants provided eight reasons why they liked SnapApp. With seven times, the mostly mentioned reason was the "good" and "innovative" concept. Or, as one user with the unlock flow ABCD (*see figure 5.13*) said "theoretically convenient". Two users found that *Snaps* were useful, while another two liked the customizable *Snap time*. Also two users liked the implementation, with one user stating it is "successful and well elaborated". Two more users said they appreciate that their data is safer, whereby one normally chose swipe and the other one PIN as unlock method. Moreover, one participant valued the saved time by SnapApp and another one the possibility to use multiple *Snaps* in a row. At last, a swipe user reckoned SnapApp being easier than PIN only.

The improvement list generated by the aggregated feedback of 17 users contains mainly bug fixes and individual change requests. Four participants demanded general bug fixing, while five specified SnapApp appearance delays to be improved. The prototype main problems were mentioned four times with unexpected locks and three times with double PIN. Two users would like the whitelist to be the setup default of SnapApp, while for another one the vibration alarm warning was triggered too late. Other items on the list were the customizable *Snap* expiration time, haptic feedback for the digit buttons of the PIN pad, the request for animations, an adjustment of the slide-to-unlock touch sensitivity, two demands for widgets like the camera or the phone shortcut, another two requests for replacing PIN as the alternative unlock method with e.g. pattern and two wishes for design enhancements of the SnapApp lockscreen. The other requested improvements were either already features of SnapApp, not compatible with the prototype implementation or clashing with the SnapApp concept. Two users demanded *Snaps* should stop to expire while another one wanted an adjustable *Snap time* - both already possible in the prototype settings panel. One user requested availability of the system settings during *Snaps*. On the one hand, this was prevented per default as the settings allowed the deactivation of SnapApp during the study. On the other hand, the system settings give access to critical preferences as well as to the data of all stored user accounts of the phone. That is, they should never be available during short access. Finally, one user demanded that the phone activates the standby mode when a *Snap has ended*, while another one wanted the device to be unlocked after a call. In both cases, the improvements would destroy the SnapApp concept. *Snaps* have to end after the lapse of the *Snap time* regardless of the usage, e.g. calls. Moreover, it would be very annoying for participants who want to use multiple *Snaps* to turn the device on again every time.

Under miscellaneous, two user comments were submitted. One reported a bug of the system PIN prompt occurring since the installation of SnapApp, which could not be reproduced. The other participant stated that it had required some time to get used to SnapApp and said further: "...and now I have adapted to it so much that I am going to use the app even a little further.".

6 Discussion

On the first impression, the results could be seen rather negative: with 15 of 18 users, the majority chose to uninstall SnapApp after the field study instead of keep using it (*see section 5.4.6*). Taking a second look reveals, that the three participants who chose to use SnapApp as primary lockscreen were former PIN users valuing security. Proven by this fact, the time-constrained access control concept has obviously accomplished a successful trade-off between convenience and security for those. In total, *Snaps* were used by 17 of 18 users in over 20% of all unlock sessions (*see section 5.2.3*), which saved them valuable time. This result looks quite promising with the background, that 94% of all participants had to face at least one or both major problems of the prototype. While the first problem of unexpected locks could be fixed during the field study, it has still made a bad impression on some users. The second problem with the system PIN prompt was unfixable as it was a general Android bug and occurred more often than on the test devices during the development. With the great variety of different devices, manufacturers and individual implementations of the default lockscreen for each single smartphone, it was impossible to predict which phones would be affected by the known bug. This was aggravated by the random selection of users, whose device models were not taken into account for the choice. Both major problems were also the most quoted reasons to justify the removal of SnapApp (*see section 5.4.7*), which relativizes the high uninstalling rate.

A closer look at the outcomes is needed for a deeper understanding. For that reason, the results are discussed in the following part by answering the research questions, which were posed in the design part of the field study (*see section 4.1*).

How well were Snaps accepted as unlock method compared to PIN?

All the users met the expectations partly as they chose *Snaps* in total only for a small part of their short sessions. On average, 20.82% of all unlock sessions during the field study were *Snaps*, used by 17 of 18 participants. The remaining user has not tried the short access once. In that particular case, the average of four unlocks a day and the default *Snap* expiration after ten minutes since the last PIN entry lead to the fact, that *Snaps* were simply never available. Overall, participants rated *SnapApp* to be almost as fast, but at the same time safer as their usual unlock method. This means, PIN and pattern users feel no big difference regarding security, while swipe users see their data as a lot more protected. A great majority of disagreeing or neutral user ratings states that the participants are not stressed during *Snaps*, which means the time-constrained access method was accepted as a pleasant alternative. Understandably, most of the users confirmed to be annoyed when the end of a *Snap* interrupts their current action, but the same amount of people claimed to simply use another *Snap* again afterwards. Nonetheless, the majority of the participants voted *SnapApp* not to be simple, which is not due to the concept, but to the problems of the prototype. Generally, the average used *Snaps* rate could be much higher, but it was limited by various conditions.

The two main problems of the SnapApp prototype certainly had a major impact on the acceptance. Unexpected locks during *Snaps* were very annoying as stated in the feedbacks and made users insecure about the functionality of SnapApp, providing they have read the manual as advised in form of the frequently asked questions. The double PIN problem was surely another reason not to use *Snaps*. Independent of the chosen unlock method, for 14 of 18 users the system PIN prompt was shown. Although it could be simply dismissed by pressing the confirm button without doing a PIN entry, some users were probably not aware despite the note in the manual. An example shows: when a former swipe user with an Android 5 smartphone installed *SnapApp*, the needed interactions to see the home screen increased from one to three. Depending on the Android version and the problems which appeared, users had to do two or three interactions instead of one, by which *SnapApp* was definitely not a time saver anymore.

As already mentioned before, *Snaps* expired ten minutes after the last PIN unlock. Enabling the *Snap* expiration per default for security reasons was probably a bad decision. It reduced the opportunities to use short access enormously, as only four of 18 users deactivated it in the SnapApp settings. The expiration had the greatest impact on participants who used their smartphones less frequently. Although the manual explained how to switch it off, one user complained that it would be a missing feature. Another one wanted to adjust the ten minutes value before the expiration. Again, the small number of users who had changed the expiration leads to the assumption that most of the participants have not used the settings.

Seven participants used *Snaps* above average ranging from 30% to 70% and had mainly pattern or swipe as usual unlock method. The remaining eleven users had *Snap* to full unlock rates of 13.22% and below. Among them, the great amount of seven PIN users stands out with a share of 63%. Especially PIN users were chosen for the study, as they had been familiar with one of the two SnapApp unlock methods before. An extensive amount of password and pattern users was expected to have an distorting impact on the results. The data now shows it is exactly the other way round, as PIN users are evidently so much accustomed to use their usual unlock method that they forget about the *Snap* alternative. This is supported by the feedback from the after-study questionnaire where a majority of the participants has admitted to have sometimes thought after a PIN unlock that they could have used a *Snap* instead. Swipe users on the other hand are hard to convince, which is proven by the fact that two participants who had none of the problems after the app update chose to uninstall anyway. In the feedback, three swipe users said they were not willing to decide which unlock method to take for every usage. Another one stated to have learned to prompt PIN again and got used to it at such a rate, that the decision which unlock method to use has almost always been PIN.

Did users adapt the Snap time and the Snap expiration to their needs?

The *Snap time* was changed by 14 of 18 participants. The remaining four have either liked the preset value or they have not found the settings panel. Eleven users started in total with the default *Snap time* of 35 seconds. During the study, five users changed it once, another five twice and four users three times before they have found their matching value. The most used *Snap time* values for short sessions were 35 seconds, 120 seconds 30 seconds and 60 seconds, which perfectly match the envisaged time range of the concept. The average *Snap* session time was 50.49 seconds (sd = 70.09s) and the logged session lengths show peaks at the most used *Snap time* values. Nonetheless, the distribution of the other session lengths was balanced, which means the user-defined *Snap time* values were chosen well and sessions were also ended by the users themselves and not by a *Snap time* lapse. The low rate of 28.29% for logged locks caused by the lapse of the *Snap time* also confirms, that users chose right, as well as the results of the after *Snap* mini questionnaires. After two thirds of all rated *Snaps*, the majority needed no further time followed by the demand of a quarter of a half *Snap time* length more. Users stated in the after *Snap* mini questionnaires they were often still typing and just not finished, but in the after-study questionnaire they said they would then simply use another *Snap* again. When the participants voted for longer extensions of the *Snap time* in the in-situ mini questionnaires, they mostly realized they need a full access instead for their spontaneously longer activity. One particular user stood out with the highest of all *Snap time* values of 2 hours and 35 seconds. This user converted SnapApp by the conjunction of the extremely long *Snap time* and an elaborated blacklist to an application-centric unlock approach, as the chance of ever reaching the end of a *Snap* was basically zero.

The *Snap* expiration was turned on per default and was switched off by only four of 18 participants. As 14 users also have changed the *Snap time*, they must have seen the option to turn it off. As there was one feedback reporting the option as missing, some users might have overseen it anyway.

Were apps used differently during Snaps and full access?

The application usage was examined by selecting the top 20 most used apps per user and unlock method. The first insight was, that fewer apps were used during *Snaps* than during full access sessions. Messaging was by far the most used app category for *Short* sessions, followed by the other category which includes the Google App for searching, the clock and the calendar most popular apps. This correlates with the general short usage times. Messaging often requires just a few words to reply and searching something, setting an alarm or quickly checking the next appointments are also fast things to do. During full unlock sessions, other and messaging changed rankings. With further searching, changing system settings and browsing the Google Play Store as most used apps, full access is needed as they either require more time or need extended privileges. The average *Snap* session endured 50.49 seconds, whereas the full unlock session ended with a mean of 198 seconds. The different amounts of usages for some categories can be simply explained by the session lengths. Reading e-mails or watching videos takes normally more time, which is why they are more often used while having full access. On the other hand, changing songs in the music category or quickly checking the journey planner are classical short actions which were more found during a *Snap*. Nevertheless, there were several app categories which almost ranked exactly the same: phone calls, web browsing, games, social networks and photography. Regardless of the same usage frequency, it is definitely a different usage. The session times differ, which means apps could not be used for the same time lengths as they occur at the same usage frequency. One example for the different usage was measured in the photography category: while the camera was more used during *Snaps* to quickly take pictures, the gallery was launched more frequently to view the images. Summed up, gallery and camera generated the same usage frequency. The same could also apply within one application: to be launched equally often, a web browser could have been used to flip through the news headlines during *Snaps* while it was used for the same amount of usages to comfortably read the whole article to the headline.

Was the blacklist feature used?

The blacklist feature was used passively by 16 of 18 users, as everyone of them had at least one logged forced lock during a *Snap*. However, most of them were caused by a launch of either the SnapApp or the system settings, which were blacklisted per default and could not be removed from the list by the user. During the concept development, the blacklist solution was preferred before a whitelist approach. It was assumed that people are not willing to go through a long configuration process. As only five users had actively configured their blacklist, the blacklist choice and the precaution measure of disabling both settings to prevent access to safety-critical preferences turned out to be the right choice. One of the five users decided to use the whitelist approach by adding all apps to the blacklist and removing just the camera app, a web browser, a pdf reader and one messaging client. The other four participants selected one to 14 apps to be blacklisted. Depending on the user preference, messaging apps, e-mail clients, contacts, tools having critical privileges or apps connected to any kind of account were added to the blacklist. Overall, the participants seem to have rather different views on which apps are worth protecting and which are not. However, the majority of the users were too lazy to configure a blacklist at all, which was backed by according feedback comments and testifies a general recklessness.

Do the user-defined settings and configurations mirror their privacy and security concerns?

According to the results, among the top four as most important user rated aspects of privacy were the following: to be able to share confidential matters with someone they trust, not to be watched or listened without permission and controlling who gets their information. Figuratively spoken for smartphone usage, users want to control who can access the information stored on their phones. Besides, when sensitive data (alias confidential matters) should only be shared with persons of trust, it has to be protected. With the adjustable *Snap* expiration and the configurable blacklist SnapApp provides two great possibilities to protect important data from unrestricted access. Being watched without permission is not always preventable, but the smartphone can be unlocked by using slide-to-unlock to start a *Snap* instead of exposing the PIN prompt for full access to unwanted observers. Participants also stated that they are worried, especially when strangers, colleagues or friends would have access to their phones running SnapApp. As a consequence, all users should have activated the *Snap* expiration and added all sensitive apps to the blacklist while setting the *Snap time* to a short adequate value.

Snap expiration was switched on by the majority, but as mentioned this was the default setting. The adjusted *Snap times* were within reason, except one extra high 2 hours 35 seconds value of one user. On the other hand, the person concerned had distinctly blacklisted ten sensitive apps. With only another four blacklist users, the vast majority declined to configure their blacklists. Either they were too lazy or they were satisfied with *Snaps* being limited to ten usages in a row and expiring automatically after ten minutes. When the participants were asked, they rated SnapApp on average being more secure than their old unlock method. Thinking of swipe users who chose convenience to be way more important than any security before, using SnapApp with its default settings and without a configured blacklist was still a major improvement to security. But as a matter of fact, the two most accurate defined blacklists out of all five were configured by swipe users, while the other three belonged to PIN users. Users commented in the feedback that “a lot of apps should be on the blacklist” and that they “should use the blacklist”, so they were at least aware what they still had to do. The most ratings for the blacklist being reasonable were neutral and disagreeing. Otherwise, when the participants were asked whether they would use SnapApp without the blacklist feature, 45% voted rather no while 40% tended to yes.

Summarizing, the blacklist was only partly used, while the settings for the *Snap time* and the *Snap* expiration were reasonable. In total, the usage of all features in different extents were still sufficient enough to get a better safety rating than the preceding unlock methods of the participants.

7 Conclusion

This thesis presented the conception of time-constrained access control for mobile devices, its implementation as an installable application prototype for Android smartphones called SnapApp. Furthermore, it shows the results of a user study, which evaluated the acceptance of the new unlock method and usage behavior. The concept was developed and consequently supplemented with additional security features (*see section 3.5*), before it was implemented as a final prototype (*see section 3.1*). SnapApp was then evaluated in a longitudinal field study (*see section 4*).

SnapApp managed to reduce explicit authentication (PIN) by one fifth and saved valuable time without impairing security. Without inventing something completely new, SnapApp uses two very well known unlock methods. On the one hand PIN, a familiar authentication method, which was already used before the invention of the smartphone touchscreen. On the other hand, it takes the fast and easy to use slide-to-unlock method and combines it with a timer, a counter and a list. This list contains prohibited apps, which are not allowed to be launched during short access, which is repeatedly checked during a *Snap*. When the SnapApp concept was developed, it became clear that the time-constrained alternative unlock method without authentication needs security features to protect sensitive data. Time-constrained means the time values were limited and called *Snap time*. As repeatedly used *Snaps* are basically full access, a limiting counter was introduced, allowing a maximum of ten short sessions in a row. At last, the blacklist restricting access to apps and the *Snap* expiration after ten minutes since the last PIN entry were added. Previous studies have shown that session lengths, perceptions of security and data sensitivity are varying from user to user [21, 7]. Consequently, the *Snap time*, the *Snap* expiration and the blacklist app items were configured by the user. The study results show that each user adapted the configuration in individual extents. The ratios of short to full access differed, as well as the number of used apps, session lengths, blacklist entries and *Snap* expiration settings. SnapApp adapts to the user by being as individual as the user himself. The security is kept at all times to the user's needs. The standard settings protect the data on the phone completely from the beginning, as *Snaps* expire per default ten minutes since the last PIN unlock. Besides that, users can edit the settings, restrict the time of short sessions and limit app usage to a single application. The distinguishing difference to other reducing unlock methods is, that SnapApp needs no extra sensors or complicated algorithms to decide, whether to show a PIN prompt or not. Instead, the user decides. As SnapApp does not rely on extra hardware, it runs on any Android smartphone with touchscreen. The limited number of participants of the user study is not representative and the prototype had two major drawbacks. Nonetheless, this thesis has proven that time-constrained access control worked well and has thereby laid the foundation for further investigations. Finding new ways does not require the reinvention of the wheel.

8 Acknowledgements

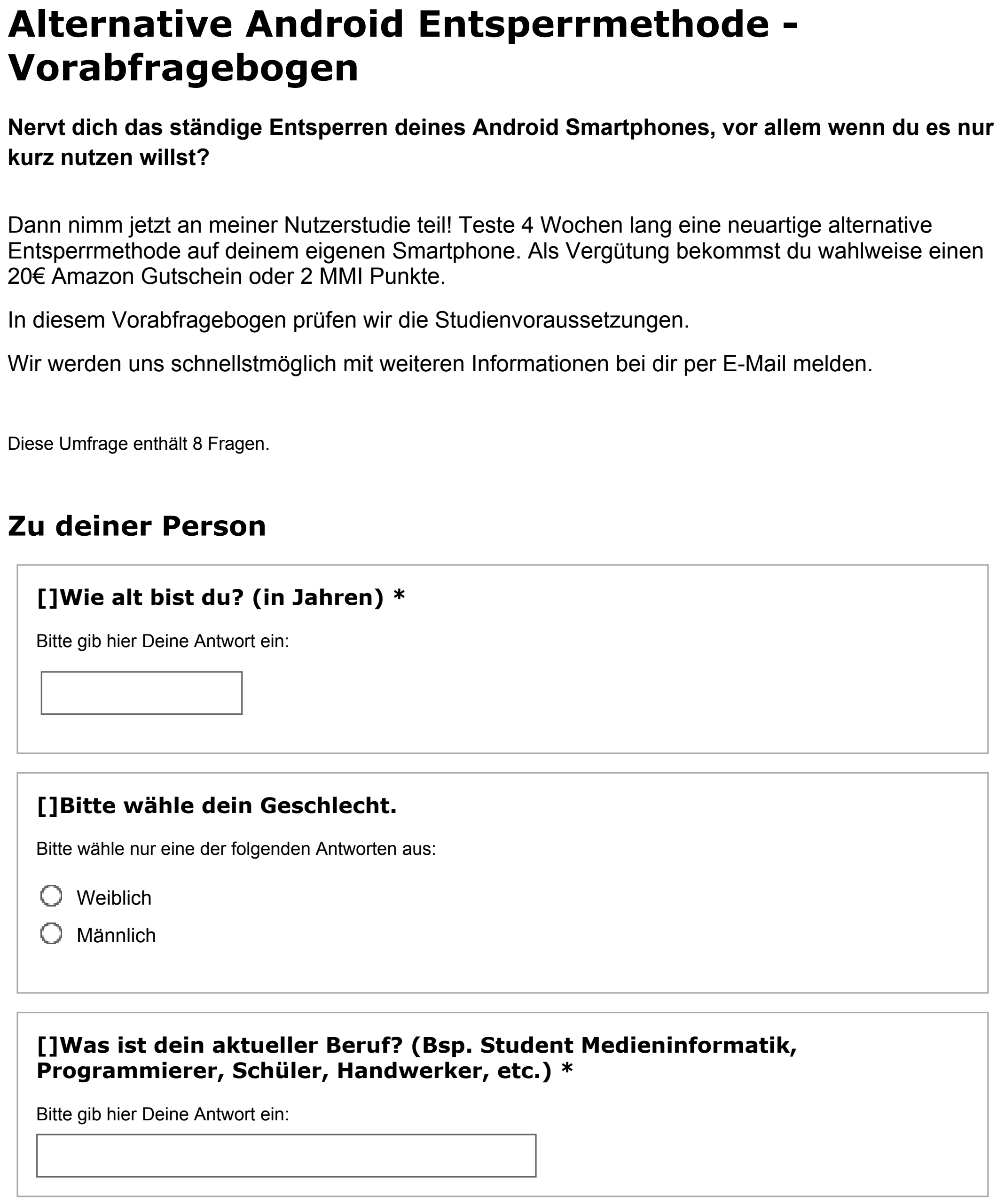
I would like to thank Daniel Buschek, Emanuel von Zezschwitz and Florian Alt for their support and patience, as well as my friends and my family. Moreover, I thank everyone who was participating in the user study.

Also, thanks to Freepik for the free usage of icons in the application prototype.

A Recruitment message

Nervt dich das ständige Entsperren deines Android Smartphones, vor allem wenn du es nur kurz nutzen willst? Dann nimm jetzt an meiner Nutzerstudie teil! Teste 4 Wochen lang eine neuartige alternative Entsperrmethode auf deinem eigenen Smartphone. Als Vergütung bekommst du wahlweise einen 20 Amazon Gutschein oder 2 MMI Punkte. Mehr Informationen und Teilnahmevoraussetzungen **>HIER<**

B Preliminary short survey



Alternative Android Entsperrmethode - Vorabfragebogen

Nervt dich das ständige Entsperren deines Android Smartphones, vor allem wenn du es nur kurz nutzen willst?

Dann nimm jetzt an meiner Nutzerstudie teil! Teste 4 Wochen lang eine neuartige alternative Entsperrmethode auf deinem eigenen Smartphone. Als Vergütung bekommst du wahlweise einen 20€ Amazon Gutschein oder 2 MMI Punkte.

In diesem Vorabfragebogen prüfen wir die Studienvoraussetzungen.

Wir werden uns schnellstmöglich mit weiteren Informationen bei dir per E-Mail melden.

Diese Umfrage enthält 8 Fragen.

Zu deiner Person

[]Wie alt bist du? (in Jahren) *

Bitte gib hier Deine Antwort ein:

[]Bitte wähle dein Geschlecht.

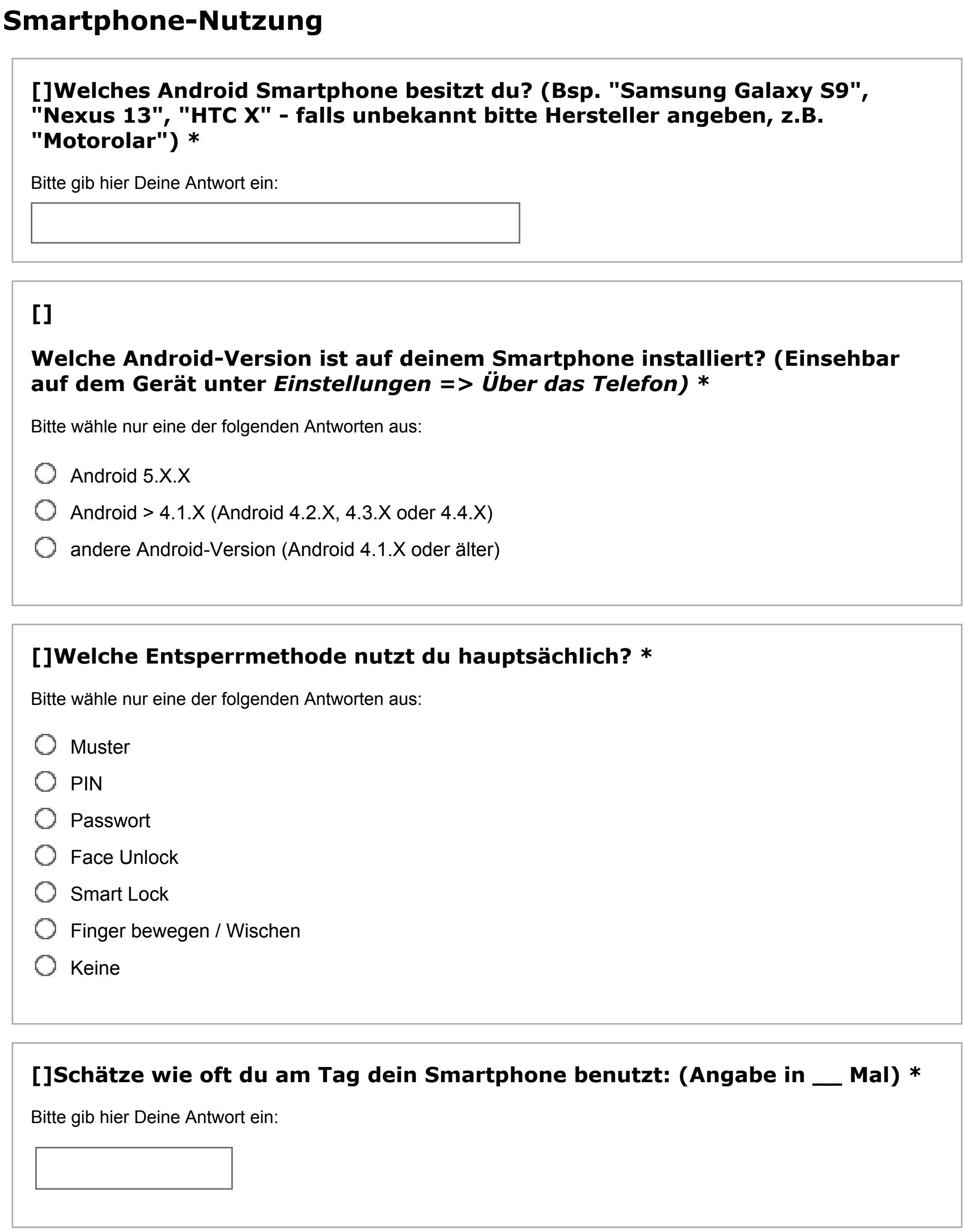
Bitte wähle nur eine der folgenden Antworten aus:

○ Weiblich

○ Männlich

[]Was ist dein aktueller Beruf? (Bsp. Student Medieninformatik, Programmierer, Schüler, Handwerker, etc.) *

Bitte gib hier Deine Antwort ein:



Smartphone-Nutzung

[]Welches Android Smartphone besitzt du? (Bsp. "Samsung Galaxy S9", "Nexus 13", "HTC X" - falls unbekannt bitte Hersteller angeben, z.B. "Motorolar") *

Bitte gib hier Deine Antwort ein:

[]

Welche Android-Version ist auf deinem Smartphone installiert? (Einsehbar auf dem Gerät unter *Einstellungen => Über das Telefon)* *

Bitte wähle nur eine der folgenden Antworten aus:

○ Android 5.X.X

○ Android > 4.1.X (Android 4.2.X, 4.3.X oder 4.4.X)

○ andere Android-Version (Android 4.1.X oder älter)

[]Welche Entsperrmethode nutzt du hauptsächlich? *

Bitte wähle nur eine der folgenden Antworten aus:

○ Muster

○ PIN

○ Passwort

○ Face Unlock

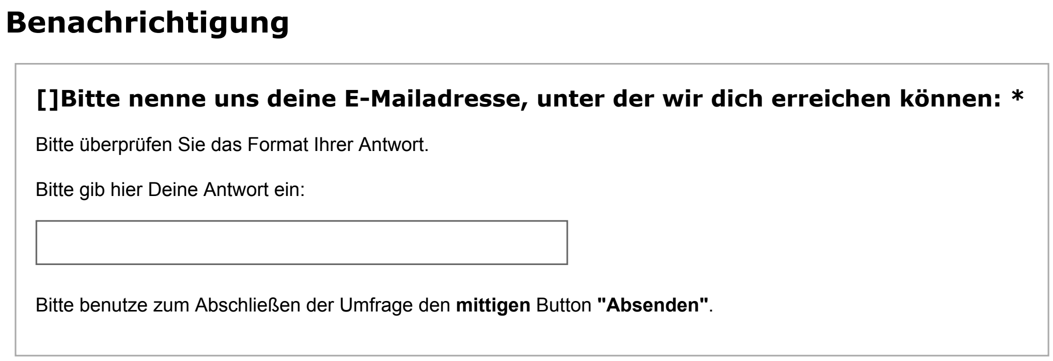
○ Smart Lock

○ Finger bewegen / Wischen

○ Keine

[]Schätze wie oft du am Tag dein Smartphone benutzt: (Angabe in __ Mal) *

Bitte gib hier Deine Antwort ein:



Benachrichtigung

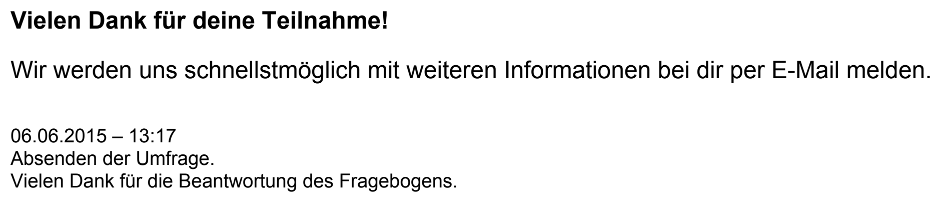
[]Bitte nenne uns deine E-Mailadresse, unter der wir dich erreichen können: *

Bitte überprüfen Sie das Format Ihrer Antwort.

Bitte gib hier Deine Antwort ein:

Bitte benutze zum Abschließen der Umfrage den **mittigen** Button **"Absenden"**.

B PRELIMINARY SHORT SURVEY



Vielen Dank für deine Teilnahme!

Wir werden uns schnellstmöglich mit weiteren Informationen bei dir per E-Mail melden.

06.06.2015 – 13:17
Absenden der Umfrage.
Vielen Dank für die Beantwortung des Fragebogens.

C Pre-study questionnaire

SnapApp Fragebogen Start

Diese Umfrage enthält 14 Fragen.

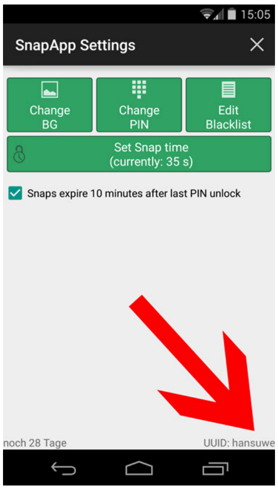
SnapApp UUID

[]Bitte trage deine SnapApp UUID ein: *

Bitte gib hier Deine Antwort ein:

Wo finde ich meine SnapApp UUID?

Öffne die SnapApp Einstellungen über das SnapApp-Icon in deinem App-Drawer. Rechts unten sollte jetzt ein Name stehen - dieser wurde zufällig ausgewählt und ist deine UUID.



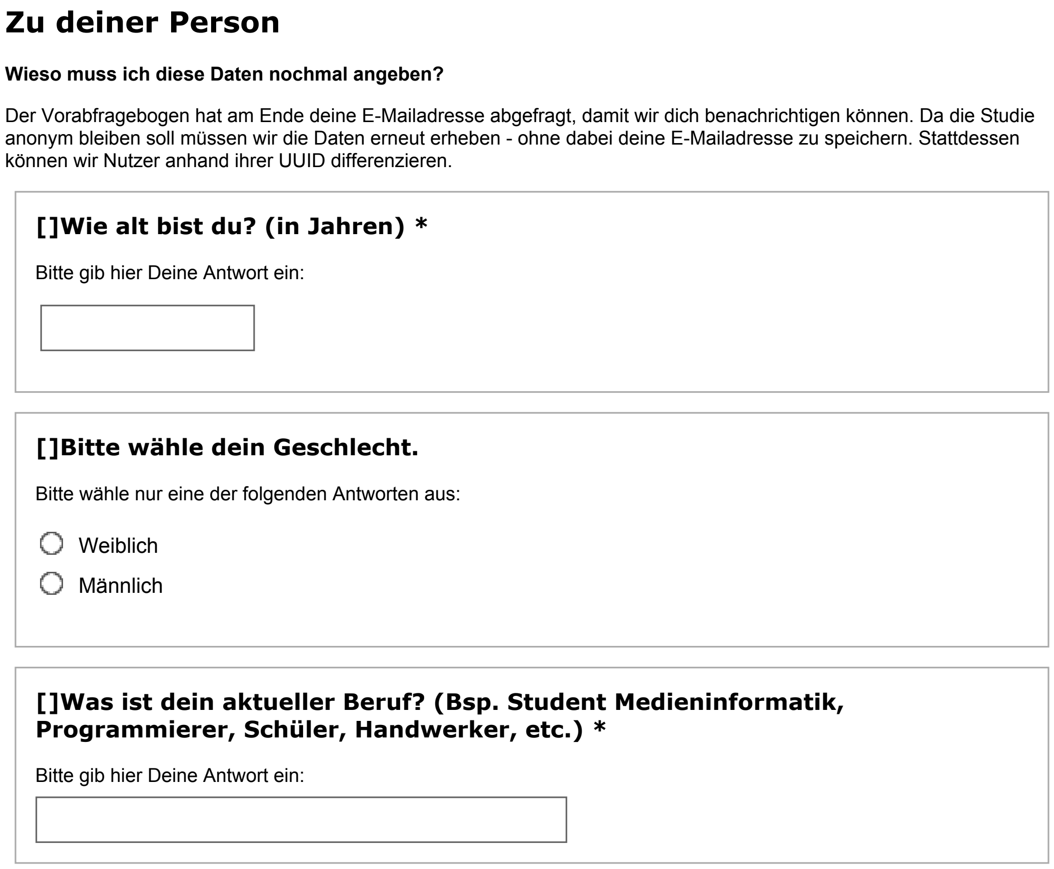
Set Snap time (currently: 35 s)

☑ Snaps expire 10 minutes after last PIN unlock

noch 28 Tage

UUID: hansuwe

Sollte dort noch **"no uuid"** stehen, dann stelle bitte sicher dass dein Smartphone eine Internetverbindung aufbauen kann und schalte danach den Bildschirm des Gerätes einmal aus und wieder ein. Nach dem Ensperrvorgang sollte jetzt eine UUID vorhanden sein.



Zu deiner Person

Wieso muss ich diese Daten nochmal angeben?

Der Vorabfragebogen hat am Ende deine E-Mailadresse abgefragt, damit wir dich benachrichtigen können. Da die Studie anonym bleiben soll müssen wir die Daten erneut erheben - ohne dabei deine E-Mailadresse zu speichern. Stattdessen können wir Nutzer anhand ihrer UUID differenzieren.

[]Wie alt bist du? (in Jahren) *

Bitte gib hier Deine Antwort ein:

[]Bitte wähle dein Geschlecht.

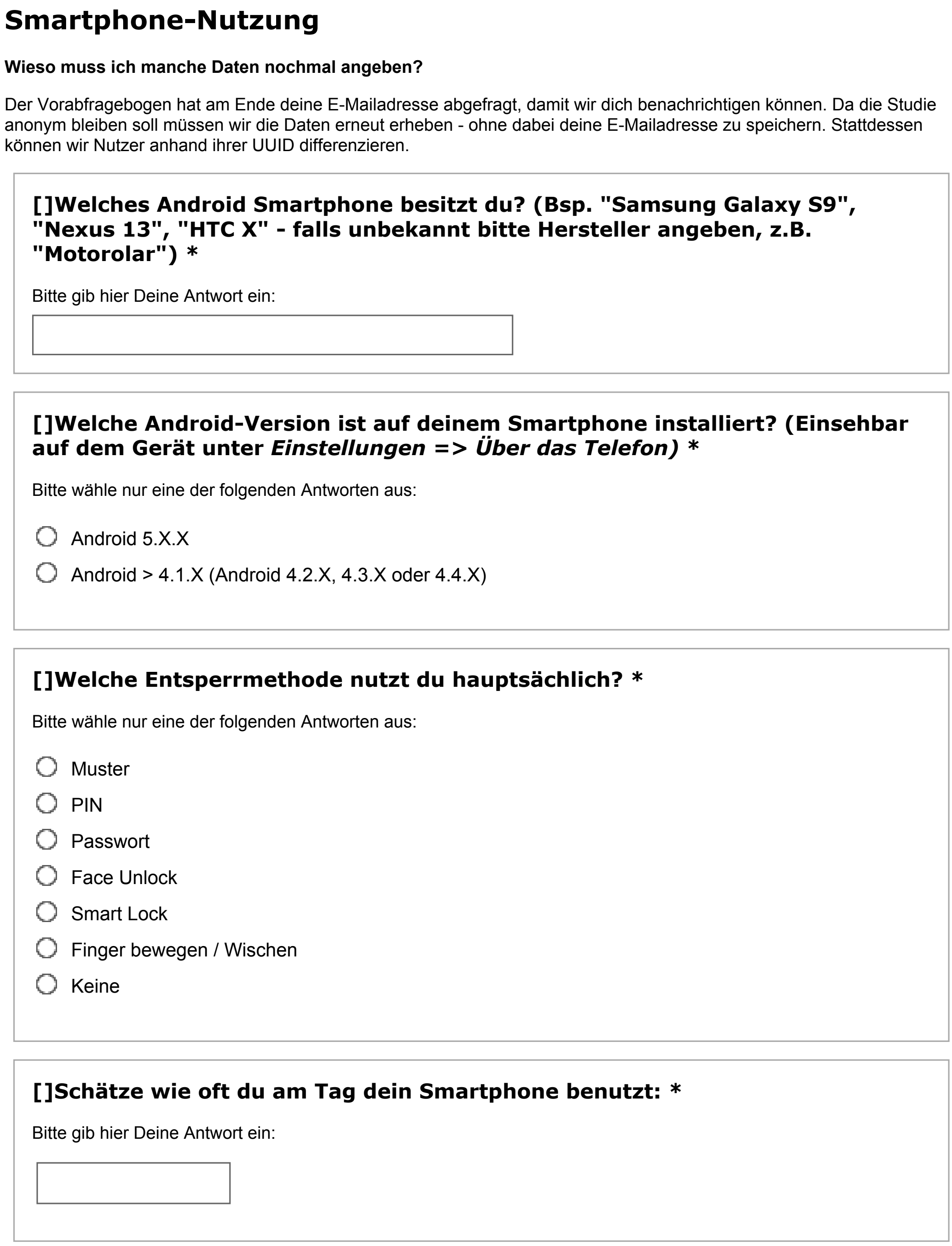
Bitte wähle nur eine der folgenden Antworten aus:

○ Weiblich

○ Männlich

[]Was ist dein aktueller Beruf? (Bsp. Student Medieninformatik, Programmierer, Schüler, Handwerker, etc.) *

Bitte gib hier Deine Antwort ein:



Smartphone-Nutzung

Wieso muss ich manche Daten nochmal angeben?

Der Vorabfragebogen hat am Ende deine E-Mailadresse abgefragt, damit wir dich benachrichtigen können. Da die Studie anonym bleiben soll müssen wir die Daten erneut erheben - ohne dabei deine E-Mailadresse zu speichern. Stattdessen können wir Nutzer anhand ihrer UUID differenzieren.

[]Welches Android Smartphone besitzt du? (Bsp. "Samsung Galaxy S9", "Nexus 13", "HTC X" - falls unbekannt bitte Hersteller angeben, z.B. "Motorolar") *

Bitte gib hier Deine Antwort ein:

[]Welche Android-Version ist auf deinem Smartphone installiert? (Einsehbar auf dem Gerät unter *Einstellungen => Über das Telefon)* *

Bitte wähle nur eine der folgenden Antworten aus:

○ Android 5.X.X

○ Android > 4.1.X (Android 4.2.X, 4.3.X oder 4.4.X)

[]Welche Entsperrmethode nutzt du hauptsächlich? *

Bitte wähle nur eine der folgenden Antworten aus:

○ Muster

○ PIN

○ Passwort

○ Face Unlock

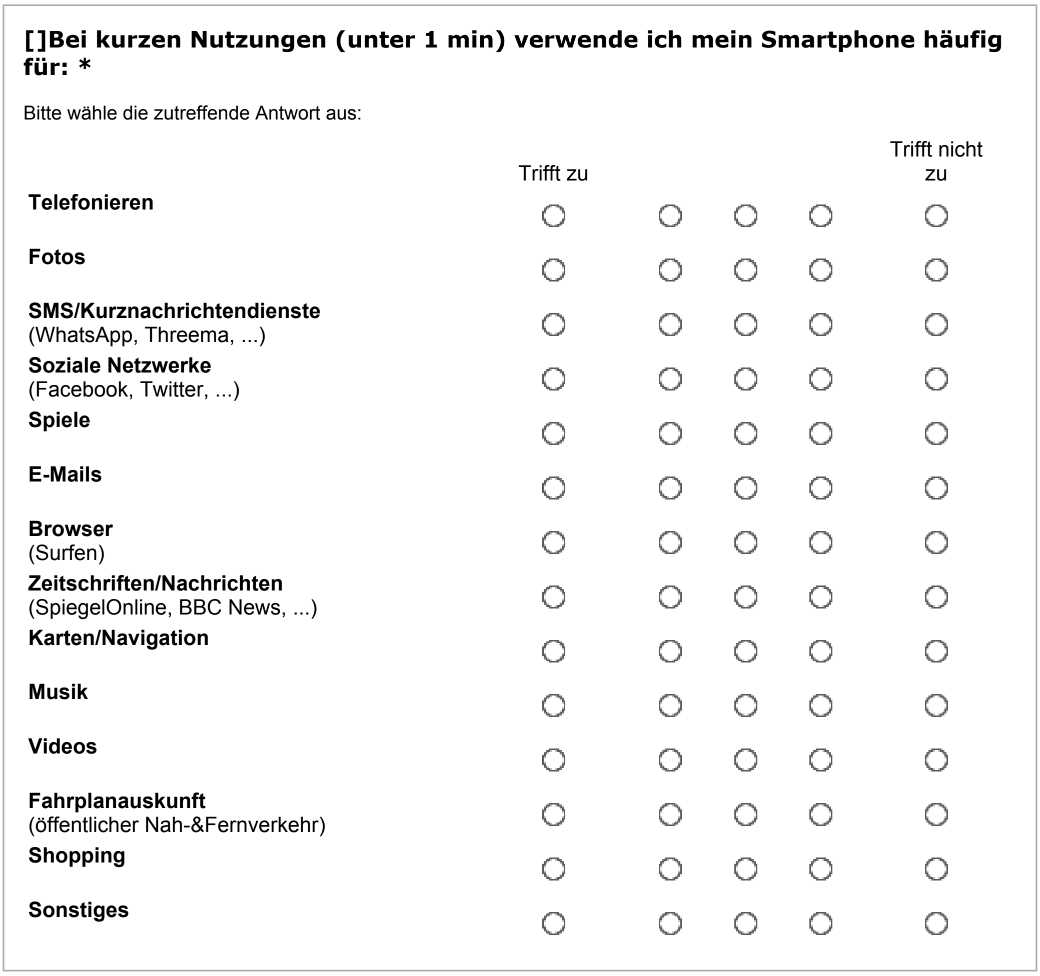
○ Smart Lock

○ Finger bewegen / Wischen

○ Keine

[]Schätze wie oft du am Tag dein Smartphone benutzt: *

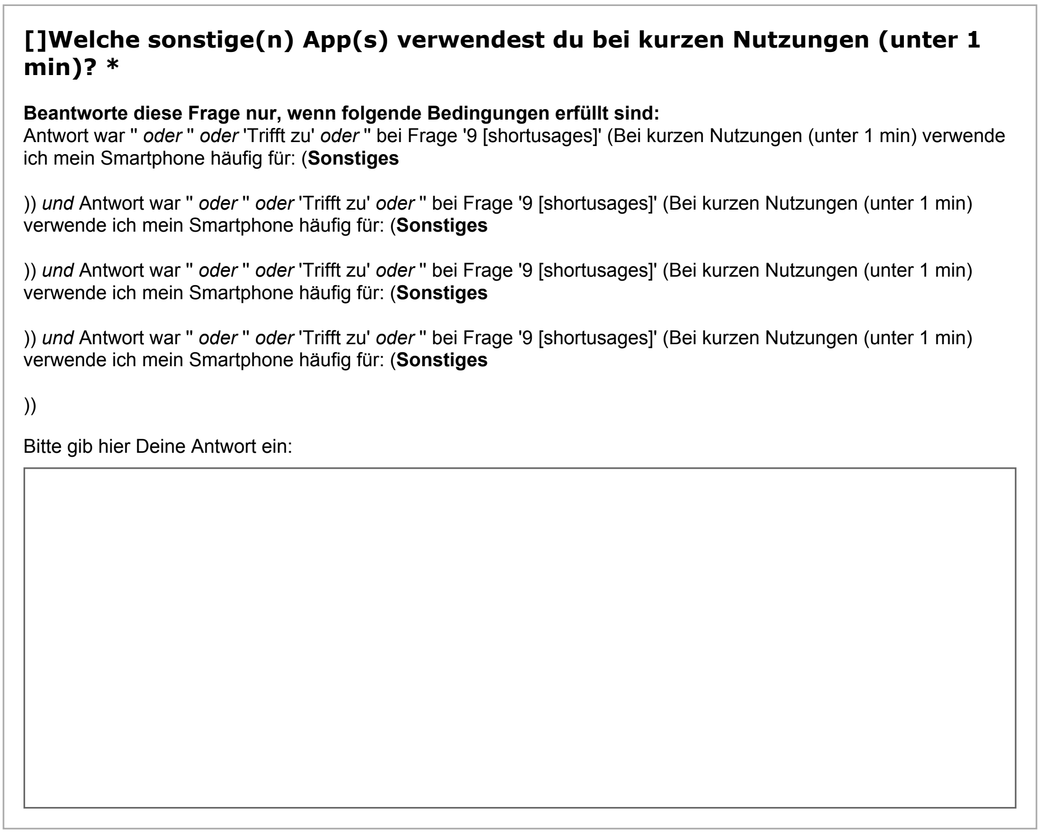
Bitte gib hier Deine Antwort ein:



[]Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
Telefonieren	○	○	○	○	○
Fotos	○	○	○	○	○
SMS/Kurznachrichtendienste (WhatsApp, Threema, ...)	○	○	○	○	○
Soziale Netzwerke (Facebook, Twitter, ...)	○	○	○	○	○
Spiele	○	○	○	○	○
E-Mails	○	○	○	○	○
Browser (Surfen)	○	○	○	○	○
Zeitschriften/Nachrichten (SpiegelOnline, BBC News, ...)	○	○	○	○	○
Karten/Navigation	○	○	○	○	○
Musik	○	○	○	○	○
Videos	○	○	○	○	○
Fahrplanauskunft (öffentlicher Nah-&Fernverkehr)	○	○	○	○	○
Shopping	○	○	○	○	○
Sonstiges	○	○	○	○	○



[]Welche sonstige(n) App(s) verwendest du bei kurzen Nutzungen (unter 1 min)? *

Beantworte diese Frage nur, wenn folgende Bedingungen erfüllt sind:
Antwort war '' *oder* '' *oder* 'Trifft zu' *oder* '' bei Frage '9 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

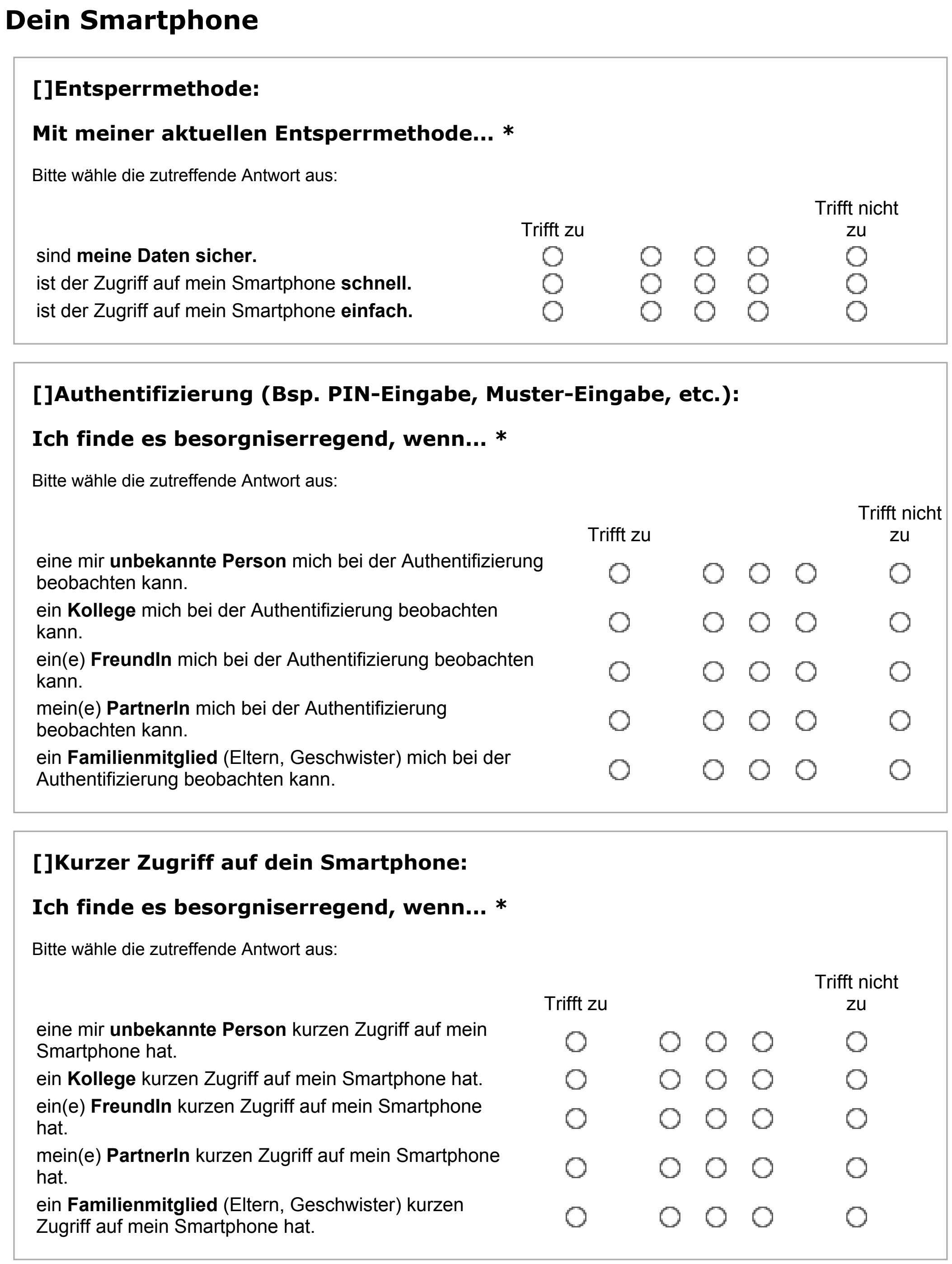
)) *und* Antwort war '' *oder* '' *oder* 'Trifft zu' *oder* '' bei Frage '9 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

)) *und* Antwort war '' *oder* '' *oder* 'Trifft zu' *oder* '' bei Frage '9 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

)) *und* Antwort war '' *oder* '' *oder* 'Trifft zu' *oder* '' bei Frage '9 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

))

Bitte gib hier Deine Antwort ein:



Dein Smartphone

[]Entsperrmethode:

Mit meiner aktuellen Entsperrmethode... *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
sind **meine Daten sicher.**	○	○	○	○	○
ist der Zugriff auf mein Smartphone **schnell.**	○	○	○	○	○
ist der Zugriff auf mein Smartphone **einfach.**	○	○	○	○	○

[]Authentifizierung (Bsp. PIN-Eingabe, Muster-Eingabe, etc.):

Ich finde es besorgniserregend, wenn... *

Bitte wähle die zutreffende Antwort aus:

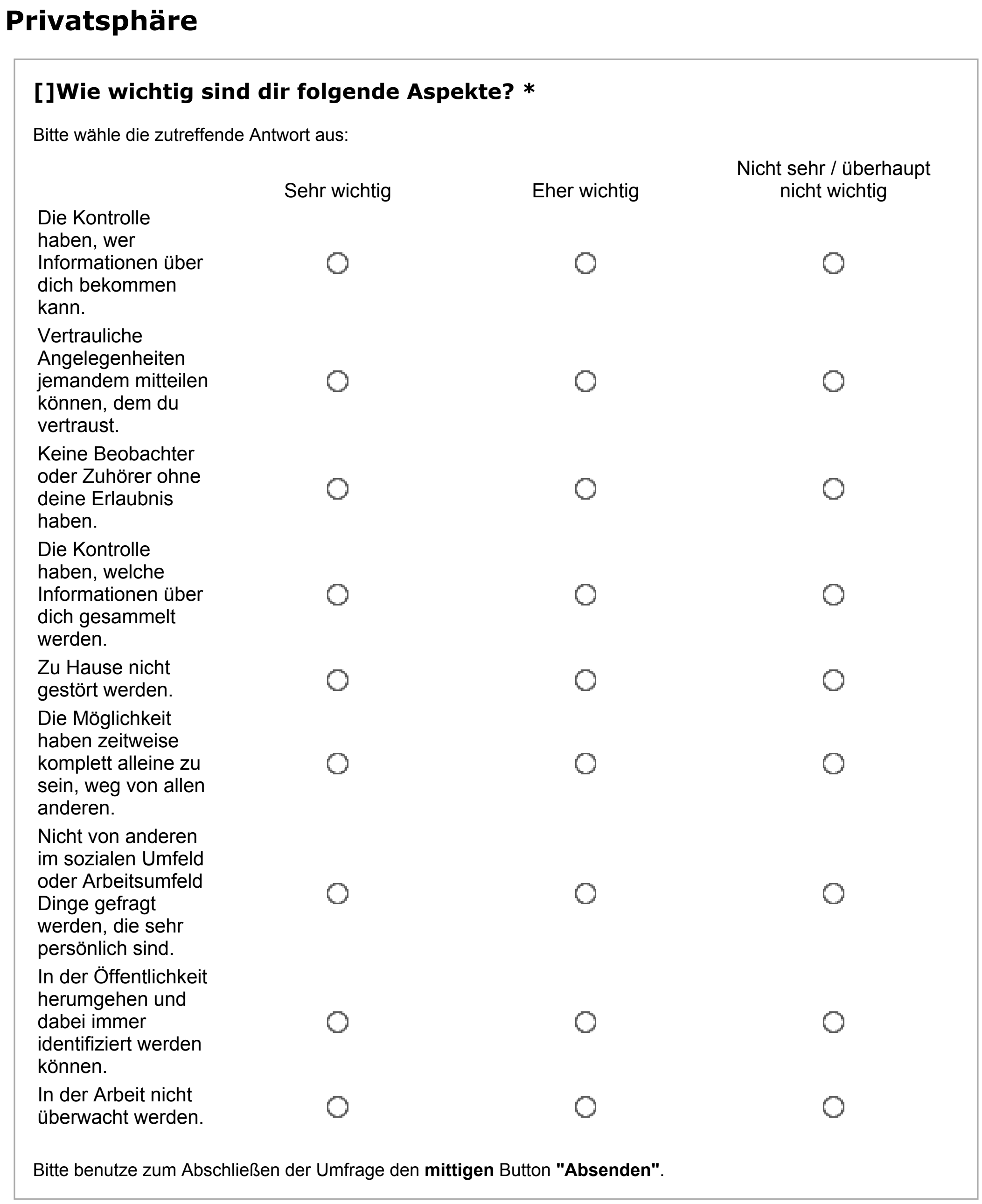
	Trifft zu				Trifft nicht zu
eine mir **unbekannte Person** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein **Kollege** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein(e) **FreundIn** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
mein(e) **PartnerIn** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein **Familienmitglied** (Eltern, Geschwister) mich bei der Authentifizierung beobachten kann.	○	○	○	○	○

[]Kurzer Zugriff auf dein Smartphone:

Ich finde es besorgniserregend, wenn... *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
eine mir **unbekannte Person** kurzen Zugriff auf mein Smartphone hat.	○	○	○	○	○
ein **Kollege** kurzen Zugriff auf mein Smartphone hat.	○	○	○	○	○
ein(e) **FreundIn** kurzen Zugriff auf mein Smartphone hat.	○	○	○	○	○
mein(e) **PartnerIn** kurzen Zugriff auf mein Smartphone hat.	○	○	○	○	○
ein **Familienmitglied** (Eltern, Geschwister) kurzen Zugriff auf mein Smartphone hat.	○	○	○	○	○



Privatsphäre

[]Wie wichtig sind dir folgende Aspekte? *

Bitte wähle die zutreffende Antwort aus:

	Sehr wichtig	Eher wichtig	Nicht sehr / überhaupt nicht wichtig
Die Kontrolle haben, wer Informationen über dich bekommen kann.	○	○	○
Vertrauliche Angelegenheiten jemandem mitteilen können, dem du vertraust.	○	○	○
Keine Beobachter oder Zuhörer ohne deine Erlaubnis haben.	○	○	○
Die Kontrolle haben, welche Informationen über dich gesammelt werden.	○	○	○
Zu Hause nicht gestört werden.	○	○	○
Die Möglichkeit haben zeitweise komplett alleine zu sein, weg von allen anderen.	○	○	○
Nicht von anderen im sozialen Umfeld oder Arbeitsumfeld Dinge gefragt werden, die sehr persönlich sind.	○	○	○
In der Öffentlichkeit herumgehen und dabei immer identifiziert werden können.	○	○	○
In der Arbeit nicht überwacht werden.	○	○	○

Bitte benutze zum Abschließen der Umfrage den **mittigen** Button **"Absenden"**.

C PRE-STUDY QUESTIONNAIRE

06.06.2015 – 13:23
Absenden der Umfrage.
Vielen Dank für die Beantwortung des Fragebogens.

D After-study questionnaire

SnapApp Fragebogen Abschluss

Hallo lieber SnapApp-Nutzer,

vielen Dank für deine Teilnahme! Um die Studie vollständig abzuschließen, würde ich dich bitten dir die Zeit zu nehmen folgenden Fragebogen auszufüllen.

Achtung, sehr wichtig:

Falls ihr die Einverständniserklärung noch nicht ausgefüllt habt, könnt ihr diese hier herunterladen:

http://snapapp.fabian-hartmann.de/downloads/einverstaendniserklaerung_snapapp.pdf

Anschließend das ausgefüllte Formular einscannen / abfotografieren und per E-Mail an snapapp@fabian-hartmann.de oder alternativ per Post an:

Daniel Buschek
Universität München, LFE Medieninformatik
Amalienstr. 17
80333 München

Diese Umfrage enthält 18 Fragen.

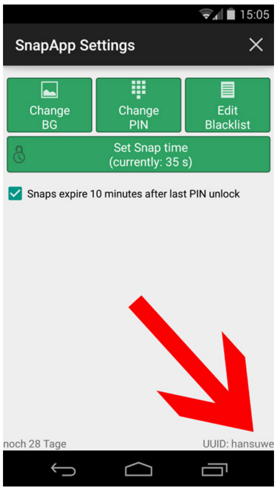
SnapApp UUID

[]Bitte trage deine SnapApp UUID ein: *

Bitte gib hier Deine Antwort ein:

Wo finde ich meine SnapApp UUID?

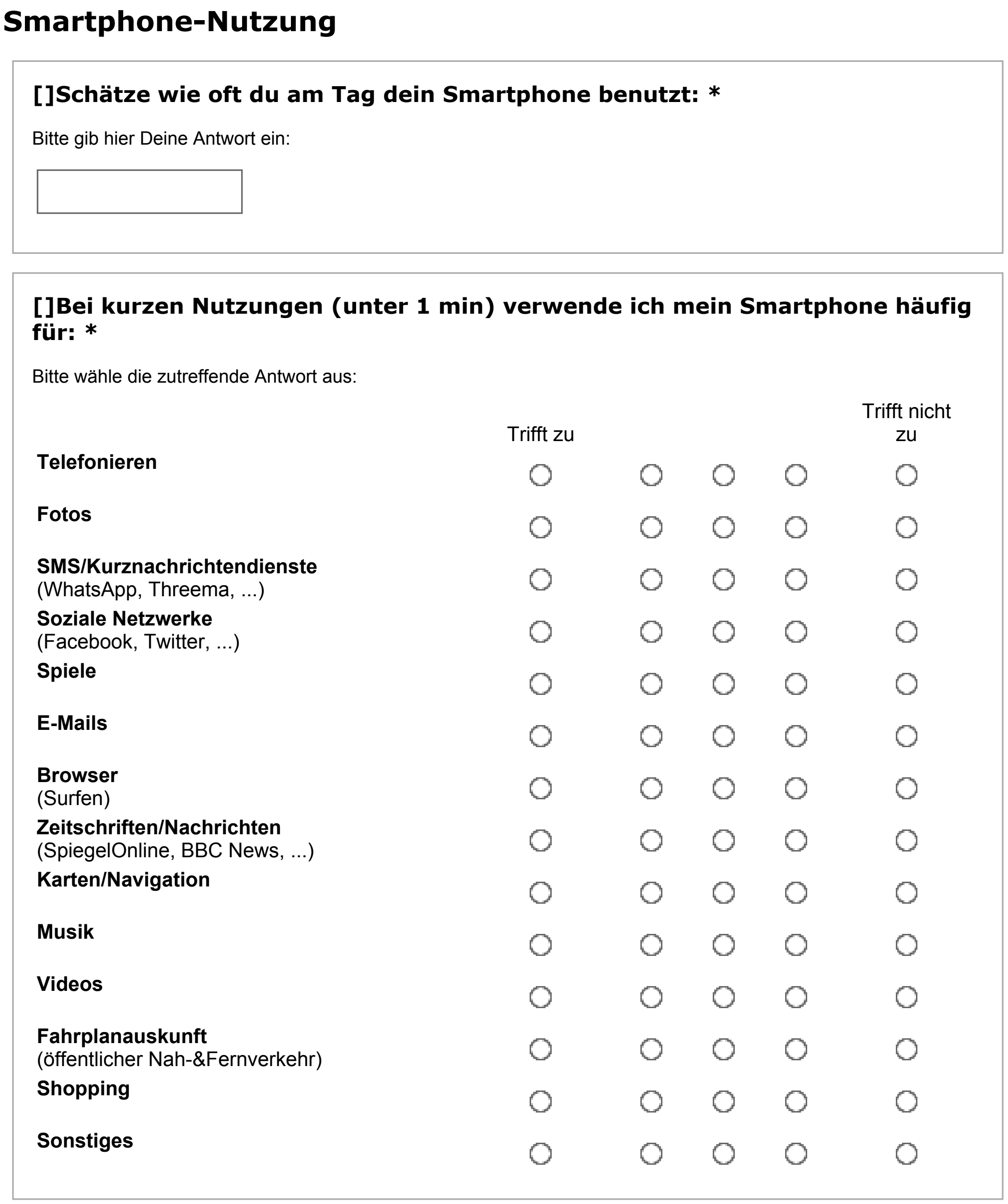
Öffne die SnapApp Einstellungen über das SnapApp-Icon in deinem App-Drawer. Rechts unten steht ein Name - dieser ist deine UUID.



☑ Snaps expire 10 minutes after last PIN unlock

noch 28 Tage

UUID: hansuwe



Smartphone-Nutzung

[]Schätze wie oft du am Tag dein Smartphone benutzt: *

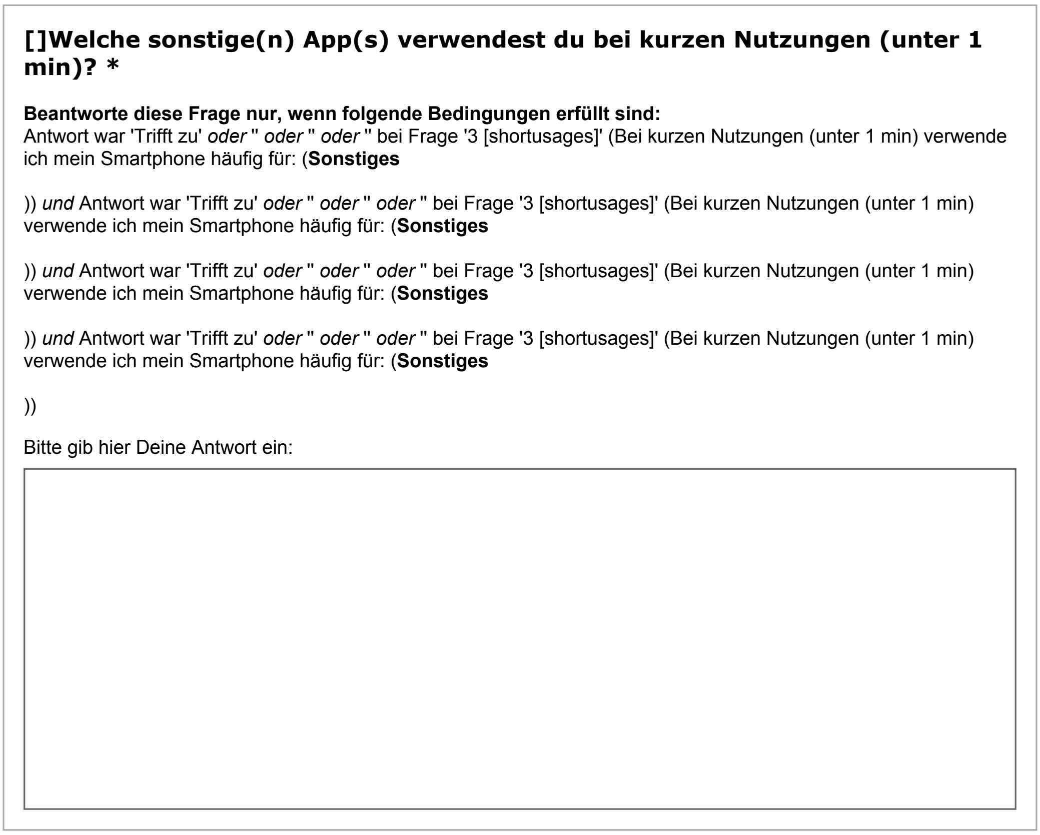
Bitte gib hier Deine Antwort ein:

□

[]Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
Telefonieren	○	○	○	○	○
Fotos	○	○	○	○	○
SMS/Kurznachrichtendienste (WhatsApp, Threema, ...)	○	○	○	○	○
Soziale Netzwerke (Facebook, Twitter, ...)	○	○	○	○	○
Spiele	○	○	○	○	○
E-Mails	○	○	○	○	○
Browser (Surfen)	○	○	○	○	○
Zeitschriften/Nachrichten (SpiegelOnline, BBC News, ...)	○	○	○	○	○
Karten/Navigation	○	○	○	○	○
Musik	○	○	○	○	○
Videos	○	○	○	○	○
Fahrplanauskunft (öffentlicher Nah-&Fernverkehr)	○	○	○	○	○
Shopping	○	○	○	○	○
Sonstiges	○	○	○	○	○



[]Welche sonstige(n) App(s) verwendest du bei kurzen Nutzungen (unter 1 min)? *

Beantworte diese Frage nur, wenn folgende Bedingungen erfüllt sind:
Antwort war 'Trifft zu' *oder* '' *oder* '' *oder* '' bei Frage '3 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

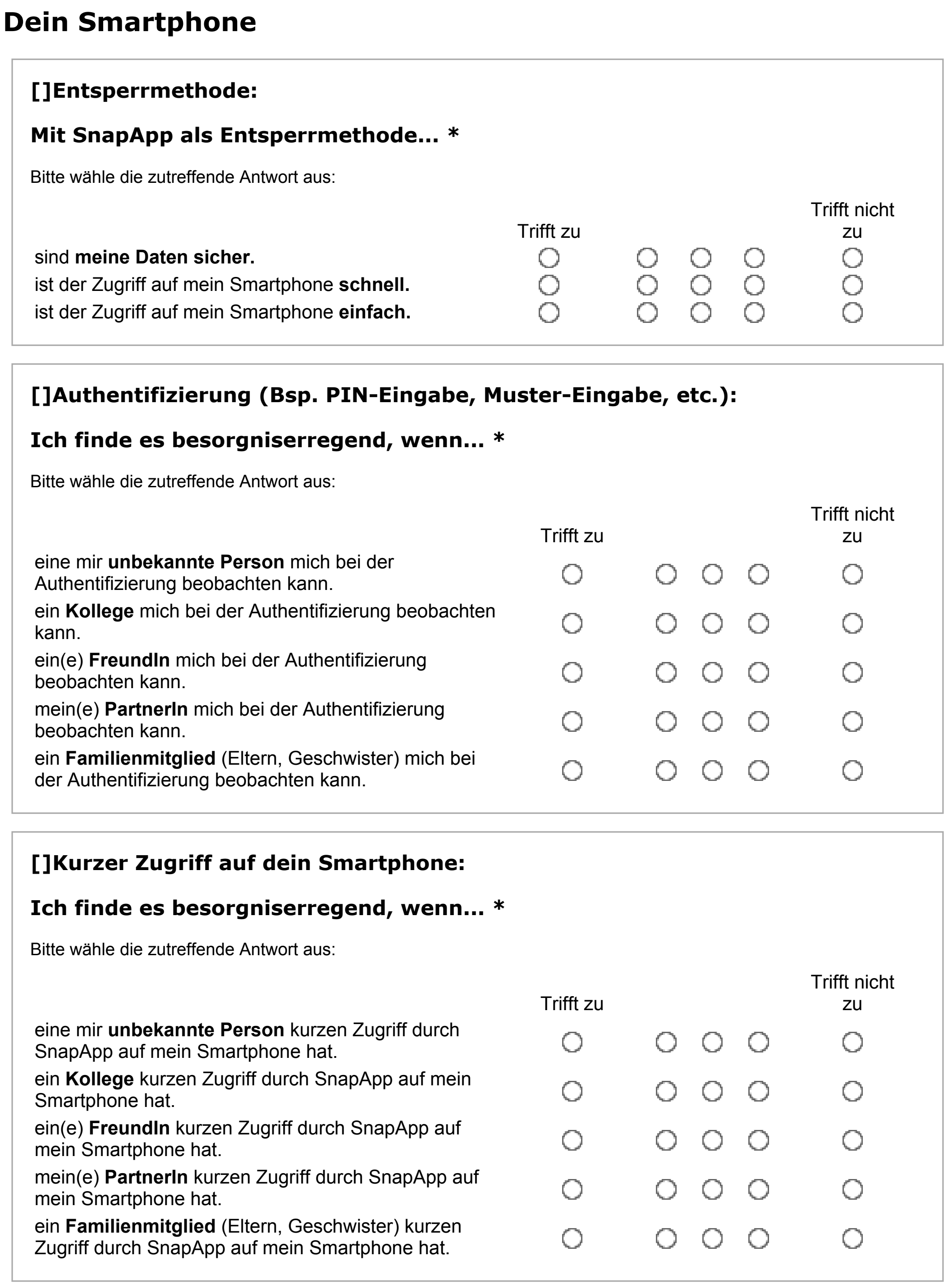
)) *und* Antwort war 'Trifft zu' *oder* '' *oder* '' *oder* '' bei Frage '3 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

)) *und* Antwort war 'Trifft zu' *oder* '' *oder* '' *oder* '' bei Frage '3 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

)) *und* Antwort war 'Trifft zu' *oder* '' *oder* '' *oder* '' bei Frage '3 [shortusages]' (Bei kurzen Nutzungen (unter 1 min) verwende ich mein Smartphone häufig für: (**Sonstiges**

))

Bitte gib hier Deine Antwort ein:



Dein Smartphone

[]Entsperrmethode:

Mit SnapApp als Entsperrmethode... *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
sind **meine Daten sicher.**	○	○	○	○	○
ist der Zugriff auf mein Smartphone **schnell.**	○	○	○	○	○
ist der Zugriff auf mein Smartphone **einfach.**	○	○	○	○	○

[]Authentifizierung (Bsp. PIN-Eingabe, Muster-Eingabe, etc.):

Ich finde es besorgniserregend, wenn... *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
eine mir **unbekannte Person** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein **Kollege** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein(e) **FreundIn** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
mein(e) **PartnerIn** mich bei der Authentifizierung beobachten kann.	○	○	○	○	○
ein **Familienmitglied** (Eltern, Geschwister) mich bei der Authentifizierung beobachten kann.	○	○	○	○	○

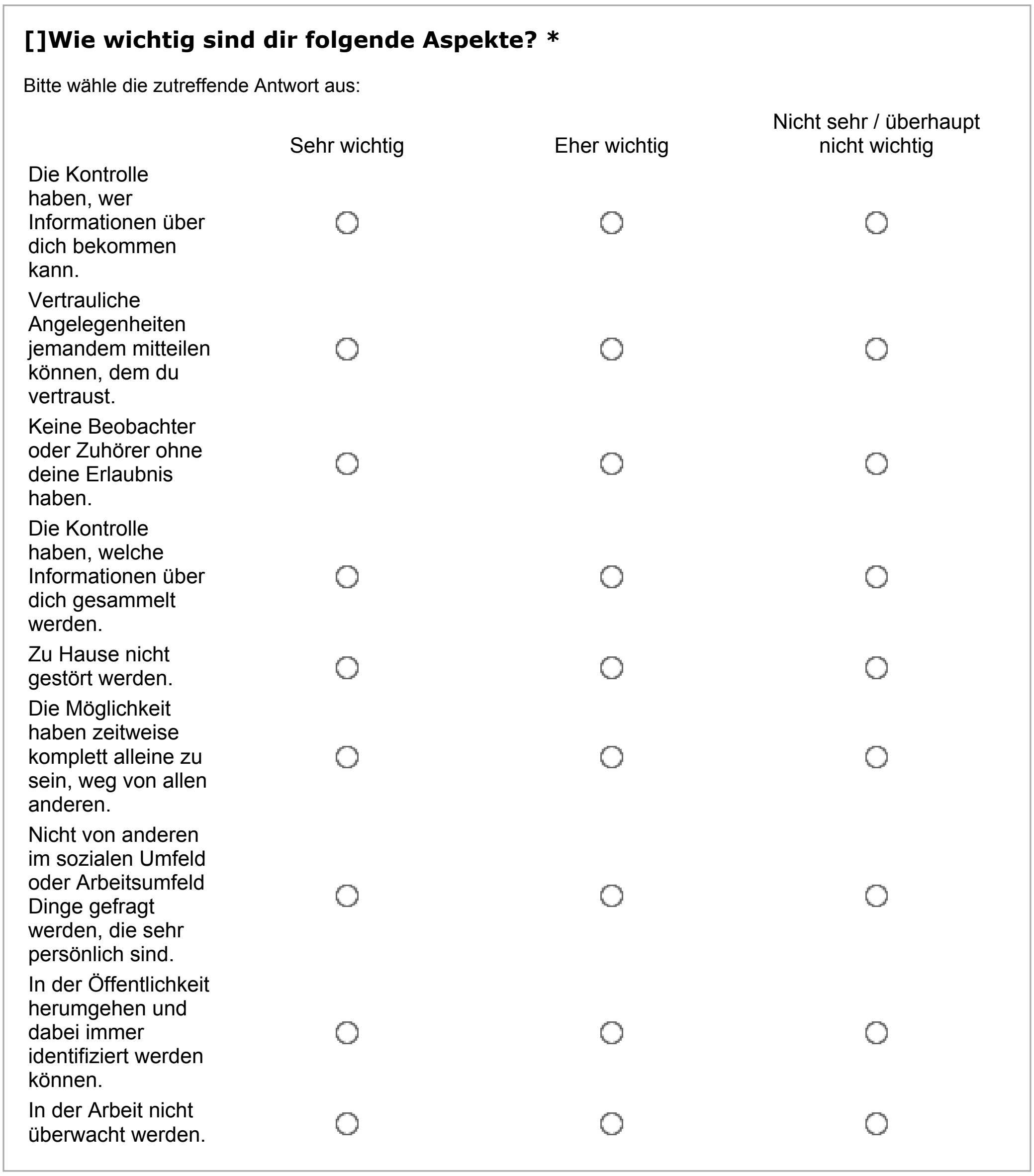
[]Kurzer Zugriff auf dein Smartphone:

Ich finde es besorgniserregend, wenn... *

Bitte wähle die zutreffende Antwort aus:

	Trifft zu				Trifft nicht zu
eine mir **unbekannte Person** kurzen Zugriff durch SnapApp auf mein Smartphone hat.	○	○	○	○	○
ein **Kollege** kurzen Zugriff durch SnapApp auf mein Smartphone hat.	○	○	○	○	○
ein(e) **FreundIn** kurzen Zugriff durch SnapApp auf mein Smartphone hat.	○	○	○	○	○
mein(e) **PartnerIn** kurzen Zugriff durch SnapApp auf mein Smartphone hat.	○	○	○	○	○
ein **Familienmitglied** (Eltern, Geschwister) kurzen Zugriff durch SnapApp auf mein Smartphone hat.	○	○	○	○	○

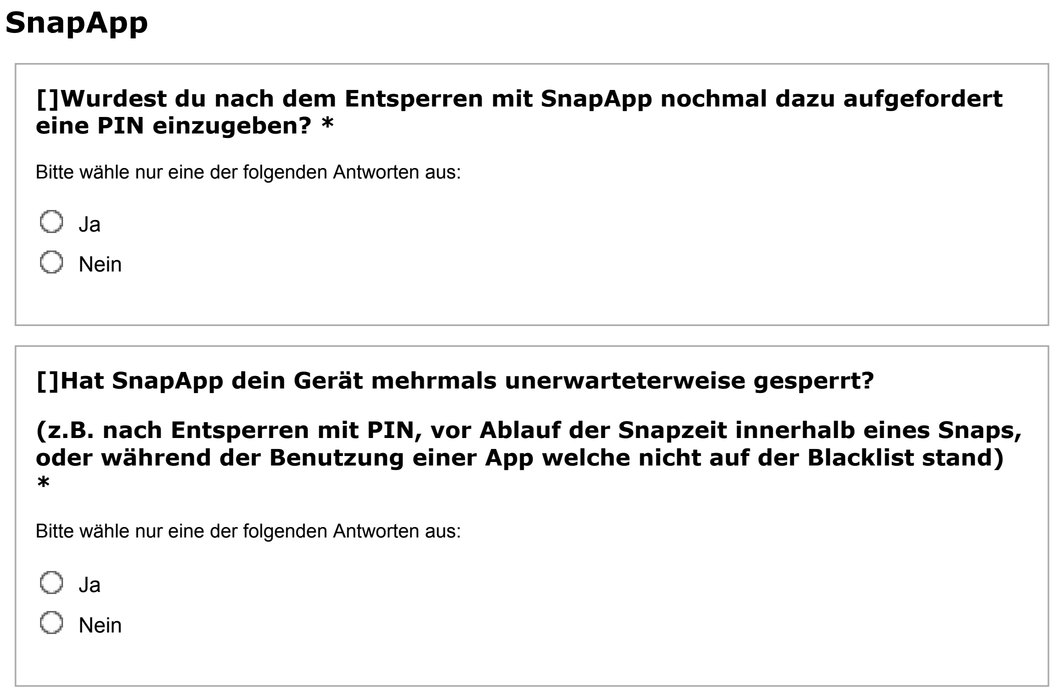
Privatsphäre



[]Wie wichtig sind dir folgende Aspekte? *

Bitte wähle die zutreffende Antwort aus:

	Sehr wichtig	Eher wichtig	Nicht sehr / überhaupt nicht wichtig
Die Kontrolle haben, wer Informationen über dich bekommen kann.	○	○	○
Vertrauliche Angelegenheiten jemandem mitteilen können, dem du vertraust.	○	○	○
Keine Beobachter oder Zuhörer ohne deine Erlaubnis haben.	○	○	○
Die Kontrolle haben, welche Informationen über dich gesammelt werden.	○	○	○
Zu Hause nicht gestört werden.	○	○	○
Die Möglichkeit haben zeitweise komplett alleine zu sein, weg von allen anderen.	○	○	○
Nicht von anderen im sozialen Umfeld oder Arbeitsumfeld Dinge gefragt werden, die sehr persönlich sind.	○	○	○
In der Öffentlichkeit herumgehen und dabei immer identifiziert werden können.	○	○	○
In der Arbeit nicht überwacht werden.	○	○	○



SnapApp

[]Wurdest du nach dem Entsperren mit SnapApp nochmal dazu aufgefordert eine PIN einzugeben? *

Bitte wähle nur eine der folgenden Antworten aus:

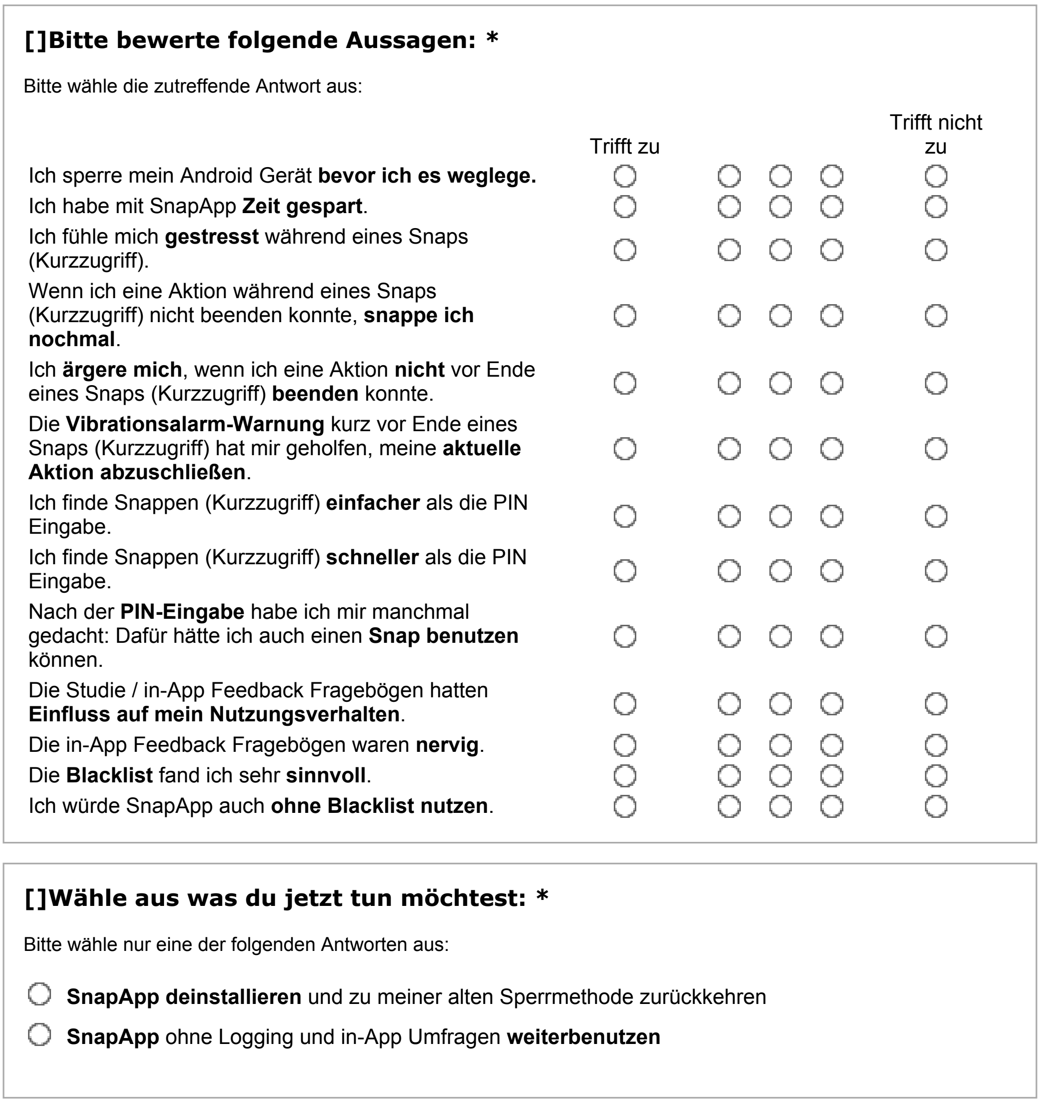
○ Ja
○ Nein

[]Hat SnapApp dein Gerät mehrmals unerwarteterweise gesperrt?

(z.B. nach Entsperren mit PIN, vor Ablauf der Snapzeit innerhalb eines Snaps, oder während der Benutzung einer App welche nicht auf der Blacklist stand) *

Bitte wähle nur eine der folgenden Antworten aus:

○ Ja
○ Nein



[]Bitte bewerte folgende Aussagen: *

Bitte wähle die zutreffende Antwort aus:

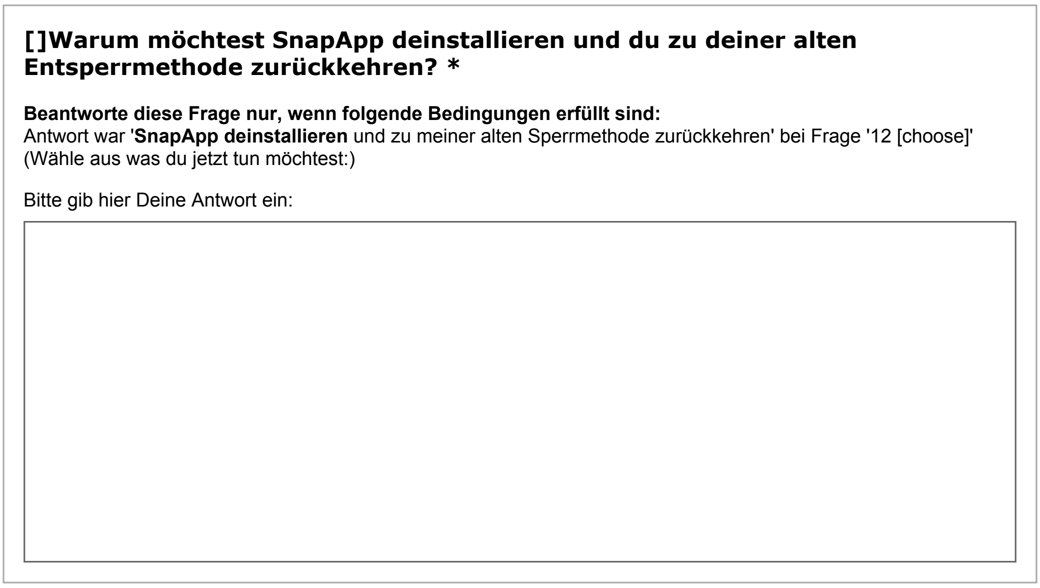
	Trifft zu				Trifft nicht zu
Ich sperre mein Android Gerät **bevor ich es weglege.**	○	○	○	○	○
Ich habe mit SnapApp **Zeit gespart**.	○	○	○	○	○
Ich fühle mich **gestresst** während eines Snaps (Kurzzugriff).	○	○	○	○	○
Wenn ich eine Aktion während eines Snaps (Kurzzugriff) nicht beenden konnte, **snappe ich nochmal**.	○	○	○	○	○
Ich **ärgere mich**, wenn ich eine Aktion **nicht** vor Ende eines Snaps (Kurzzugriff) **beenden** konnte.	○	○	○	○	○
Die **Vibrationsalarm-Warnung** kurz vor Ende eines Snaps (Kurzzugriff) hat mir geholfen, meine **aktuelle Aktion abzuschließen**.	○	○	○	○	○
Ich finde Snappen (Kurzzugriff) **einfacher** als die PIN Eingabe.	○	○	○	○	○
Ich finde Snappen (Kurzzugriff) **schneller** als die PIN Eingabe.	○	○	○	○	○
Nach der **PIN-Eingabe** habe ich mir manchmal gedacht: Dafür hätte ich auch einen **Snap benutzen** können.	○	○	○	○	○
Die Studie / in-App Feedback Fragebögen hatten **Einfluss auf mein Nutzungsverhalten**.	○	○	○	○	○
Die in-App Feedback Fragebögen waren **nervig**.	○	○	○	○	○
Die **Blacklist** fand ich sehr **sinnvoll**.	○	○	○	○	○
Ich würde SnapApp auch **ohne Blacklist nutzen**.	○	○	○	○	○

[]Wähle aus was du jetzt tun möchtest: *

Bitte wähle nur eine der folgenden Antworten aus:

○ **SnapApp deinstallieren** und zu meiner alten Sperrmethode zurückkehren

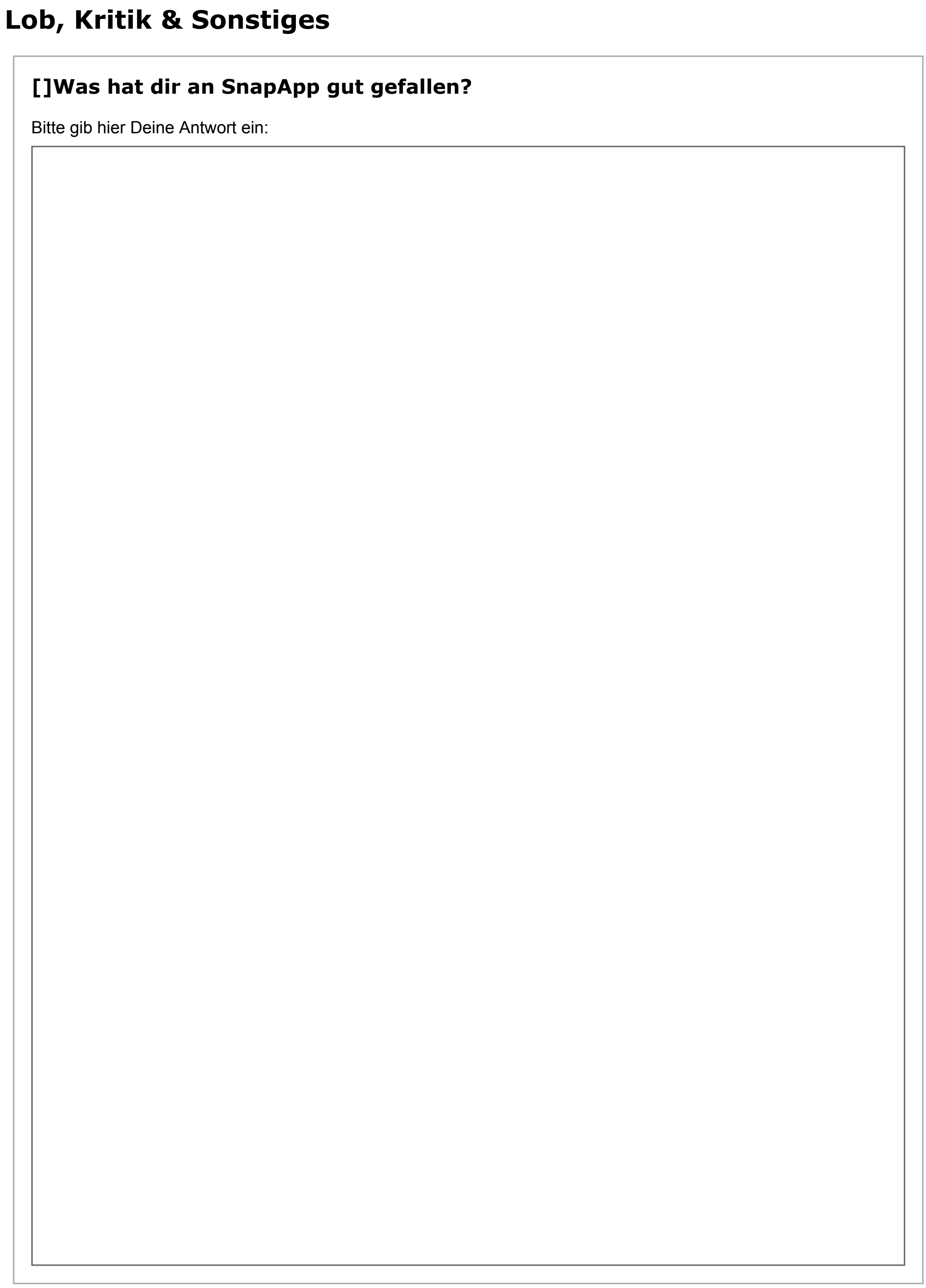
○ **SnapApp** ohne Logging und in-App Umfragen **weiterbenutzen**



[]Warum möchtest SnapApp deinstallieren und du zu deiner alten Entsperrmethode zurückkehren? *

Beantworte diese Frage nur, wenn folgende Bedingungen erfüllt sind:
Antwort war '**SnapApp deinstallieren** und zu meiner alten Sperrmethode zurückkehren' bei Frage '12 [choose]' (Wähle aus was du jetzt tun möchtest:)

Bitte gib hier Deine Antwort ein:

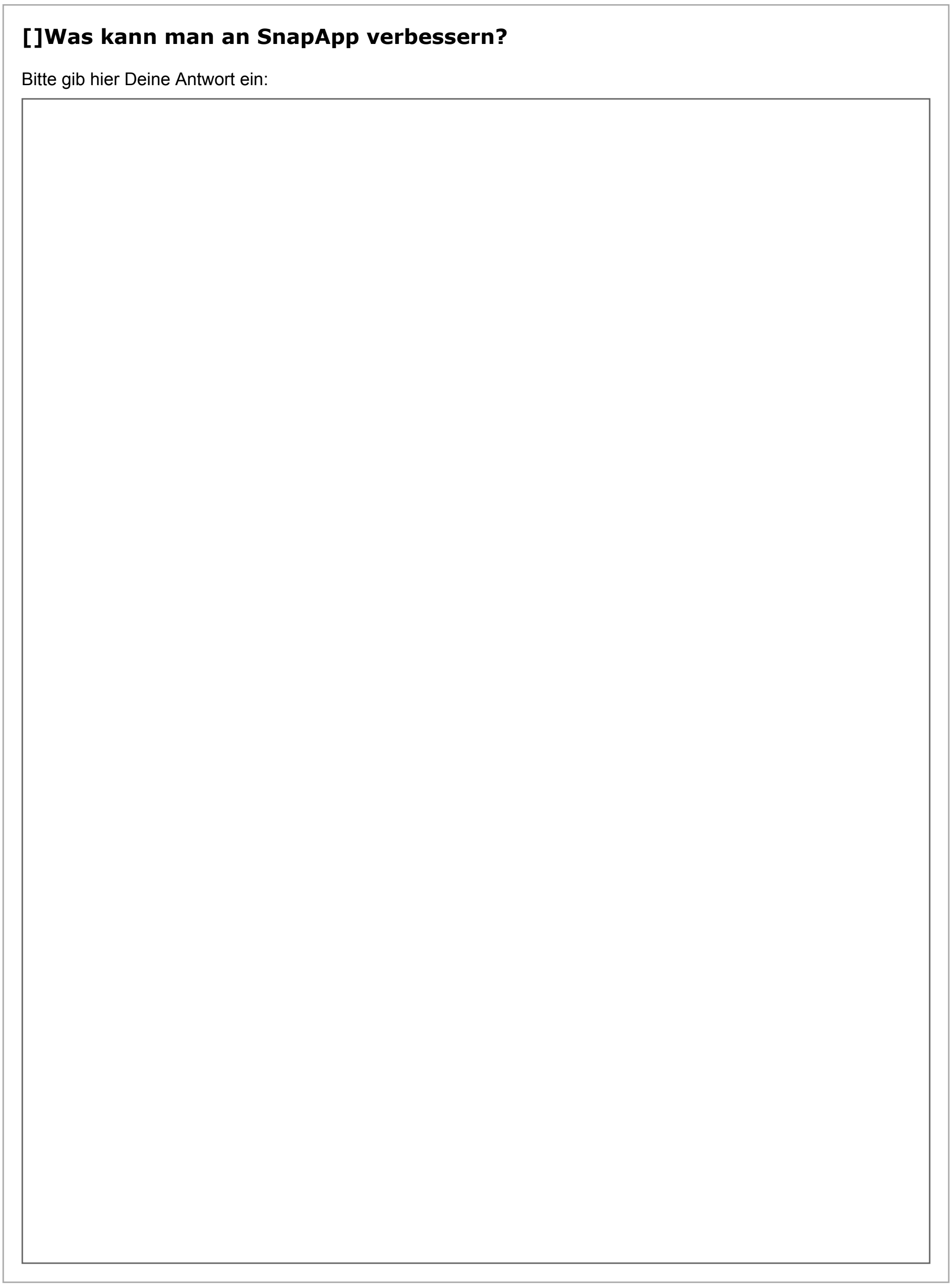


Lob, Kritik & Sonstiges

[]Was hat dir an SnapApp gut gefallen?

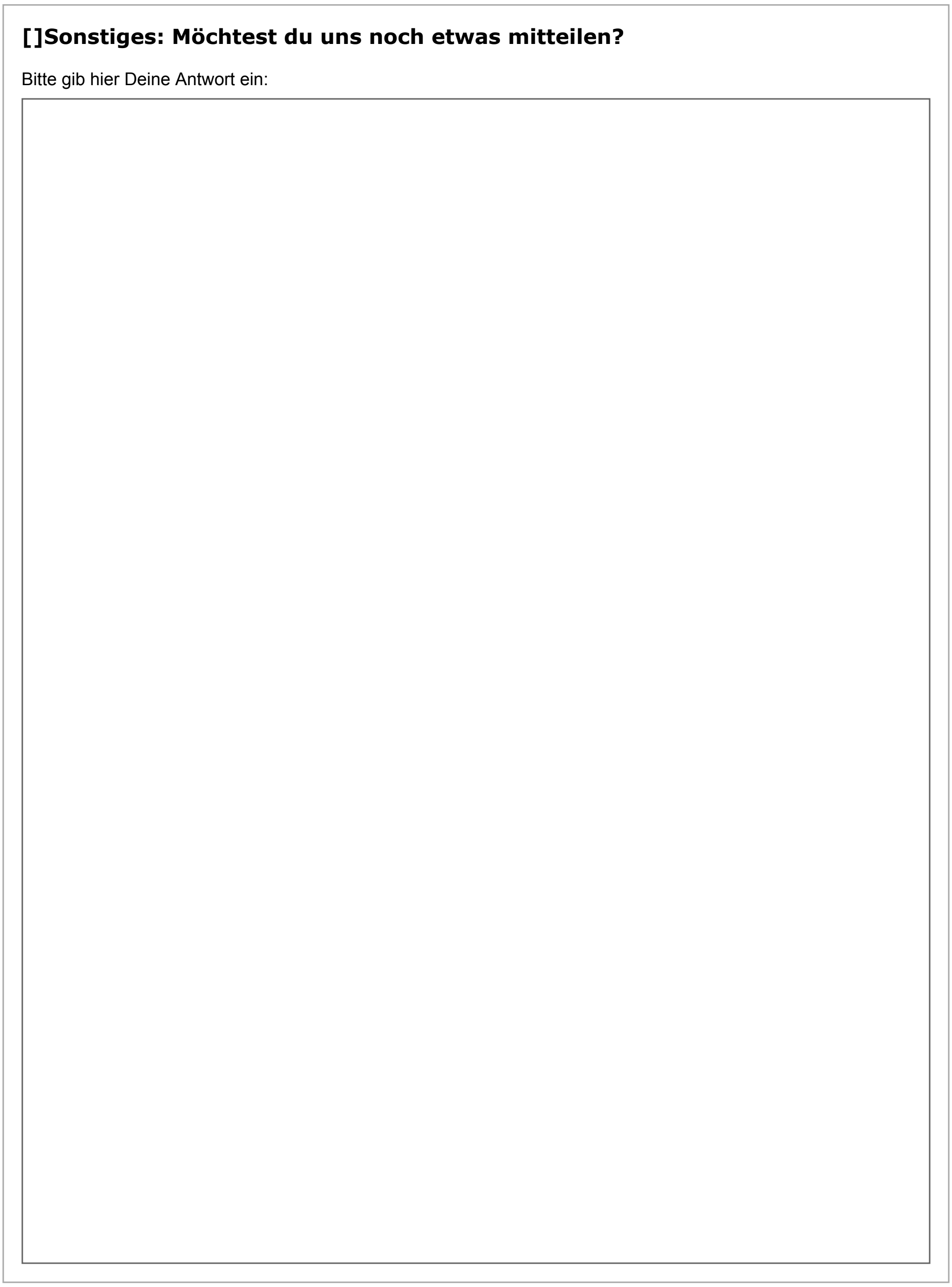
Bitte gib hier Deine Antwort ein:

D AFTER-STUDY QUESTIONNAIRE



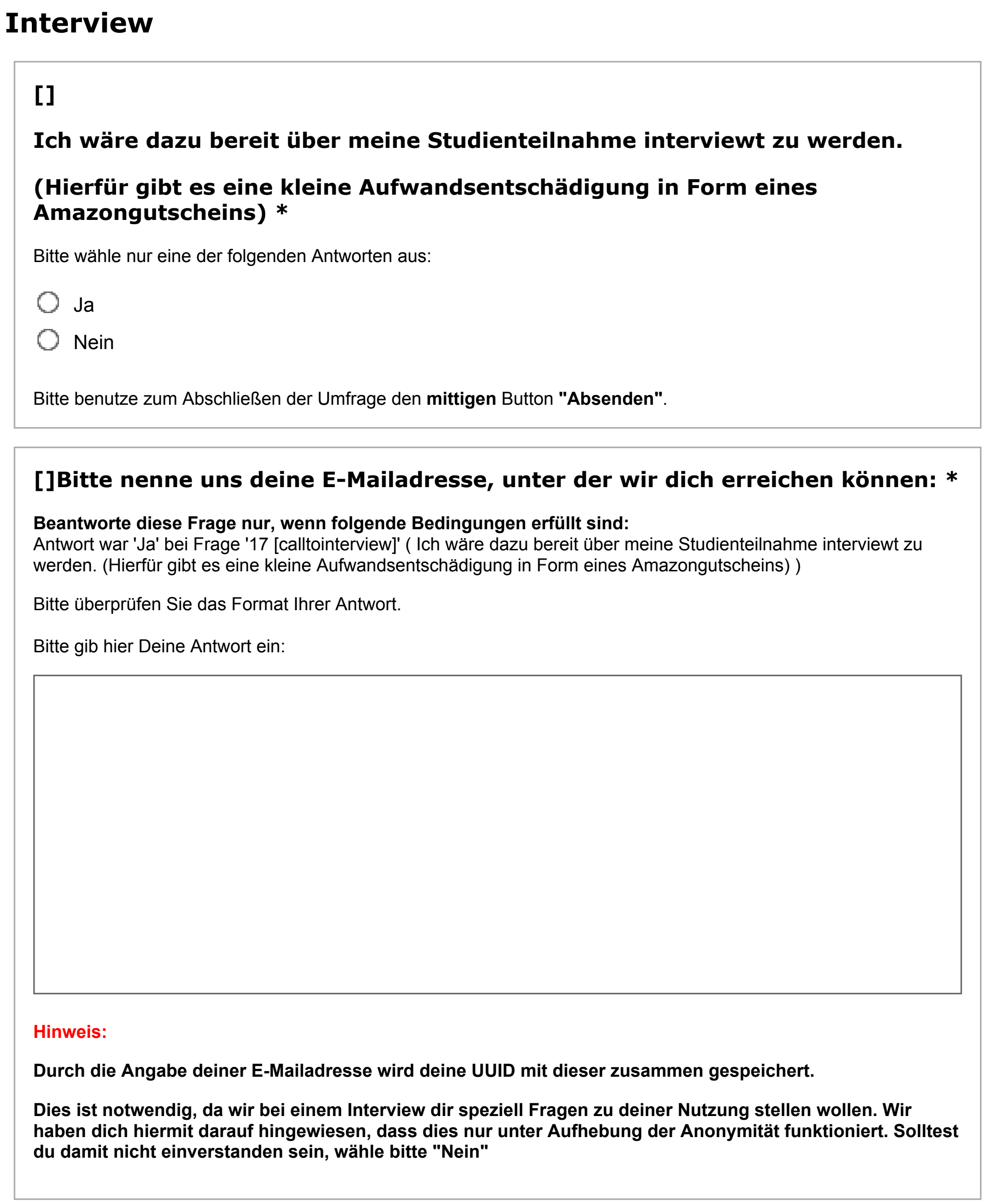
[]Was kann man an SnapApp verbessern?

Bitte gib hier Deine Antwort ein:



[]Sonstiges: Möchtest du uns noch etwas mitteilen?

Bitte gib hier Deine Antwort ein:



Interview

[]

Ich wäre dazu bereit über meine Studienteilnahme interviewt zu werden.

(Hierfür gibt es eine kleine Aufwandsentschädigung in Form eines Amazongutscheins) *

Bitte wähle nur eine der folgenden Antworten aus:

○ Ja
○ Nein

Bitte benutze zum Abschließen der Umfrage den **mittigen** Button **"Absenden"**.

[]Bitte nenne uns deine E-Mailadresse, unter der wir dich erreichen können: *

Beantworte diese Frage nur, wenn folgende Bedingungen erfüllt sind:
Antwort war 'Ja' bei Frage '17 [calltointerview]' (Ich wäre dazu bereit über meine Studienteilnahme interviewt zu werden. (Hierfür gibt es eine kleine Aufwandsentschädigung in Form eines Amazongutscheins))

Bitte überprüfen Sie das Format Ihrer Antwort.

Bitte gib hier Deine Antwort ein:

Hinweis:

Durch die Angabe deiner E-Mailadresse wird deine UUID mit dieser zusammen gespeichert.

Dies ist notwendig, da wir bei einem Interview dir speziell Fragen zu deiner Nutzung stellen wollen. Wir haben dich hiermit darauf hingewiesen, dass dies nur unter Aufhebung der Anonymität funktioniert. Solltest du damit nicht einverstanden sein, wähle bitte "Nein"

Contents of the CD

- Master thesis as PDF and Latex files
- All used figures
- Processed and raw logged usage data
- Questionnaires and questionnaire data
- Scripts for R and Python used for analysis of the logged data
- Source code of the SnapApp Android application prototype

References

[1] Apple Inc. Apple - Apple Pay. `https://www.apple.com/apple-pay/`, 2015. [Online; accessed 03-August-2015].

[2] Apple Inc. Apple - iPhone 6 - Touch ID. `https://www.apple.com/iphone-6/touch-id/`, 2015. [Online; accessed 03-August-2015].

[3] A. Arif, M. Pahud, K. Hinckley, and W. Buxton. A tap and gesture hybrid method for authenticating smartphone users. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 486–491, New York, NY, USA, 2013. ACM.

[4] A. S. Arif and A. Mazalek. Slide-to-unlock revisited: Two new user authentication techniques for touchscreen-based smartphones. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MOBIQUITOUS '14, pages 389–390, ICST, Brussels, Belgium, Belgium, 2014. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[5] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith. Smudge attacks on smartphone touch screens. In *Proceedings of the 4th USENIX Conference on Offensive Technologies*, WOOT'10, pages 1–7, Berkeley, CA, USA, 2010. USENIX Association.

[6] R. Biddle, S. Chiasson, and P. Van Oorschot. Graphical passwords: Learning from the first twelve years. *ACM Comput. Surv.*, 44(4):19:1–19:41, Sept. 2012.

[7] M. Böhmer, B. Hecht, J. Schöning, A. Krüger, and 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, pages 47–56, New York, NY, USA, 2011. ACM.

[8] H. Bojinov and D. Boneh. Mobile token-based authentication on a budget. In *Proceedings of the 12th Workshop on Mobile Computing Systems and Applications*, HotMobile '11, pages 14–19, New York, NY, USA, 2011. ACM.

[9] Chaos Computer Club e.V. Ccc | chaos computer club konkretisiert biometrie-debatte an schäubles fingerabdruck. `http://www.ccc.de/updates/2008/schaubles-finger`, 2008. [Online; accessed 02-August-2015].

[10] S. Consolvo, B. Harrison, I. Smith, M. Y. Chen, K. Everitt, J. Froehlich, and J. a. Landay. Conducting In Situ Evaluations for and With Ubiquitous Computing Technologies. *International Journal of Human-Computer Interaction*, 22(1-2):103–118, 2007.

[11] Consumer Reports. Cell phone security | wireless threats - consumer reports. `http://www.consumerreports.org/privacy0613`, June 2013.

[12] M. Conti, I. Zachia-Zlatea, and B. Crispo. Mind how you answer me!: Transparently authenticating the user of a smartphone when answering or placing a call. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '11, pages 249–259, New York, NY, USA, 2011. ACM.

[13] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann. Touch me once and i know it's you!: Implicit authentication based on touch screen patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 987–996, New York, NY, USA, 2012. ACM.

[14] A. De Luca, E. von Zezschwitz, N. D. H. Nguyen, M.-E. Maurer, E. Rubegni, M. P. Scipioni, and M. Langheinrich. Back-of-device authentication on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2389–2398, New York, NY, USA, 2013. ACM.

[15] S. Egelman, S. Jain, R. S. Portnoff, K. Liao, S. Consolvo, and D. Wagner. Are you ready to lock? In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 750–761, New York, NY, USA, 2014. ACM.

[16] Etherington, Darrell. Androids Smart On-Body Unlock Function Could Eliminate A Long-time Headache. `http://on.tcrn.ch/l/SNH4`, 2015. [Online; accessed 02-August-2015].

[17] S. Flügge, H. Scharf, S. Fahl, and M. Smith. Poster: Preliminary investigation of an nfc-unlock mechanism for android. 2013.

[18] H. Gascon, S. Uellenbeck, C. Wolf, and K. Rieck. Continuous Authentication on Mobile Devices by Analysis of Typing Motion Behavior. *Sicherheit*, 2014.

[19] Google Inc. Set up your device for automatic unlock - nexus help. `https://support.google.com/nexus/answer/6093922?hl=en`, 2015. [Online; accessed 03-August-2015].

[20] A. Gupta, M. Miettinen, N. Asokan, and M. Nagy. Intuitive security policy configuration in mobile devices using context profiling. In *Proceedings of the 2012 ASE/IEEE International Conference on Social Computing and 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust*, SOCIALCOM-PASSAT '12, pages 471–480, Washington, DC, USA, 2012. IEEE Computer Society.

[21] M. Harbach, E. von Zezschwitz, A. Fichtner, A. D. Luca, and M. Smith. It's a hard lock life: A field study of smartphone (un)locking behavior and risk perception. In *Symposium On Usable Privacy and Security (SOUPS 2014)*, pages 213–230, Menlo Park, CA, July 2014. USENIX Association.

[22] E. Hayashi, S. Das, S. Amini, J. Hong, and I. Oakley. Casa: Context-aware scalable authentication. In *Proceedings of the Ninth Symposium on Usable Privacy and Security*, SOUPS '13, pages 3:1–3:10, New York, NY, USA, 2013. ACM.

[23] E. Hayashi, O. Riva, K. Strauss, A. J. B. Brush, and S. Schechter. Goldilocks and the two mobile devices: Going beyond all-or-nothing access to a device's applications. In *Proceedings of the Eighth Symposium on Usable Privacy and Security*, SOUPS '12, pages 2:1–2:11, New York, NY, USA, 2012. ACM.

[24] R. M. Hogarth, M. Portell, and A. Cuxart. What risks do people perceive in everyday life? A perspective gained from the experience sampling method (ESM). *Risk Analysis*, 27(6):1427–1439, 2007.

[25] F. Hong, M. Wei, S. You, Y. Feng, and Z. Guo. Waving authentication: Your smartphone authenticate you on motion gesture. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pages 263–266, New York, NY, USA, 2015. ACM.

[26] S. S. Intille, J. Rondoni, C. Kukla, I. Ancona, and L. Bao. A context-aware experience sampling tool. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '03, pages 972–973, New York, NY, USA, 2003. ACM.

[27] M. Jakobsson, E. Shi, P. Golle, and R. Chow. Implicit authentication for mobile devices. In *Proceedings of the 4th USENIX Conference on Hot Topics in Security*, HotSec'09, pages 9–9, Berkeley, CA, USA, 2009. USENIX Association.

[28] A. K. Karlson, A. B. Brush, and S. Schechter. Can i borrow your phone?: Understanding concerns when sharing mobile phones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 1647–1650, New York, NY, USA, 2009. ACM.

[29] H. G. Kayacik, M. Just, L. Baillie, D. Aspinall, and N. Micallef. Data driven authentication: On the effectiveness of user behaviour modelling with mobile device sensors. *arXiv preprint arXiv:1410.7743*, 2014.

[30] H. Khan and U. Hengartner. Towards application-centric implicit authentication on smartphones. In *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*, HotMobile '14, pages 10:1–10:6, New York, NY, USA, 2014. ACM.

[31] S. Kujala and T. Miron-Shatz. Emotions, experiences and usability in real-life mobile phone use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1061–1070, New York, NY, USA, 2013. ACM.

[32] S. Kurkovsky and E. Syta. Digital natives and mobile phones: A survey of practices and attitudes about privacy and security. In *Technology and Society (ISTAS), 2010 IEEE International Symposium on*, pages 441–449, June 2010.

[33] T. Kwon and S. Na. TinyLock: Affordable defense against smudge attacks on smartphone pattern lock systems. *Computers and Security*, 42:137–150, 2014.

[34] Lardinois, Frederic. Google Launches Android M Preview With Fingerprint Scanner Support, Android Pay, Improved Permissions And Battery Life. `http://on.tcrn.ch/l/9ej5`, 2015. [Online; accessed 01-August-2015].

[35] Mads Haahr. Random.org - true random number service. `https://www.random.org/lists/`, 2015. [Online; accessed 03-August-2015].

[36] J. Maguire and K. Renaud. You only live twice or "the years we wasted caring about shoulder-surfing". In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers*, BCS-HCI '12, pages 404–409, Swinton, UK, UK, 2012. British Computer Society.

[37] N. Micallef, M. Just, L. Baillie, M. Halvey, and H. G. Kayacik. Why arent users using protection? investigating the usability of smartphone locking. MobileHCI, 2015.

[38] N. Micallef, M. Just, L. Baillie, and G. Kayacik. Poster : Towards an app-driven mobile authentication model. pages 1–2, 2013.

[39] A. Möller, M. Kranz, B. Schmid, L. Roalter, and S. Diewald. Investigating self-reporting behavior in long-term studies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2931–2940, New York, NY, USA, 2013. ACM.

[40] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, and K. Beznosov. Understanding users' requirements for data protection in smartphones. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering Workshops*, ICDEW '12, pages 228–235, Washington, DC, USA, 2012. IEEE Computer Society.

[41] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, and K. Beznosov. Know your enemy: The risk of unauthorized access in smartphones by insiders. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 271–280, New York, NY, USA, 2013. ACM.

[42] X. Ni, Z. Yang, X. Bai, A. Champion, and D. Xuan. Diffuser: Differentiated user access control on smartphones. In *Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on*, pages 1012–1017, Oct 2009.

[43] C. Nickel. Accelerometer-based biometric gait recognition for authentication on smartphones. 2012.

[44] D. Norman. When Security Gets in the Way. *Interactions*, 2010.

[45] PayPal (Europe) S.à r.l. et Cie, S.C.A. Use your fingerprint and paypal to shop. `https://www.paypal-pages.com/samsunggalaxys5/us/index.html`, 2015. [Online; accessed 03-August-2015].

[46] P. J. Phillips. Improving face recognition technology. *Computer*, 44(3):84–86, Mar. 2011.

[47] A. P. Pons and P. Polak. Understanding user perspectives on biometric technology. *Commun. ACM*, 51(9):115–118, Sept. 2008.

[48] O. Riva, C. Qin, and K. Strauss. Progressive authentication: deciding when to authenticate on mobile phones. *Proceedings of the 21 st ...*, pages 1–16, 2011.

[49] F. Schaub, R. Deyhle, and M. Weber. Password entry usability and shoulder surfing susceptibility on different smartphone platforms. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, MUM '12, pages 13:1–13:10, New York, NY, USA, 2012. ACM.

[50] Srilenkha R. and Jayakumar D. A Secure Screen Lock System for Android Smart Phones using Accelerometer Sensor. 1(10):96–100, 2015.

[51] F. Stajano. Will your digital butlers betray you? In *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*, WPES '04, pages 37–38, New York, NY, USA, 2004. ACM.

[52] F. Stajano. One user, many hats; and, sometimes, no hat: Towards a secure yet usable pda. In B. Christianson, B. Crispo, J. Malcolm, and M. Roe, editors, *Security Protocols*, volume 3957 of *Lecture Notes in Computer Science*, pages 51–64. Springer Berlin Heidelberg, 2006.

[53] L. Staneková and M. Stanek. How to choose a PIN - assessment of dictionary methods. 2013.

[54] M. Tamviruzzaman, S. I. Ahamed, C. S. Hasan, and C. O'brien. epet: When cellular phone learns to recognize its owner. In *Proceedings of the 2Nd ACM Workshop on Assurable and Usable Security Configuration*, SafeConfig '09, pages 13–18, New York, NY, USA, 2009. ACM.

[55] F. Tari, A. A. Ozok, and S. H. Holden. A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. In *Proceedings of the Second Symposium on Usable Privacy and Security*, SOUPS '06, pages 56–66, New York, NY, USA, 2006. ACM.

[56] H. Taylor. Most people are 'privacy pragmatists' who, while concerned about privacy, will sometimes trade it off for other benefits. *The Harris Poll*, 17(19):1–6, 2003.

[57] S. Uellenbeck, T. Hupperich, C. Wolf, T. Holz, G. Horst, S. Uellenbeck, and T. Hupperich. Tactile One-Time Pad: Smartphone Authentication Resilient Against Shoulder Surfing. 2014.

[58] D. Van Bruggen, S. Liu, M. Kajzer, A. Striegel, C. R. Crowell, and J. D'Arcy. Modifying smartphone user locking behavior. In *Proceedings of the Ninth Symposium on Usable Privacy and Security*, SOUPS '13, pages 10:1–10:14, New York, NY, USA, 2013. ACM.

[59] E. von Zezschwitz, A. De Luca, B. Brunkow, and H. Hussmann. Swipin: Fast and secure pin-entry on smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1403–1406, New York, NY, USA, 2015. ACM.

[60] E. von Zezschwitz, P. Dunphy, and A. De Luca. Patterns in the wild: a field study of the usability of pattern and pin-based authentication on mobile devices. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*, MobileHCI '13, pages 261–270, New York, NY, USA, 2013. ACM.

[61] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma. Toward a practical face recognition system: Robust alignment and illumination by sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(2):372–386, Feb. 2012.

[62] M. Wilson and L. Chen. Securing Sensitive Data Stored on Smartphones Using Face Recognition to Unlock Mobile Devices. 2012.

[63] M. Wobig. Entsperren von Android Smartphones mit Hilfe von Near Field Communication. (August), 2012.