Explanation Beats Context: The Effect of Timing & Rationales on Users' Runtime Permission Decisions

Yusra Elbitar[§*], Michael Schilling[§], Trung Tin Nguyen[§*], Michael Backes[§], Sven Bugiel[§]
[§] CISPA Helmholtz Center for Information Security, [*] Saarland University

Abstract

Current mobile platforms leave it up to the app developer to decide when to request permissions *(timing)* and whether to provide explanations why and how users' private data are accessed *(rationales)*. Given these liberties, it is important to understand how developers should use timing and rationales to effectively assist users in their permission decisions. While guidelines and recommendations for developers exist, no study has systematically investigated the actual influence of timing, rationales, and their combinations on users' decision-making process. In this work, we conducted a comparative online study with 473 participants who were asked to interact with mockup apps drawn from a pool of 120 variations of 30 apps. The study design was guided by developers' current permission request practices derived from a dynamic analysis of the top apps on *Google Play*. Our results show that there is a clear interplay between timing and rationales on users' permission decisions and the evaluation of their decisions, making the effect of rationales stronger when shown upfront and limiting the effect of timing when rationales are present. We therefore suggest adaptation to the available guidelines. We also find that permission decisions depend on the individuality of users, indicating that there is no one-fits-all permission request strategy, upon we suggest better individual support and outline one possible solution.

1 Introduction

Mobile platforms such as Android and iOS handle some of users' most private data, can precisely record information using available sensors, and are "always on". To keep users in control, these platforms make it possible for users to delegate access rights (permissions) to apps. As such, the user decides *which* app is granted which permissions, while it is up to the app developer to decide *when* to ask the user for permission and whether to provide an explanation as to *why and how* data is accessed. The timing of permission requests, along with the accompanying explanations or "rationales", form a one-way communication channel from developers to users. This channel conveys information meant to help users make informed permission decisions which reflect their individual values and privacy preferences in a given context.

Prior work [1] as well as current Google guidelines [2] contain recommendations for developers about when and how permissions should be requested. Although the available advice seems straightforward, there is not enough scientific evidence to thoroughly support it. We unfortunately do not know how timing, rationales, and their combinations affect users' decisions, which strategies in asking for permissions help users the most, and whether those guidelines agree with users' preferences. In the literature, a large body of work has focused on understanding the reasons behind users' permission decisions [3–9], but all those prior studies have been conducted either on the obsolete install-time permission model or on the current permission model but without considering the different variations depending on timing and rationales within the model itself. Other researchers studied the isolated effect of rationales on users' permission decisions [10] or developers' current rationale practices [10, 11]. Prior works that considered both timing and rationales only reported the status quo of developers' current permission request practices [12]. This leaves a gap in the understanding of the effects and interactions of these variables on users' decisions and whether these decisions mirror the individual interests of users.

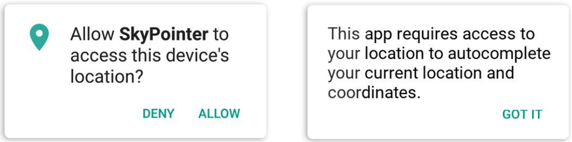
In this work, we will focus on how timing (upfront/in-context) and rationales (presence/absence) affect users' perception of their decision as well as how developers can use these factors to best support users in deciding whether to grant permissions. To answer those questions, we conducted the first analysis (to the best of our knowledge) of the communication channel for permission requests between the app developer and the user from both perspectives. We started by dynamically analyzing the top apps on Google Play to explore how developers currently request permissions at app runtime (Section 4). During this first step, we captured over 2,500 dangerous permission requests. Based on those findings, we then designed and conducted a comparative user study with

473 participants from Amazon MTurk to investigate the effect of timing, presence/absence of rationales, and their interactions on users' permission decisions (Section 5). Our study focused on one standardized rationale design and wording which was informed by the empirical analysis. To ensure generalizability, we provided participants with realistic settings by using a total of 30 interactive mockup apps. We created four versions of each app to request a permission for each possible combination of timing and rationales (i.e., upfront with and without rationale, and in context with and without rationale). Throughout our study we collected answers to around 1,800 permission requests which capture participants' permission decision, their perception of having made an informed decision, their satisfaction with the decision, their perceived control over the decision, and how clearly they understood the purpose of the requested permission.

Our results (Section 6) indicate a mutual interplay between the timing of permission requests and rationales. Overall, we found that rationales increase grant rates and have a positive effect on users' perception of their decisions. However, this effect is stronger when rationales are added upfront rather than in context. As for timing, on one hand, asking for permissions in context has a positive effect on users' perception when no rationales are present. On the other hand, requesting permissions in context always has a positive effect on grant rates, regardless of the presence of rationales. Based on these findings (Section 7), we suggest the adaptation of Google's current guidelines [2] to better support users in their decision-making process. Going beyond these aspects, however, we also found that permission decisions depend on individual differences between users. As a consequence, we argue that there is no one-size-fits-all permission request strategy. Therefore, current mobile platforms could benefit from built-in support for users to customize permission requests. This could be realised through a system setting that would enable users to configure when they would like to see permission requests and whether they prefer to see rationales.

2 Background

Apps run in a limited-access sandbox and need permissions for certain features (e.g., camera and microphone) and user's private data (e.g., contacts and location). In previous versions of Android, permissions were requested at app installation time, meaning that users could either grant all requested permissions or abort the installation process. In Oct. 2015, Android 6.0 introduced the runtime permission model, where dangerous permissions (i.e., permissions that protect sensitive data or functionalities) are requested at runtime. Under this model, similar permissions are grouped together (e.g., `Read_Contacts` and `Write_Contacts` belong to the `CONTACTS` group). To request a permission, the developer uses the `requestPermissions()` API which the user sees as a system dialog of the requested permission group (Figure 1a).



Allow **SkyPointer** to access this device's location?

This app requires access to your location to autocomplete your current location and coordinates.

(a) Permission request (b) Sample rationale

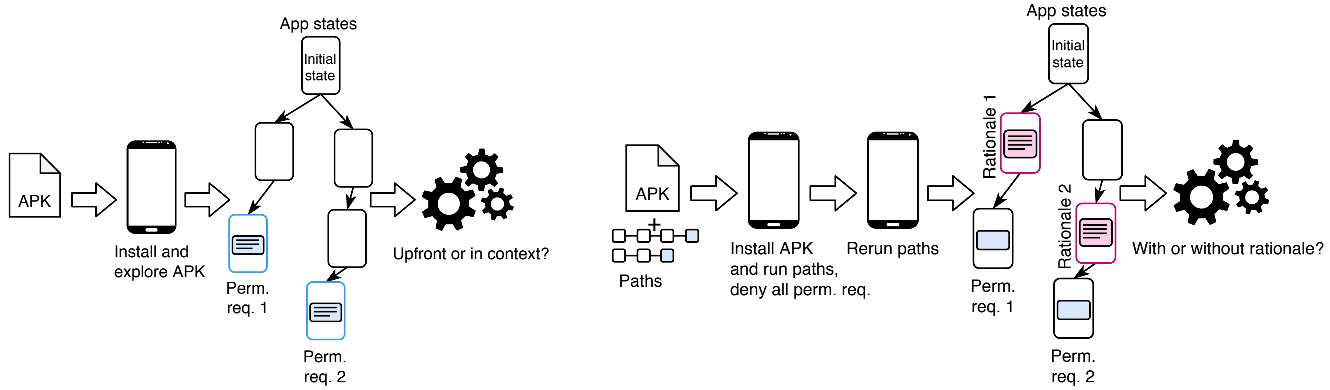
Figure 1: Android's permission dialog and a sample rationale.

The runtime permission model encourages developers to help users understand why an app requires certain permissions. Developers can decide when a permission is requested and if they want to provide rationales, thereby implicitly opening a one-way communication channel with the users to inform them about the intentions of the app. The request can either be made upfront at app launch or in context, when the app accesses the protected resource. As for rationales, developers are free to choose the design and wording of rationales (e.g., Figure 1b). A rationale can either be provided before or after a permission request, or only after the request has been denied using the `shouldShowRequestPermissionRationale()` API. In the light of these liberties, it is essential to understand the effect of timing and presence/absence of rationales on users' decision-making process to better support developers in requesting permissions.

3 Related Work

Both developers and users are an essential part of the runtime permission model. On one hand, developers provide information about permission requests through context and rationales in hope of permission approvals that are necessary for the intended functionalities of their apps. On the other hand, users utilize the provided information to make an informed decision in accordance with their individual preferences. Unfortunately, users are often not able to make informed decisions because they do not understand the requested permissions, their purpose, and the risks involved with granting them [7, 13–15]. Consequently, their expectations are often violated [4].

As a solution, prior research suggested providing rationales to clarify why the requested permission is needed by the app [16–22]. Tools using automated procedures to extract this information were created to help developers who might forget to explain all permission usages or are not aware of all usages (e.g., due to 3rd party code) [8,23,24]. Additionally, the status quo of rationales revealed that only a small portion of apps provide rationales [10, 11], and if provided they do not communicate useful information, except that a specific permission is required [11]. Based on these findings, the challenge is to help developers create meaningful rationales [25], which is orthogonal to understanding the effect of the presence/absence of rationales on users' permission decisions.



(a) Step 1 identifies timing of permission requests: A set of heuristics are applied on extracted paths, consisting of a list of app-states from start to each permission request.

(b) Step 2 identifies rationales: Extracted paths are rerun twice, in the first run all permissions are denied, in the second run rationales are extracted.

Figure 2: Steps of the empirical analysis

Other tools to support developers include solutions to automatically migrate install-time permission requests to the runtime permission model [12, 26], or guidelines on how permissions should generally be requested to minimize the burden on users [1]. Recent work also developed a tool that warns developers if their requested permissions are unlikely to be requested by similar apps [27].

To reduce the burden on users, previous work suggested to predict users' permission decisions [5] based on a set of privacy profiles [5, 6, 28–30] or to provide them with privacy nudges [9, 31]. Researchers also proposed a permission manager that would allow users a fine-grained permission control [32]. This line of work considers the current permission model as inadequate or incomplete and takes a more radical approach to aid users in their permission decisions. However, these changes need to be adopted by system vendors.

The reasons why users grant or deny permissions has received considerable attention in research. It was shown that users' decisions often depend on the functionality associated with the permission [3–7], the perceived permission sensitivity [3, 4, 8], the user's prior privacy experience [9] and privacy concerns [3]. We considered all these factors as control variables in our study with the aim of extending previous work.

4 Empirical Analysis

We conducted an empirical analysis of rationales and timing of permission requests in the top apps from Google Play. The main goal of this analysis was to provide a valid foundation for the standardized rationale design and select the apps for the user study (see Sections 5.5 and 5.6 for more details). Our crawler collected the top 100 free apps in each category from Play (Dec. 2018–June 2019). We expected to find a representative sample of apps using runtime permission requests, since we conducted the analysis three years after the runtime permission model was introduced (with the release of Android 6 in Oct. 2015) and one month after this model became mandatory for all new apps and app updates [33]. The top 100 apps varied during the 7-months long crawling period. We therefore collected more than 200,000 unique apps.

Our initial approach to detect timing of permission requests and rationales was to use static analysis. However, we discovered that this approach cannot provide reliable information about the exact position of permission requests in the GUI control-flow. Thus, we used static analysis only to reduce the number of apps that will be subjected to dynamic analysis by filtering out all apps that do not request dangerous permissions in their manifest and do not call the `requestPermissions()` API. We also removed non-English as well as game-related apps. From the resulting set of 12,794 apps, we then randomly selected 10,000 apps for further analysis.

4.1 Classification of Permission Requests

For the dynamic analysis we extended DroidBot [34], a lightweight test input generator for Android apps. In two analysis steps, we determined the timing of permission requests (step 1) and the presence of rationales (step 2).

Identify timing (step 1): This step occupied most of the dynamic analysis time (~30–60 min per app). As shown in Figure 2a, we first installed and launched the app of interest. Then we waited around 60 seconds before exploring the app. This step was important to correctly identify upfront permission requests that would otherwise have been categorized as in-context because some apps take time to load (e.g., using a splash screen). The output of the dynamic analysis was the shortest path to all permission requests found. Each path consisted of a list of states from app launch to the permission request of interest, on which we applied a set of heuristics to identify the timing. For example, if the permission request appeared without clicking on some UI element, we considered the timing upfront.

Identify rationales (step 2): To also find rationales that were only displayed after a permission has been denied, we first reinstalled the app, followed each permission request path from step 1, and denied all requests (as shown in Figure 2b). Then, we ran each path again and collected the resulting app states, possibly with new rationale messages. To extract these messages, we used rationales that were obtained with a CNN classifier by previous work [11] in a Latent Semantic Analysis (LSA) to group similar rationales under one topic. These topics were then used in a semantic similarity analysis [35] that assigned a score to each sentence in the permission request path. All sentences that were at least 40% similar to a rationale topic were then manually verified as rationales. We used the evaluation of 100 randomly selected permission requests (50 categorized with rationale and 50 without) as a benchmark to evaluate this threshold. The classification of this subset had a precision of 94% and a recall of 100%.

From our initial app set, we successfully analyzed 7,998 apps and found 2,071 apps that requested at least one dangerous permission at runtime (total of 2,569 permission requests). Upon closer inspection, we found that part of this discrepancy was due to the fact that many apps included the `requestPermissions()` API in third-party library code that was never executed, what meant that we spent time dynamically analyzing apps that did not actually request permissions at runtime. Further, low code coverage of dynamic analysis (e.g., through login-forms) is a known limitation of available analysis tools, which prevented us from reaching all permission requests. Nevertheless, we collected an adequate number of rationales that were used in the selection process of the standardized rationale for the user study.

4.2 Findings

As the results are biased towards upfront permission requests, we should consider them with reservation. Nevertheless, we reveal different ways of showing rationales in terms of design, quality, wording, and timing.

Timing and presence/absence of rationales. Of the 2,569 found permission requests, 70% were displayed upfront and 16% showed rationales that were evenly distributed among upfront and in-context requests. The most frequently requested permission was `STORAGE` (56% of 2,569) followed by `LOCATION` (19%), `CAMERA` (9%), `PHONE` (6%), `CONTACTS` (3%), and `MICROPHONE` (3%). We only found a small number of permission requests for `SMS`, `CALENDAR`, and `PHONELOG`, which is consistent with prior work [3,36]. A chi-square test of independence was performed to examine the relation between timing and permission type. Due to too few observations we excluded the permissions `SMS`, `CALENDAR`, and `PHONELOG` from the analysis. We found that the proportion of in-context permission requests significantly differed between permissions ($X^2(5) = 49.562$, $p < 0.001$, $Cramer's V = 0.139$). For example, the highest proportion of in-context requests was found for `STORAGE` (34%), closely followed by `MICROPHONE` (33%), and `CAMERA` (32%). While the lowest proportion was seen for `LOCATION` (23%) and `PHONE` (12%), which are often associated with background functionalities and are therefore most frequently requested upfront. Whereas, there was no significant association between permission type and presence/absence of rationales ($X^2(5) = 8.06$, $p = 0.153$, $Cramer's V = 0.056$).

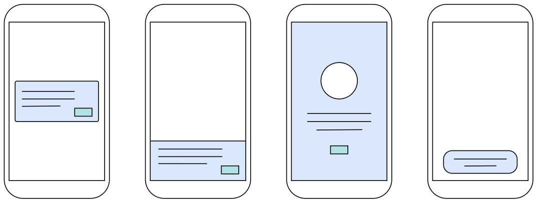

(a) Dialog (b) Banner (c) Fullscreen (d) Snackbar

Figure 3: The different rationale designs.

Design and wording of rationales. We found four general design patterns for rationales. They were displayed as either dialogs, fullscreen views, banners, or snackbars (as highlighted in Figure 3). Each design pattern was shown before a permission request or after a permission denial, except for snackbars which were only used after a permission was denied. Additionally, each design provided rationales for one or multiple permissions. We also noticed that most rationales provided an acknowledge button (e.g., ok, got it, proceed), while around half of the dialogs additionally included a cancel button (e.g., cancel, exit, not now, skip). The fullscreen views had the most design variations, compared to the other options, which mostly used the default Android layout.

As for the content, rationales either provided more information compared to the default permission request dialog (i.e., reasons why the app needs the permission and how it will be used) or they just signified that some permission is required or has been denied (e.g., this app requires this permission: to work perfectly, run normally, function properly). We found that about 50% of the rationales provided additional information, thus fulfilling the true purpose of rationales.

5 User Study

The aim of this user study is to assess whether there is an effect of timing and presence/absence of rationales on users' permission decisions. To isolate these effects, we used the findings from the empirical analysis to define a standardized rationale that also explains how and why a permission is needed (providing additional information). More precisely, we want to answer the following questions: How does the

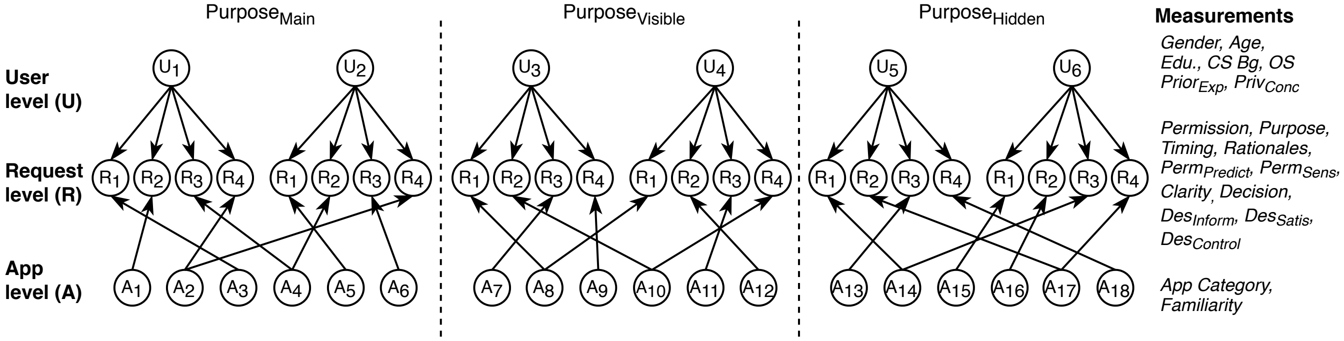

$\text{Purpose}_{\text{Main}}$
$\text{Purpose}_{\text{Visible}}$
$\text{Purpose}_{\text{Hidden}}$
Measurements
User level (U)
Request level (R)
App level (A)
U_1 U_2 U_3 U_4 U_5 U_6
R_1 R_2 R_3 R_4
A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_8 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15} A_{16} A_{17} A_{18}
Gender, Age, Edu., CS Bg, OS $\text{Prior}_{\text{Exp}}$, $\text{Priv}_{\text{Conc}}$
Permission, Purpose, Timing, Rationales, $\text{Perm}_{\text{Predict}}$, $\text{Perm}_{\text{Sens}}$, Clarity, Decision, $\text{Des}_{\text{Inform}}$, $\text{Des}_{\text{Satis}}$, $\text{Des}_{\text{Control}}$
App Category, Familiarity

Figure 4: Hierarchical structure of the user study.

interaction of timing and presence/absence of rationales affect (1) users' runtime permission decision, (2) the evaluation of their decision, and (3) their perceived clarity of the permission purpose? Since timing and rationales differentiate the runtime permission request model from its predecessor, it is essential to understand how these factors affect users from different perspectives, even after considering other key factors found in prior work. By answering this question, we expect to gain insights on how developers should request permissions to maximize the benefits of the runtime permission model. Based on these findings, we will also discuss Google's guidelines [2] and potential system support.

For a holistic understanding of user's perspective, we included both the permission decision (grant/deny) and the subjective evaluation of this decision as outcome variables, where the latter reflects whether the decision was made according to users' individual privacy preferences in a given context. For this, we used the Decision Evaluation Scales (DES) [37] which we adopted from the field of health psychology. These scales were originally designed to evaluate patients' decision to uptake/refuse a treatment choice. Comparing users' permission decision with patients' treatment choice, both have two options: grant/deny a permission or uptake/refuse a treatment. Additionally, both have a direct impact on users' security or patients' health. Based on these similarities, this measure fits the context of our study, especially considering that the DES account for the multidimensional nature of decisions and capture (1) whether users received sufficient information to make an informed decision, (2) their satisfaction with the decision, and (3) their perceived control over the decision. We also measure users' understanding of why the app needs the requested permission, which provides information about how certain combinations of timing (upfront/in-context) and rationales (with/without) better communicate permission purposes.

5.1 Study Design

We designed the study as an online experiment with repeated measures. Experimental research has the unique strength of high internal validity because it is able to isolate causal relationships through systematic manipulation of the variables of interest (timing and presence/absence of rationales) while controlling for the spurious effect of other extraneous variables (user and app-related differences) [38, 39]. We used a within-subject design (repeated measures) because it reduces errors associated with individual differences and because the alternative (between-subjects) was shown to produce misleading results for studies involving judgment [40]. Since every study design has its limitations, we address these in Section 8. To make responses of users easier to compare, participants were asked about permission requests with the same general purpose. These purposes were identified from previous work [5–7] and encompass that the permission is required for the main functionality of the app (Purpose$_{\text{Main}}$), a visible feature functionality (Purpose$_{\text{Visible}}$), or a hidden feature functionality (Purpose$_{\text{Hidden}}$). Despite certain advantages (e.g., high external validity), we chose not to conduct this study as a field study because surveying users' permission decisions in the wild presents certain drawbacks. For example, if we were to use an app with accessibility features, we would have to constantly log app changes, which is an invasion of privacy and would lead to opt-in bias. We also would need to first revoke all permissions in order to monitor participants' decisions and to deny all requests once for most rationales to be shown, requiring participants to follow a complex workflow.

Our study had a hierarchical structure in which users interacted with permission requests from different apps. To account for the fact that observations for the same user and app would be similar to each other, we designed this study using a multilevel model [41]. Multilevel models are used for the statistical analysis of hierarchical data, where groups in the study are treated as a random sample from a population of groups. This allows us to make inferences about the population of apps and users, beyond the ones present in the study [41]. Figure 4 depicts the levels of the user study. Each user interacted with four permission requests on the Level$_{\text{Request}}$, one per possible combination of timing (upfront/in-context) and rationales (presence/absence). These permission requests belonged to four different apps and the order of the requests was randomized. Level$_{\text{Request}}$ records the outcome variables,

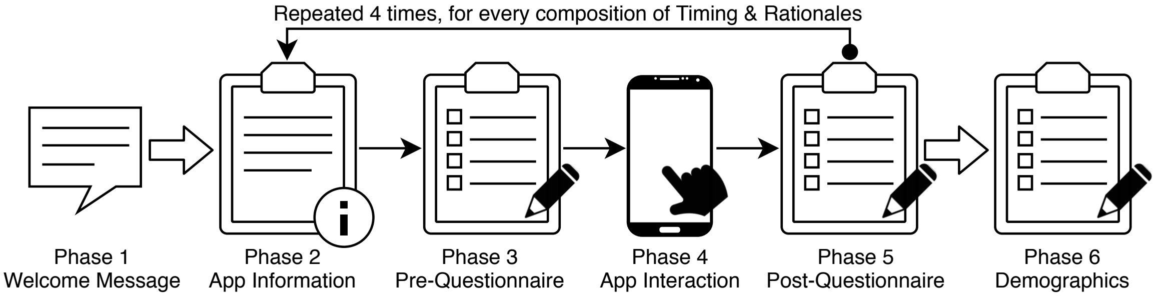


Figure 5: Overview of study procedure. Timing = upfront/in-context, Rationales = with/without.

which are influenced by the variables of interest, in addition to the type, purpose, predictability, clarity, and sensitivity of permission requests. $\text{Level}_{\text{App}}$ represents the diverse characteristics of apps, including app category and participants' familiarity with the app. Whereas $\text{Level}_{\text{User}}$ represents the diverse characteristics of users (i.e., participants' gender, age, education, computer science background, mobile OS, privacy concerns, and prior privacy experiences).

5.2 Procedure

As shown in Figure 5, participants first read about the study and gave their consent (phase 1). This was followed by the main part of the study during which participants went through phases 2–5 four times, once per possible composition of timing (upfront/in-context) and rationales (presence/absence), each time for a different app. These phases were designed to come closest to users' interaction with real-life apps. For that, we gave participants a goal to achieve through the app. We also provided participants with the app's description, name, and icon so they had an idea what the app was about. In addition, we used interactive mockup apps, allowing participants to click through the app interactively, just like on their real phones. The user study procedure with a sample mockup app is shown in Appendix A. Phases 2–5 are described next.

App information (phase 2): Participants were introduced to the app by receiving a brief description of its functionalities, and a goal they needed to achieve through the app (e.g., you want to use this app to have a conference call with your work colleagues, or you want to use this app to backup your vehicle's data). Each goal was based on one of the app's functionalities that would also require a permission. We also provided participants with the app name and icon.

Pre-questionnaire (phase 3): This phase covers users' first impression of the app. We asked participants whether they were familiar with the app, and if they would expect it to request access to a permission protected resource specific to each app. We also measured the perceived sensitivity ($\text{Perm}_{\text{Sens}}$), and clarity of the permission purpose ($\text{Clarity}_{\text{Pre}}$).

App interaction (phase 4): We reminded participants of the goal they want to achieve through the app and then asked them to interact with an interactive mockup app like they would on their own phones. Each app interaction ended with a permission request dialog. The order in which participants interacted with the different combinations of timing (upfront/in-context) and rationales (with/without) was randomized.

Post-questionnaire (phase 5): After participants interacted with the app, we asked them if they would grant the requested permission (Decision). We again recorded participants' clarity on the permission purpose ($\text{Clarity}_{\text{Post}}$). Other questions inquired about the purpose category of the permission request ($\text{Perm}_{\text{Purp}}$), and some questions were only present when rationales were provided. They investigated the origin of the rationale message ($\text{Rationale}_{\text{Origin}}$), and their collection of the content of that message ($\text{Rationale}_{\text{Recall}}$). Then, on a separate screen, we reminded participants about their previous decision and asked them to evaluate their choice using the Decisions Evaluation Scales (DES), consisting of informed decision ($\text{Des}_{\text{Inform}}$), decision satisfaction ($\text{Des}_{\text{Satis}}$), and decision control ($\text{Des}_{\text{Control}}$).

After answering questions for the four apps, participants were asked to provide some demographic information (phase 6). The study procedure and all measurements were tested and adjusted after running a pilot study with 25 participants.

5.3 Recruitment and Incentives

Data were collected on MTurk using TurkPrime [42], an online platform that facilitates setting up and executing studies on MTurk. We paid participants \$12.00/hour, meaning that participants received \$3.00 for completing this 15 min study.

To ensure high quality of data collected through MTurk, we followed a number of suggestions in the literature [43,44]. MTurk workers could only participate in the study if they had a US account and had an approval rate of at least 95%. In order to also collect responses from naive workers (i.e., workers who were not repeatedly exposed to similar studies), we set the required number of completed HITs between 0 and 100 for about 10% of all HITs. Additionally, we added the completion code at the beginning of the study (phase 1) to increase participants' trust (only 1.15% tried to submit the completion code without doing the survey). Finally, we provided one attention check item in the middle of the study and monitored whether participants interacted with the mockup apps. We excluded participants who failed the attention check and did not interact with at least two of the mockup apps.

Since power analysis for multilevel models is still considered a complex problem [41], we estimated the required sample size without considering the multilevel structure of our data. Using G*Power [45], we estimate that we need at least 400 participants. A total of 698 MTurk workers attempted to participate in our study, from which we removed 225 respondents based on the screening criteria described above. Our final sample included 473 participants, 36.8% ($N = 174$) of whom identified themselves as female. The mean age was 37.08 years ($SD = 10.59$). The majority of participants attended college, 17.5% did not finish their studies, 51.4% had a bachelor's degree, and 18.4% had a graduate degree. 69.8% owned an Android smartphone, and 28.3% an iPhone. About one third of all participants had a background in computer science. Appendix B shows the demographics of the sample.

5.4 Measurements

We used different measurements in our study, which are described next and are listed in the questionnaire in Appendix A.

5.4.1 Decision Evaluation Scales (DES)

We used the Decision Evaluation Scales (DES) [37] to assess users' permission decisions. It consist of three subscales: informed decision, decision satisfaction, and decision control. These scales were originally developed to assess how patients evaluate their medical treatment choice. Since such choices often involve multiple parties (e.g., doctors and family members) and permission decisions tend to be made individually, we had to adjust each subscale. To do so, we used an expert rating procedure to select suitable items per subscale. The experts came from both the field of computer science ($N = 3$) and psychology ($N = 4$). The final instruction was the following: "In a previous question you chose to {grant/deny} this app access to your {permission protected resource}. We would like to know how you feel about this decision."

Informed Decision (Des$_{\text{Inform}}$): This subscale measures whether users feel that they have received sufficient information to make a decision, it consists of four items ($\alpha = 0.76$). Sample items are "I made a well-informed choice" and "I know the pros and cons of granting this app access to my {permission protected resource}." Items are scored on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate better informed decision.

Decision Satisfaction (Des$_{\text{Satis}}$): Measures the general feeling of users in terms of confidence and satisfaction with their decision. Sample items are "I am satisfied with my decision" and "I am doubtful about my choice" (reverse coded), with four items in total ($\alpha = 0.84$). Items were rated on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate higher/greater satisfaction.

Decision Control (Des$_{\text{Control}}$): Measures whether users had the feeling that they were forced to their decision. This scale consists of four items ($\alpha = 0.80$). Sample items are "I feel that the app forced me to make this decision" (reverse coded) and "This was my own decision." Items are scored on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate more perceived control.

5.4.2 Permission Clarity (Clarity)

The extent to which users understand why an app needs a permission was shown to affect users' permission decisions [3, 5, 31]. Therefore, we developed a three item scale to measure the clarity of permission purposes ($\alpha = 0.91$). We were particularly interested in whether interacting with the app increases the initial clarity of a permission request. So we used this scale once before (Clarity$_{\text{Pre}}$) and once after app interaction (Clarity$_{\text{Post}}$). Sample items are "It is clear to me why this app needs access to my {permission protected resource}" and "I have no idea why this app wants access to my {permission protected resource}" (reverse coded). Items are scored on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate greater clarity.

Additionally we recorded participants' permission decisions (Decision): "Based on your interaction with this app, would you grant this app access to your {permission protected resource}?" We also asked participants about what they thought was the purpose of the requested permission (Perm$_{\text{Purp}}$). Answer options included "for the main functionality of the app", "for some additional feature functionality", "do not know", or "for some other reason." Additionally, questions were also asked when rationales were provided. We recorded who, in the participants' opinion, provided the rationale (Rationale$_{\text{Origin}}$): "the mobile operating system", "the app developer" or "some other entity". We also asked participants to recall the content of the rationale (Rationale$_{\text{Recall}}$).

5.4.3 Control Variables from Previous Work

Previous research identified several situational, app and user-specific variables that may also influence users' permission decisions. Therefore, we included the following variables in our study to control for their effects: (1) Permission purpose (Purpose$_{\text{Main}}$, Purpose$_{\text{Visible}}$, Purpose$_{\text{Hidden}}$), the purpose associated with a permission request is one of the major predictors for permission decisions [3–7]. That is why we classified permission requests in one of three permission purpose categories. (2) Permission sensitivity (Perm$_{\text{Sens}}$), previous research found that permissions that the user considers sensitive were more likely to be denied [3, 4, 8]. (3) Privacy concerns

($\text{Priv}_{\text{Conc}}$) and (4) prior privacy experience ($\text{Prior}_{\text{Exp}}$) are related to users' attitude towards their private data, thus, both may affect users' permission decisions [3, 9]. Next, we explain how these variables were measured.

Permission Sensitivity ($\textbf{Perm}_{\textbf{Sens}}$): Three items were used to measure the perceived sensitivity of requested permissions, adapted from prior literature [46] to fit in the context of permission requests ($\alpha = 0.80$). The instructions for participants were the following "When using mobile apps, many people find that there are some resource accesses (permissions) that they are generally comfortable granting, some accesses that they are only comfortable granting under certain conditions, and some accesses are too sensitive that they never or only rarely are comfortable granting. Given the information that this app will request access to your {permission protected resource}. Please indicate to what extent you agree or disagree with the following statements." Sample items are "In general, I do not feel comfortable granting access to my {permission protected resource}" and "The access to my {permission protected resource} is very sensitive to me." Items are scored on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate higher/greater sensitivity.

Privacy Concerns ($\textbf{Priv}_{\textbf{Conc}}$): We measured privacy concerns using a 3-item scale from previous work [47], which was originally developed by Smith et al. [48]. We slightly adapted this scale to measure privacy concerns in apps ($\alpha = 0.85$). Sample items are "Compared to others, I am more sensitive about the way mobile apps handle my personal information" and "To me, it is the most important thing to keep my privacy intact from mobile apps." Items are scored on a 7-point scale (1 = strongly disagree; 7 = strongly agree), where higher scores indicate higher/more privacy concerns.

Prior Privacy Experience ($\textbf{Prior}_{\textbf{Exp}}$): We measured prior privacy experience using a 3-item scale from previous work [48], which was adapted to measure prior privacy experience with apps ($\alpha = 0.80$). Sample items are "How often have you personally experienced incidents whereby your personal information was used by some mobile app without your authorization?" and "How much have you heard or read during the last year about the use and potential misuse of the information collected from mobile apps?" Items are scored on a 7-point scale (1 = never; 7 = very great extent), where higher scores indicate more exposure to privacy experiences.

Other Control Variables: Because users might behave differently when they expect and know something, we controlled for predictability of permission requests and users' familiarity with the app in addition to user demographics.

5.5 App Selection

The user study covered a wide range of apps that requested different permissions for various purposes to rule out possible alternative explanations for our results depending on app-related differences. To achieve that, we selected a set of apps from different categories, each requesting a permission for one of the three permission purposes ($\text{Purpose}_{\text{Main}}$, $\text{Purpose}_{\text{Visible}}$, $\text{Purpose}_{\text{Hidden}}$). However, we could not rely on the standard Play categories, as apps are organized into superordinate topics, where one topic can contain apps with completely different functionalities (e.g., productivity category contains both barcode scanner and calendar apps). Therefore, we clustered the apps from the empirical analysis based on their description into 25 clusters using the Latent Dirichlet Allocation (LDA) topic modelling technique and randomly selected 10 clusters for the user study. We then manually choose three apps per cluster, each requesting a permission for one of the three permission purposes. Based on our empirical analysis, we limited the study to the six most commonly found permissions (`MICROPHONE`, `CONTACTS`, `PHONE`, `CAMERA`, `LOCATION`, and `STORAGE`). We excluded any app that required login (e.g., banking and dating apps), since we did not analyze those in our empirical analysis. A list of the apps used in this study and their categories are shown in Appendix D.

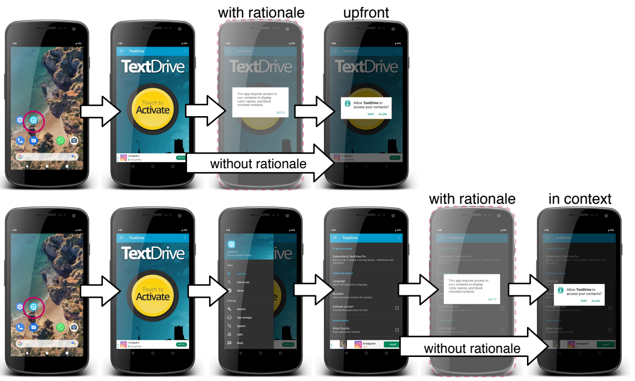


Figure 6: Four different versions of the same app depending on timing (upfront/in-context) and rationales (with/without).

In total, we used 30 apps, of which we captured screenshots of the states that led to the permission request. We used these screenshots to create interactive mockup apps that worked similar to real-world apps. Each app was then modified to request a permission for each of the four possible combinations of timing (upfront/in-context) and rationales (with/without), resulting in a total of 120 app variations. Figure 6 shows such an app with the four different versions.

5.6 Rationale Selection

To investigate the effect of presence/absence of rationales as they are intended to be [49, 50], we decided to only use rationales with additional information. We also focused on one standardized rationale design to ensure comparability of the results, which was informed by the empirical analysis.

Our study apps showed participants one permission request preceded by a rationale (depending on the experiment version). For that, we chose the dialog design because it has the highest priority in conveying information to the user [51], and because the alternatives are often used for different purposes (e.g., fullscreen views explain multiple permissions and snackbars are displayed after a permission request).

As for the wording of rationales, guidelines of Google and Apple recommend that rationales should use sentence case, be short, clear, accurate, and polite so people do not feel pressured [49, 50, 52]. From the empirical analysis, we extracted rationales that followed these guidelines (e.g., "Access to camera is required to make new photos", "This app needs your permission to store images to your device," and "This application requires the manage phone call permission to be approved in order to use the favorite store functionality") and derived a general sentence structure to use in our study: *This app requires access to your {permission} to {list of purposes}*. A sample rationale used in this study is found in Figure 1b

The extracted sentence structure then had to be filled with meaningful permission purposes for each user study app. For that, we manually ran each app, checked the app's source code, description, and rationale (if available). Then, we manually selected reasonable purposes from a list of most common permission purposes that we extracted from our empirical analysis and related work [11] using Part-of-Speech tagging (POS tagging) [53]. Examples of purposes include: find bus stops nearby, block harassing calls, and use speech translation.

5.7 Ethical Considerations

The study design and protocol were reviewed and approved by the Ethics Review Board of our institution. We followed the guidelines for academic requesters outlined by MTurk workers [54]. All server-side software (i.e., Limesurvey Community Edition software) was self-hosted on a maintained and hardened server to which only the researchers involved in this study have access. At the beginning of the study, there was an informed consent procedure, which provided participants with details about the purpose of the study and the type of data being collected. We also informed participants about the option to withdraw from the study at any time.

6 Results

We used multilevel regression analysis to evaluate the effects of timing and presence/absence of rationales on users' permission decisions (Decision), the evaluation of their decisions (DES: $\text{Des}_{\text{Inform}}$, $\text{Des}_{\text{Satis}}$, $\text{Des}_{\text{Control}}$), and the perceived clarity of the permission purpose ($\text{Clarity}_{\text{Post}}$). All analyses were performed with R 4.0.2 [55] and the package LME4 [56]. As a data preparation step, we calculated mean scores for measurements with multiple items. We also centered all $\text{Level}_{\text{User}}$ and $\text{Level}_{\text{Request}}$ variables by their total mean (grand mean centering) to facilitate interpretation of regression models.

A correlation analysis showed that participants' education, their computer science background, their familiarity with the app, the predictability of the requested permission, and the requested permission type were highly correlated. We also observed a high positive correlation between the perceived sensitivity of permissions and participants' privacy concerns, meaning that participants who care about their privacy usually tend to find permissions more sensitive [3]. Additionally, we found a significant negative correlation between participants' permission clarity prior app interaction and the purpose of the permission, which is conclusive since the purpose of permissions requested for the main functionality or a visible feature may be more clear to users than for a hidden feature.

6.1 Model Construction

We used a linear multilevel model for $\text{Des}_{\text{Inform}}$, $\text{Des}_{\text{Satis}}$, $\text{Des}_{\text{Control}}$, and $\text{Clarity}_{\text{Post}}$, whereas Decision (binary) was modeled using a generalized linear multilevel model. The comparison between a simple and a multilevel regression model showed that multilevel models explain our data significantly better (see Appendix C). To prevent overparameterization of the models, we built and tested them in a step-by-step approach, following recommendations in the literature [41] in each step. All models were calculated using maximum likelihood estimation to ensure their comparability. Next, we explain the model building process, which was held constant for all outcome variables.

In a first step, after a simple regression model, we created a random intercept model by adding app and user as random effects. In a second step, we included all variables that were identified from previous work: $\text{Clarity}_{\text{Pre}}$ [3, 5, 31], $\text{Priv}_{\text{Conc}}$ [3], $\text{Prior}_{\text{Exp}}$ [9], Purpose [3–7], and $\text{Perm}_{\text{Sens}}$ [3, 4, 8]. We also added participants' decision (Decision) as a control variable to the DES, because the decision outcome (i.e., granting or denying a permission request) has an influence on users' comfort level with their decisions [3]. In a third step, we added the variables of interest, Timing (upfront, in-context) and Rationales (with, without). Finally, in a fourth step, we added the interaction between Timing and Rationales when this improved the model fit. For more details about the model building process, see Appendix C.

6.2 Final Models

The final models were recalculated using Restricted Maximum Likelihood Estimation, which leads to a more conservative and less error-prone estimation of the parameters [41]. Table 1 shows the final model for each outcome variable.

We followed suggestions of literature [57] to identify and handle outliers. We checked for multi-construct outliers on the aggregated $\text{Level}_{\text{App}}$ and found no conspicuous data points.

Table 1: The final multilevel models.

	Decision *Odds Ratio* (*std.* β)	Des$_{\text{Inform}}$ β (*std.* β)	Des$_{\text{Satis}}$ β (*std.* β)	Des$_{\text{Control}}$ β (*std.* β)	Clarity$_{\text{Post}}$ β (*std.* β)
Level$_{\text{User}}$					
(Intercept)	2.92 (1.06)**	3.90 (-0.54)***	6.17 (0.24)***	5.34 (0.07)***	4.53 (-0.21)***
Priv$_{\text{Conc}}$	0.64 (-0.57)***	-0.02 (-0.02)	0.06 (0.07)	0.00 (0.00)	-0.01 (-0.01)
Prior$_{\text{Exp}}$	1.91 (1.00)***	-0.03 (-0.03)	-0.25 (-0.35)***	-0.29 (-0.32)***	-0.09 (-0.07)***
Level$_{\text{Request}}$					
Purpose					
VisibleFeature	1.35 (0.3)	0.24 (0.18)*	0.03 (0.03)	0.18 (0.13)	0.14 (0.07)
HiddenFeature	0.35 (-1.05)*	-0.05 (-0.04)	0.07 (0.06)	-0.05 (-0.04)	-0.48 (-0.23)**
Clarity$_{\text{Pre}}$	2.06 (1.53)***	0.18 (0.28)***	0.09 (0.18)***	0.07 (0.12)***	0.59 (0.61)***
Perm$_{\text{Sens}}$	0.53 (-0.99)***	-0.01 (-0.01)	0.00 (0.01)	-0.01 (-0.01)	-0.03 (-0.02)
Decision(Allow)	–	0.44 (0.32)***	-0.57 (-0.53) ***	-0.30 (-0.22)***	–
Timing(InContext)	1.48 (0.39)*	0.30 (0.22)***	0.08 (0.08)	0.06 (0.04)	0.36 (0.18)***
Rationales(WithRationale)	2.73 (1.00)***	0.66 (0.48)***	0.18 (0.17)***	0.07 (0.05)	0.93 (0.46)***
Interaction(Timing:Rationales)	–	-0.37 (-0.27)***	-0.20 (-0.19)**	–	-0.37 (-0.18)**
Marginal R^2	0.483	0.211	0.198	0.136	0.504
Conditional R^2	0.765	0.476	0.549	0.679	0.562

Three-level regression model for each outcome variable. The coefficients for Decision are shown as odds ratios, where values <1 indicate a lower likelihood to grant permissions and values >1 indicate a higher likelihood. $std.\,\beta = standardized\,\beta$. $^{*}p < .05, ^{**}p < .001, ^{***}p < .0001$. Decision coding: $0 = deny$, $1 = allow$. N$_{\text{User}}$ = 473, N$_{\text{App}}$ = 30, N$_{\text{Request}}$ = 1824. Note that Level$_{\text{App}}$ is not shown because the final models do not contain variables from that level.

Then, we checked for multi-construct outliers on the Level$_{\text{User}}$ and found 3 participants with conspicuous Mahalanobis distances. We also found 6 outliers on the Level$_{\text{Request}}$. Since the removal of outliers did not change the model fits, significance levels, and conclusions, we opted to keep them in the analysis [57]. Additionally, we checked the final models for multicollinearity and found no such case ($VIF < 2$).

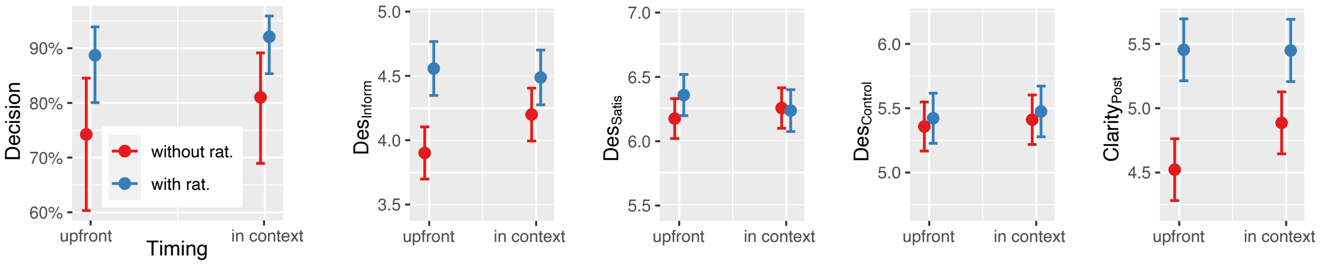
Effect of users' individuality: The final models were able to explain 47.6%–76.5% of the total variance in the outcome variables (Conditional R^2), whereby it is worth to note that **a large proportion of this variance is explained by the individual differences between users**. For example, in the final Decision model, intraclass correlation for the Level$_{\text{User}}$ was $ICC = 0.490$, which means that 49% of the empirical variance of permission decisions can solely be explained by individual differences between users. The same applies for the DES: Des$_{\text{Inform}}$ ($ICC_{\text{User}} = 0.321$), Des$_{\text{Satis}}$ ($ICC_{\text{User}} = 0.432$), and Des$_{\text{Control}}$ ($ICC_{\text{User}} = 0.625$). In contrast, differences between users in the Clarity$_{\text{Post}}$ model only explained 7.8% of the empirical variance, which is due to the fact that we controlled for Clarity$_{\text{Pre}}$ in the same model.

6.3 Effect of Timing and Rationales

Permission Decision (Decision): Participants' permission decision was explained best (76.5% of the empirical variance) by a model including the two main variables of interest but not their interaction ($Model\ Step\ 3$, $AIC = 1449.35$, $LogLik = -713.68$). We found that both **timing and rationales had a positive effect on grant rates**. When permissions were requested in context, grant rates increased by 48% ($odds\ ratio = 1.48$, $standardized\ \beta = 0.39$, $p = 0.017$). Additionally, it was 173% more likely that participants grant permissions when rationales were provided compared to permission requests without rationales ($odds\ ratio = 2.73$, $std.\,\beta = 1.00$, $p < 0.001$). Overall, if permissions were requested upfront and without rationales, they were granted in only 74% of the cases, while they were granted in 92% of the cases if they were requested in context and with rationales (see Figure 7a for an overview of the predicted probabilities of granting permissions).

Informed Decision (Des$_{\text{Inform}}$): Participants' perception of having made an informed decision was explained best (47.6% of the empirical variance) by a model including the variables of interest and their interaction ($Model\ Step\ 4$, $AIC = 5633.44$, $LogLik = -2802.72$). The model shows a significant interaction of timing and rationales ($\beta = -0.37$, $std.\,\beta = -0.27$, $p < 0.001$). Overall, **rationales had a positive effect on whether participants' decision was informed, however, this effect was stronger when rationales were shown upfront** instead of in context. Furthermore, timing was only significant when no rationales were present. This means that **without rationales, requesting permissions in context increases informed decision**, as is depicted in Figure 7b.

Decision Satisfaction (Des$_{\text{Satis}}$): Participants' satisfaction with their decision was explained best (54.9% of the empirical variance) by a model including the two main variables of interest as well as their interaction ($Model\ Step\ 4$, $AIC = 4695.43$, $LogLik = -2333.72$). The results show a significant interaction of timing and rationales ($\beta = -0.20$, $std.\,\beta = -0.19$, $p = 0.003$). On one hand, **when permissions were**



(a) Decision

(b) $\text{Des}_{\text{Inform}}$

(c) $\text{Des}_{\text{Satis}}$

(d) $\text{Des}_{\text{Control}}$

(e) $\text{Clarity}_{\text{Post}}$

Figure 7: Effects of timing and rationales on each outcome variable. Means were predicted holding all other variables constant on the reference/average level. Error bars represent 95% confidence intervals of the predicted means.[1]

requested upfront, rationales had a positive effect on decision satisfaction, but when requested in context, rationales had no significant effect. On the other hand, **timing had no effect on satisfaction** (see Figure 7c).

Decision Control ($\textbf{Des}_{\textbf{Control}}$): Participants' perceived control over their permission decision was explained best (67.9% of the empirical variance) by a model that included the two variables of interest but without their interaction ($Model\,Step\,3$, $AIC = 5243.57$, $LogLik = -2608.78$). The results show **no significant effect of timing and rationales on decision control**, as shown in Figure 7d.

Permission Clarity ($\textbf{Clarity}_{\textbf{Post}}$): Participants' perceived clarity of the permission purpose was explained best (56.2% of the empirical variance) by a model including the two main variables of interest as well as their interaction ($Model\,Step\,4$, $AIC = 6418.44$, $LogLik = -3196.22$). After controlling for the initial clarity of permission requests, we found a significant interaction of timing and rationales ($\beta = -0.37$, $std.\beta = -0.18$, $p = 0.003$). On one hand, the effect of timing was only significant without rationales, meaning that **post clarity increased when permissions were requested in context without rationales.** On the other hand, **rationales significantly increased permission clarity** for both upfront and in-context permission requests, however, this effect is stronger for upfront requests, as shown in Figure 7e.

6.4 Effect of Other Variables

Privacy Concerns ($\textbf{Priv}_{\textbf{Conc}}$): Participants' privacy concerns had a negative effect on the likelihood to grant permissions ($odds\,ratio = 0.64$, $std.\beta = -0.57$, $p < 0.001$), but not on the other outcome variables. In other words, participants with higher privacy concerns are less likely to grant permissions than those with lower concerns.

Prior Privacy Experience ($\textbf{Prior}_{\textbf{Exp}}$): The data revealed that the more participants dealt with privacy related experiences in the past, the more likely they were to grant permissions ($odds\,ratio = 1.91$, $std.\beta = 1.00$, $p < 0.001$). Whereas for decision satisfaction, decision control, and clarity of the requested permission more privacy related experiences decreased the score of these scales. Only for informed decision, we could not find an effect of prior privacy experience.

Permission Clarity ($\textbf{Clarity}_{\textbf{Pre}}$): Participants' initial clarity of the permission purpose had a significant effect on all outcome variables. Having an initial understanding of the permission purpose increased the odds to grant permissions by 106% ($odds\,ratio = 2.06$, $std.\beta = 1.53$, $p < 0.001$). Also, for all three DES, a positive effect of initial clarity was found. Furthermore, the clearer the permission request was before interacting with the app, the clearer it was afterwards ($\beta = 0.59$, $std.\beta = 0.61$, $p < 0.001$).

Permission Sensitivity ($\textbf{Perm}_{\textbf{Sens}}$): There was a negative effect of permission sensitivity on decision ($odds\,ratio = 0.53$, $std.\beta = -0.99$, $p < 0.001$). Meaning, that permissions perceived as sensitive are less likely to be granted.

Permission Decision as a Control Variable (Decision): As for the effect of permission decision, we found that granting a permission increased the perception that the decision was informed, while it decreased decision satisfaction and the perception of being in control.

Effect of other control variables: To rule out potential alternative explanations for our results, we built additional models to examine whether there were any changes in the outcomes due to the ordering of permission requests, having interacted with the app before, and the predictability of permissions. None of these control variables significantly changed the effect of timing and rationales on the outcome variables. Also, we did not find a significant effect of gender or age. Neither did participants' education, having a computer science background, or participants' mobile OS explain any additional

[1] Due to our within-subject design and the resulting paired data, the confidence intervals from Figure 7 cannot be interpreted as an indicator of the statistical significance of the main/interaction effects [58].

variance of our data. Additionally, we built the DES models with and without Decision as a control variable and found no significant difference in the effect of timing and rationales.

6.5 Rationale Recall (Rationale$_{\text{Recall}}$)

To further rule out potential alternative explanations for our results, we built the models again for attentive participants only. For that, two researchers analyzed and independently coded the free text answers of participants' ability to recall the content of the rationale messages. The analysis showed almost perfect inter-rater agreement between the two coders ($Cohen's\,\kappa = 0.87$) and all differences were resolved in agreement. Four themes emerged in the coding process: (1) Participants correctly recalled all or parts of the rationale message (correct), (2) they did not recall the content of the rationale and provided unrelated responses (unrelated), (3) they admit to have forgotten the content of the rationale (forgotten), or (4) they claim to have not seen the rationale dialog (unseen). From all rationale recall answers ($N = 899$), 49% were coded as correct, 45% as unrelated, 5% as forgotten, and 1% as unseen. These percentages reflect the user's general inattention to security and privacy related information [59–61] that would have also occurred if participants interacted with the apps on their real phones. Each model was built again for attentive participants who recalled the content of at least one of the rationales. We found that the effect on timing and rationales was consistent and did not change. The only difference was that rationales had a significant effect on Des$_{\text{Control}}$. In order to stay on the conservative side, we only considered the results of the main analysis.

6.6 Rationale Origin (Rationale$_{\text{Origin}}$)

Participants were asked once about the rationale origin for each app that displayed a rationale. However, since each participant interacted with two apps with rationales, we only considered the last response given. We found that 57% (270) of the participants correctly identified the app developer as the provider of the rationale, while 37% (175) thought that it came from the operating system. We checked whether the operating system of the participant's mobile phone was one of the reasons for this misunderstanding, which was not the case. The remaining 26 participants said that they do not know who provided rationales and 2 gave unrelated answers.

6.7 Permission Purpose (Perm$_{\text{Purp}}$)

We found that asking participants about the purpose of permissions did not yield useful insights, as the responses reflected participants' subjective perception of permission purposes. Therefore, we do not report on the results.

7 Discussion

Our study is the first to explore the effect of timing and rationales and their interplay on users' runtime permission decisions and the evaluation of their decisions. We found that timing and rationales matter even after accounting for user and app-level differences identified in previous work. In addition, we showed that timing and rationales should not be evaluated in isolation because both might influence one another. We also found that a large proportion of the variance in the outcome variables can be explained by the individual differences between users.

Effect of timing. Requesting permissions in context primarily benefits developers, as such an approach increases grant rates. Whereas requesting permissions in-context only has a positive effect on users' perception of their decisions without rationales.

Effect of rationales. Requesting permissions with rationales benefits both developers and users, as such an approach increases grant rates, helps users in making informed decisions by increasing their understanding of the permission purpose, and positively affects decision satisfaction. Whereby, the benefits of rationales are greatest for upfront requests, when users may lack contextual data for decision making.

Alternative to Google's guidelines. Google's guidelines recommend to use four strategies to help developers keep deny rates to a minimum [2]. The guidelines suggest requesting app-critical permissions upfront and function-specific permissions in context, in addition to providing rationales for unclear permissions. While these suggestions seem straightforward, we found while designing our study and also in previous work [8], that permission clarity is a subjective measure. Thus, it is unreasonable to require developers to accurately evaluate which permission requests might not be clear to their end users (and therefore require a rationale). In addition, our results show that some permission request strategies are, on average, less effective than others. For example, when asked for permission upfront without rationale, users are least likely to grant permissions and positively perceive their permission decisions. Therefore, it is less effective than the other three strategies. We also found that adding rationales (upfront as well as in-context) benefits both developers and end users. Developers primarily profit from increased grant rates, while users are able to make informed decisions that they better understand and are more satisfied with.

Based on these findings, we propose to adjust Google's guidelines as follows. Instead of four permission request strategies, we limit developers' choices to two strategies only. Permissions should be either requested upfront with rationale or in context with rationale. Therefore, unlike Google's guidelines, we recommend that rationales should *always* be present, while preserving their suggestion to request app-critical permissions upfront and function-specific permissions in context.

With this simplification, we expect to keep grant rates at a high level and at the same time make users feel comfortable with their runtime permission decisions.

Individually tailored system support. Google's guidelines put the burden on developers to decide when to request permissions (timing) and whether to provide further explanations (rationales). Even with our improvements, developers' still have to time permission requests for all users. Additionally, our results showed that users differ in their decisions and the way they make those decisions, led by their own values and preferences. So, instead of a strategy that attempts to fit all users with the burden on developers, our intuitive deduction is to provide a solution to support users' individuality.

One concrete suggestion is the customization of permission requests on a per-user basis, realized by the operating system. Users could use system settings to determine when they want to be asked for permissions and whether they prefer to see rationales. While developers only have to follow a simple pattern to label the in-context positions for permissions and provide a list of rationales (similar to iOS [62]). One advantage of this consistent approach is that users will not be surprised/annoyed by permission requests because they know when to expect them. Since requesting permissions on Android and iOS is similar, this solution is also realizable in iOS. However, it should be noted that design changes to the mobile OS must protect against malicious developers who could provide misleading or erroneous in-context timings and rationales, which is orthogonal to our work. While the actual design and evaluation of such systems is part of future work.

Rationale origin misconception. While the majority of participants identified the developer as the author of rationale messages, a large number still thought that the rationales were created by the operating system (37%). This could be a side-effect of using standardized rationales for the apps in our user study. However, rationale messages in iOS are already integrated in the standard permission dialog [62]. Therefore, we recommend adding an indicator that the rationale is provided by the app developer. This could be a short message proceeding the rationale. For example: "{App name} says: {Rationale message of the app developer}." However, this solution is only applicable when the rationale is standardized by the operating system, as in iOS. Whereas in Android, currently only the app developer is able to highlight the origin of the rationale (e.g., through custom themes and wording).

Generalizability of our findings. When interacting with modern technology, users are often confronted with security and privacy-relevant decisions. Such decisions must be informed while being consistent with users' individual values and preferences. To offer users more transparency, previous research focused on providing comprehensive privacy policies (e.g., in the form of "privacy nutrition facts" [63]) and effective browser security warning messages [60, 64, 65].

Consistent with these findings and in the context of permission requests, we found that users made better-informed decisions and were more satisfied with their decisions when they were provided transparency, in the form of rationales and to a lesser extent by requesting permissions in the appropriate context (timing). Thereby, our results are also consistent with previous work on other channels in the mobile domain (e.g., provide security-related behavior in app descriptions [19], explain permission usage based on code [16], and aid users in the app-selection process [66–68]), all of which emphasize the crucial importance of transparency for users' decision making process. Not least, these research results might also be a reason for the recently increasing efforts of the two major mobile operating systems towards transparency of privacy and security of apps, e.g., by introducing "privacy labels" in iOS or an upcoming safety section in Google Play [69].

In line with these efforts to aid users in their decisions, we recommend that rationales should always be provided by developers. However, future research is needed to optimize how frequently they are displayed to the user, e.g., leveraging machine learning to learn individual preferences [29, 30]. For example, depending on users' individual preferences, a user who always denies a certain permission or always denies permission for certain app types may not need additional rationales in these situations. We believe that our findings on rationales are also applicable to other security and privacy critical decisions. While how rationales should look like is system dependent, they all need to strike a balance between adequately informing and overwhelming users. Since our results show that just the presence of rationales is beneficial, future work could study the magnitude of this effect depending on different rationale designs and contents.

8 Threats to Validity

As with any empirical study, there are aspects of our study design that might limit the generalizability of results. First, our data was collected in a highly standardized, somewhat artificial situation. Therefore, it might be fair to question whether our results fully reflect the behavior of users on real apps. However, only such experimental research methods that provide conscious control of all aspects of a situation (high internal validity), allow the direct inference of causal relationships [39]. To address potential negative effects of this design decision, we followed best practice recommendations for this kind of experimental studies [70]. For example, our participants were given a consistent storyline and clear goals they should reach with their apps as well as interactive mockup apps. These measures ensure a high level of immersion for participants, which, as prior work has shown, leads to the highest possible generalizability of the study results [70–72].

Second, our research topic – permission requests – was obvious to our participants at several points in our study, which may have primed their behavior in a certain way. For example, we asked participants about a permission prior app interaction (making them aware that the app will request this permission). This was necessary, as some variables (i.e., permission sensitivity/predictability/clarity) could only be accurately measured before users interacted with the app. However, from the users' perspective this is very similar to checking permissions in the app store before installing the app. Another priming could have resulted from the fact that each participant went through the main part of the study for several apps. We mitigated potential carryover- and order-effects arising from this within-study design by randomizing the order of the permission request types (upfront/in-context, with/without rationale) and checking that the order did not affect our results.

9 Conclusion

In this work, we showed that timing of permission requests and presence/absence of rationales have an effect on users' permission decisions and the evaluation of their decisions. We found that the effect of timing and rationales depend on one another and should not be evaluated on their own. Based on the results, we suggest that the current Google guidelines should be refined to better aid users in their decision-making process. Further, we highlight that permission decisions mainly depend on the individuality of users, suggesting that there is no one-fits-all permission request strategy. As a conclusion, current mobile platforms could benefit from a customized solution on a per-user basis, in which users can define when permissions should be requested and whether rationales should be given.

Acknowledgment

We thank Kassem Fawaz and the anonymous reviewers for their insightful comments and suggestions.

References

[1] A. Porter Felt, S. Egelman, M. Finifter, D. Akhawe, and D. A. Wagner, "How to ask for permission," in *Proc. 7th USENIX Workshop on Hot Topics in Security (HotSec'12)*, 2012.

[2] Material Design, "Android permissions," https://material.io/design/platform-guidance/android-permissions.html, accessed: 2021-05-26.

[3] B. Bonné, S. T. Peddinti, I. Bilogrevic, and N. Taft, "Exploring decision making with Android's runtime permission dialogs using in-context surveys," in *Proc. 13th Symposium on Usable Privacy and Security (SOUPS'17)*, 2017.

[4] P. Wijesekera, A. Baokar, A. Hosseini, S. Egelman, D. A. Wagner, and K. Beznosov, "Android permissions remystified: A field study on contextual integrity," in *Proc. 24th USENIX Security Symposium (SEC'15)*, 2015.

[5] B. Liu, M. S. Andersen, F. Schaub, H. Almuhimedi, S. Zhang, N. M. Sadeh, Y. Agarwal, and A. Acquisti, "Follow my recommendations: A personalized privacy assistant for mobile app permissions," in *Proc. 12th Symposium on Usable Privacy and Security (SOUPS'16)*, 2016.

[6] J. Lin, B. Liu, N. M. Sadeh, and J. I. Hong, "Modeling users' mobile app privacy preferences: Restoring usability in a sea of permission settings," in *Proc. 10th Symposium on Usable Privacy and Security (SOUPS'14)*, 2014.

[7] J. Lin, N. M. Sadeh, S. Amini, J. Lindqvist, J. I. Hong, and J. Zhang, "Expectation and purpose: Understanding users' mental models of mobile app privacy through crowdsourcing," in *ACM Conference on Ubiquitous Computing, (Ubicomp'12)*, 2012.

[8] I. Shklovski, S. D. Mainwaring, H. H. Skúladóttir, and H. Borgthorsson, "Leakiness and creepiness in app space: Perceptions of privacy and mobile app use," in *Conference on Human Factors in Computing Systems (CHI'14)*, 2014.

[9] B. Zhang and H. Xu, "Privacy nudges for mobile applications: Effects on the creepiness emotion and privacy attitudes," in *Proc. 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW'16)*, 2016.

[10] J. Tan, K. Nguyen, M. Theodorides, H. Negrón-Arroyo, C. Thompson, S. Egelman, and D. A. Wagner, "The effect of developer-specified explanations for permission requests on smartphone user behavior," in *Conference on Human Factors in Computing Systems (CHI'14)*, 2014.

[11] X. Liu, Y. Leng, W. Yang, W. Wang, C. Zhai, and T. Xie, "A large-scale empirical study on Android runtime-permission rationale messages," in *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2018.

[12] I. Gasparis, A. Aqil, Z. Qian, C. Song, S. V. Krishnamurthy, R. Gupta, and E. Colbert, "Droid M+: Developer support for imbibing Android's new permission model," in *Asia Conference on Computer and Communications Security (AsiaCCS'18)*, 2018.

[13] P. G. Kelley, S. Consolvo, L. F. Cranor, J. Jung, N. M. Sadeh, and D. Wetherall, "A conundrum of permissions: Installing applications on an android smartphone," in *Proc. 16th International Conference on Financial Cryptography and Data Security (FC'12)*, 2012.

[14] A. Porter Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. A. Wagner, "Android permissions: user attention, comprehension, and behavior," in *8th Symposium on Usable Privacy and Security (SOUPS'12)*, 2012.

[15] A. Porter Felt, S. Egelman, and D. A. Wagner, "I've got 99 problems, but vibration ain't one: a survey of smartphone users' concerns," in *Proc. Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM'12)*, 2012.

[16] Z. Qu, V. Rastogi, X. Zhang, Y. Chen, T. Zhu, and Z. Chen, "Autocog: Measuring the description-to-permission fidelity in Android applications," in *Proc. 21st ACM Conference on Computer and Communications Security (SIGSAC'14)*, 2014.

[17] H. Gao, C. Guo, Y. Wu, N. Dong, X. Hou, S. Xu, and J. Xu, "Autoper: Automatic recommender for runtime-permission in Android applications," in *Proc. 43rd IEEE Annual Computer Software and Applications Conference (COMPSAC'19)*, 2019.

[18] X. Pan, Y. Cao, X. Du, B. He, G. Fang, R. Shao, and Y. Chen, "Flowcog: Context-aware semantics extraction and analysis of information flow leaks in Android apps," in *Proc. 27th USENIX Security Symposium, (SEC'18)*, 2018.

[19] M. Zhang, Y. Duan, Q. Feng, and H. Yin, "Towards automatic generation of security-centric descriptions for Android apps," in *Proc. 22nd ACM Conference on Computer and Communications Security (SIGSAC'15)*, 2015.

[20] H. Wang, J. I. Hong, and Y. Guo, "Using text mining to infer the purpose of permission use in mobile apps," in *International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'15)*, 2015.

[21] R. Pandita, X. Xiao, W. Yang, W. Enck, and T. Xie, "WHYPER: Towards automating risk assessment of mobile applications," in *Proc. 22th USENIX Security Symposium (SEC'13)*, 2013.

[22] Y. Feng, L. Chen, A. Zheng, C. Gao, and Z. Zheng, "Ac-net: Assessing the consistency of description and permission in Android apps," *IEEE Access*, vol. 7, pp. 57 829–57 842, 2019.

[23] E. Pan, J. Ren, M. Lindorfer, C. Wilson, and D. R. Choffnes, "Panoptispy: Characterizing audio and video exfiltration from android applications," *Proc. Priv. Enhancing Technol.*, vol. 2018, pp. 33–50, 2018.

[24] R. Stevens, J. Ganz, V. Filkov, P. T. Devanbu, and H. Chen, "Asking for (and about) permissions used by android apps," in *Proc. 10th Working Conference on Mining Software Repositories (MSR'13)*, 2013.

[25] X. Liu, Y. Leng, W. Yang, C. Zhai, and T. Xie, "Mining Android app descriptions for permission requirements recommendation," in *26th IEEE International Requirements Engineering Conference (RE)*, 2018.

[26] D. Bogdanas, "Dperm: Assisting the migration of Android apps to runtime permissions," *CoRR*, 2017. [Online]. Available: http://arxiv.org/abs/1706.05042

[27] S. T. Peddinti, I. Bilogrevic, N. Taft, M. Pelikan, Ú. Erlingsson, P. Anthonysamy, and G. Hogben, "Reducing permission requests in mobile apps," in *Proc. Internet Measurement Conference (IMC'19)*, 2019.

[28] B. Liu, J. Lin, and N. M. Sadeh, "Reconciling mobile app privacy and usability on smartphones: Could user privacy profiles help?" in *Proc. 23rd International World Wide Web Conference (WWW'14)*, 2014.

[29] P. Wijesekera, A. Baokar, L. Tsai, J. Reardon, S. Egelman, D. A. Wagner, and K. Beznosov, "The feasibility of dynamically granted permissions: Aligning mobile privacy with user preferences," in *Proc. 28th IEEE Symposium on Security and Privacy (SP'17)*, 2017.

[30] K. Olejnik, I. Dacosta, J. S. Machado, K. Huguenin, M. E. Khan, and J. Hubaux, "Smarper: Context-Aware and automatic runtime-permissions for mobile devices," in *Proc. 28th IEEE Symposium on Security and Privacy (SP'17)*, 2017.

[31] H. Almuhimedi, F. Schaub, N. M. Sadeh, I. Adjerid, A. Acquisti, J. Gluck, L. F. Cranor, and Y. Agarwal, "Your location has been shared 5, 398 times!: A field study on mobile app privacy nudging," in *33rd Conference on Human Factors in Computing Systems (CHI'15)*, 2015.

[32] L. Tsai, P. Wijesekera, J. Reardon, I. Reyes, S. Egelman, D. A. Wagner, N. Good, and J. Chen, "Turtle guard: Helping Android users apply contextual privacy preferences," in *13th Symposium on Usable Privacy and Security (SOUPS'17)*, 2017.

[33] E. Cunningham, "Improving app security and performance on Google Play," https://android-developers.googleblog.com/2017/12/improving-app-security-and-performance.html, accessed: 2021-05-26.

[34] Y. Li, Z. Yang, Y. Guo, and X. Chen, "Droidbot: A lightweight ui-guided test input generator for Android," in *Proc. 39th International Conference on Software Engineering (ICSE'17)*, 2017.

[35] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *Proc. 21st National Conference on Artificial Intelligence (AAAI'06)*, 2006.

[36] K. K. Micinski, D. Votipka, R. Stevens, N. Kofinas, M. L. Mazurek, and J. S. Foster, "User interactions and permission use on android," in *Conference on Human Factors in Computing Systems (CHI'17)*, 2017.

[37] P. F. Stalmeier, M. S. Roosmalen, L. C. Verhoef, J. E. Hoekstra-Weebers, J. C. Oosterwijk, U. Moog, N. Hoogerbrugge, and W. A. van Daal, "The decision evaluation scales," *Patient Education and Counseling*, vol. 57, pp. 286–293, 2005.

[38] A. Bahattacherjee, *Social science research: Principles, methods and practices (2nd ed.)*. Global text project, 2012.

[39] P. E. Spector, *Research Designs*. SAGE Publications, 1981.

[40] M. Birnbaum, "How to show that 9 > 221: Collect judgments in a between-subjects design," *Psychological Methods*, vol. 4, pp. 243–249, 1999.

[41] J. J. Hox, *Multilevel Analysis: Techniques and Applications (2nd ed.)*. Routledge/Taylor & Francis Group, 2010.

[42] L. Litman, J. Robinson, and T. Abberbock, "Turkprime.com: A versatile crowdsourcing data acquisition platform for the behavioral sciences," *Behavior Research Methods*, vol. 49, pp. 433–442, 2017.

[43] M. Keith, L. Tay, and P. Harms, "Systems perspective of Amazon Mechanical Turk for organizational research: Review and recommendations," *Frontiers in Psychology*, vol. 8, p. 1359, 2017.

[44] J. Robinson, C. Rosenzweig, A. J. Moss, and L. Litman, "Tapped out or barely tapped? Recommendations for how to harness the vast and largely unused potential of the Mechanical Turk participant pool," *PLOS ONE*, vol. 14, pp. 1–29, 2019.

[45] F. Faul, E. Erdfelder, A. Buchner, and A.-G. Lang, "Statistical power analyses using g*power 3.1: Tests for correlation and regression analyses," *Behavior research methods*, vol. 41, pp. 1149–60, 2009.

[46] T. Dinev, H. Xu, H. J. Smith, and P. J. Hart, "Information privacy and correlates: An empirical attempt to bridge and distinguish privacy-related concepts," *European Journal of Information Systems*, vol. 22, pp. 295–316, 2013.

[47] N. K. Malhotra, S. S. Kim, and J. Agarwal, "Internet users' information privacy concerns (IUIPC): The construct, the scale, and a causal model," *Information Systems Research*, vol. 15, pp. 336–355, 2004.

[48] H. J. Smith, S. J. Milberg, and S. J. Burke, "Information privacy: Measuring individuals' concerns about organizational practices," *MIS Quarterly*, vol. 20, pp. 167–196, 1996.

[49] Android Developer Guide, "Request app permissions," https://developer.android.com/training/permissions/requesting, accessed: 2021-05-26.

[50] Apple Developer Guide, "Requesting permissions," https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/requesting-permission/, accessed: 2021-05-26.

[51] Material Design, "Dialogs," https://material.io/components/dialogs, accessed: 2021-05-26.

[52] ——, "Writing," https://material.io/design/communication/writing.html, accessed: 2021-05-26.

[53] NLTK, "Categorizing and tagging words," https://www.nltk.org/book/ch05.html, accessed: 2021-05-26.

[54] "Guidelines for academic requesters," https://www.yumpu.com/en/document/read/31225336/guidelines-for-academic-requesters, accessed: 2021-05-26.

[55] R Core Team, *R: A Language and Environment for Statistical Computing*, 2020. [Online]. Available: https://www.R-project.org/

[56] D. Bates, M. Mächler, B. Bolker, and S. Walker, "Fitting linear mixed-effects models using lme4," *Journal of Statistical Software*, vol. 67, pp. 1–48, 2015.

[57] H. Aguinis, R. K. Gottfredson, and H. Joo, "Best-practice recommendations for defining, identifying, and handling outliers," *Organizational Research Methods*, vol. 16, pp. 270–301, 2013.

[58] G. Cumming and S. Finch, "Inference by eye: Confidence intervals and how to read pictures of data," *American psychologist*, vol. 60, p. 170, 2005.

[59] S. E. Schechter, R. Dhamija, A. Ozment, and I. Fischer, "The emperor's new security indicators," in *18th IEEE Symposium on Security and Privacy (SP'07)*, 2007.

[60] D. Akhawe and A. Porter Felt, "Alice in warningland: A large-scale field study of browser security warning effectiveness," in *22th USENIX Security Symposium (SEC'13)*, 2013.

[61] C. Bravo-Lillo, S. Komanduri, L. F. Cranor, R. W. Reeder, M. Sleeper, J. S. Downs, and S. E. Schechter, "Your attention please: Designing security-decision UIs to make genuine risks harder to ignore," in *9th Symposium on Usable Privacy and Security (SOUPS'13)*, 2013.

[62] Apple Developer Guide, "Requesting access to protected resources," https://developer.apple.com/documentation/uikit/protecting_the_user_s_privacy/requesting_access_to_protected_resources, accessed: 2021-05-26.

[63] P. G. Kelley, J. Bresee, L. F. Cranor, and R. W. Reeder, “A "nutrition label" for privacy,” in *Proc. 5th Symposium on Usable Privacy and Security (SOUPS’12)*, 2009.

[64] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor, “Crying wolf: An empirical study of SSL warning effectiveness,” in *Proc. 18th USENIX Security Symposium, (SEC’09)*, 2009.

[65] S. Egelman, L. F. Cranor, and J. I. Hong, “You’ve been warned: an empirical study of the effectiveness of web browser phishing warnings,” in *Conf. on Human Factors in Computing Systems (CHI’08)*, 2008.

[66] M. Harbach, M. Hettig, S. Weber, and M. Smith, “Using personal examples to improve risk communication for security & privacy decisions,” in *Conf. on Human Factors in Computing Systems (CHI’14)*, 2014.

[67] I. Liccardi, J. N. Pato, D. J. Weitzner, H. Abelson, and D. D. Roure, “No technical understanding required: Helping users make informed choices about access to their personal data,” in *11th International Conference on Mobile and Ubiquitous Systems (MOBIQUITOUS’14)*, 2014.

[68] P. G. Kelley, L. F. Cranor, and N. M. Sadeh, “Privacy as part of the app decision-making process,” in *ACM SIGCHI Conference on Human Factors in Computing Systems (SIGCHI’13)*, 2013.

[69] S. Frey, “New safety section in Play will give transparency into how apps use data,” https://android-developers.googleblog.com/2021/05/new-safety-section-in-google-play-will.html, accessed: 2021-05-26.

[70] H. Aguinis and K. J. Bradley, “Best practice recommendations for designing and implementing experimental vignette methodology studies,” *Organizational Research Methods*, vol. 17, pp. 351–371, 2014.

[71] D. J. Woehr and C. E. Lance, “Paper people versus direct observation: An empirical examination of laboratory methodologies,” *Journal of Organizational Behavior*, vol. 12, pp. 387–397, 1991.

[72] R. Hughes and M. Huby, “The application of vignettes in social and nursing research,” *Journal of advanced nursing*, vol. 37, pp. 382–6, 2002.

A Study Procedure

This section lists the questions of the survey in the same order they were shown to participants. Note that Sections A.1 and A.2 are repeated four times per participant.

A.1 Pre-Questionnaire

The {first/second/third/last} app of interest is called {app name}. Imagine the following scenario: You have recently installed the {app name} app on your phone. {sentence describing the major functionalities of the app}. You want to use this app to {objective to use the app}.

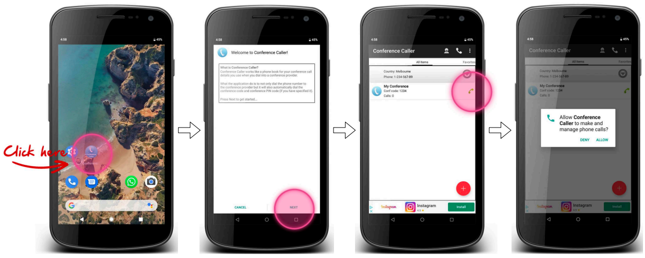
[Show a screenshot of the homescreen with the app icon.]

App Familiarity (Familiarity): Have you used this app before? (i) Yes (ii) No (iii) Do not know.

Permission Predictability ($Perm_{Predict}$): Would you expect this app to request access to your {permission protected resource}? (i) Yes (ii) No.

Permission Sensitivity ($Perm_{Sens}$): When using mobile apps, many people find that there are some resource accesses (permissions) that they are generally comfortable granting, some accesses that they are only comfortable granting under certain conditions, and some accesses are too sensitive that they never or only rarely are comfortable granting. Given the information that this app will request access to your {permission protected resource}. Please indicate to what extent you agree or disagree with the following statements. (i) In general, I do not feel comfortable granting access to my {permission protected resource} (ii) I feel that this app requires access to a very private resource (iii) The access to my {permission protected resource} is very sensitive to me.

Permission Clarity before app interaction ($Clarity_{Pre}$): (i) I understand the reason for this app to request access to my {permission protected resource} (ii) I have no idea why this app wants access to my {permission protected resource} (iii) It is clear to me why this app needs access to my {permission protected resource}.



What is Conference Caller?
Conference Caller works like a phone book for your conference call details you use when you dial into a conference provider.

What the application do is to not only dial the phone number to the conference provider but it will also automatically dial the conference code and conference PIN code (if you have specified it).

Press Next to get started...

Figure 8: Sample interactive mockup app interaction

A.2 Post-Questionnaire

Now, imagine that you downloaded {app name} on your phone to {objective to use the app}. Below this text is an interactive mockup app of {app name}. Please interact with the app as you would on your own phone until access to your {permission protected resource} is requested. You can repeat your interaction with the app by clicking the reset button. Then answer the following questions.

[Show interactive mockup app same as in Figure 8.]

Permission Decision (Decision): Based on your interaction with this app, would you grant this app access to your {permission protected resource}? (i) Yes (ii) No.

Permission Purpose ($Perm_{Purp}$): In your opinion, for what does this app need access to your {permission protected resource}? (i) For the main functionality of the app (app cannot function without it) (ii) For some additional feature functionality (iii) Do not know (iv) For some other reason.

Permission Clarity after app interaction ($Clarity_{Post}$): After interacting with the above mockup app, please indicate to what extent you agree or disagree with the following statements. (i) I understand the reason for this app to request access to my {permission protected resource} (ii) I have no idea why this app wants access to my {permission protected resource} (iii) It is clear to me why this app needs access to my {permission protected resource}.

Only for requests with rationales: Rationale Origin (Rationale$_{Origin}$): Who do you think provided the explanatory message *"This app requires access to your {permission protected resource} to..."* that was displayed in a separate dialog immediately before requesting access to your {permission protected resource}? (i) The mobile operating system (ii) The app developer (iii) Do not know (iv) Some other entity.

Decision Evaluation Scales (DES): In a previous question you chose to {allow/deny} this app access to your {permission protected resource}. We would like to know how you feel about this decision. Please state to what extent you agree or disagree with the following statements.

Decision Satisfaction (Des$_{Satis}$): (i) I expect to stick with my decision (ii) I am satisfied with my decision (iii) I am doubtful about my choice (iv) I would make the same decision if I had to interact with this app again.

Informed Decision (Des$_{Inform}$): (i) I am satisfied with the information I received (ii) I know the pros and cons of granting this app access to my {permission protected resource} (iii) I would have liked more information about how the app will use the access to my {permission protected resource} (iv) I made a well-informed choice.

Decision Control (Des$_{Control}$): (i) I felt pressured by the app to make this decision (ii) The app allowed me to make my own decision (iii) I feel that the app forced me to make this decision (iv) This was my own decision.

Only for requests with rationales: Rationale Recall (Rationale$_{Recall}$): While interacting with the {app name} app you saw a dialog explaining why the app needs access to your {permission protected resource}. It started with: *"This app requires access to your {permission protected resource} to ..."* Please complete this message as far as you remember. Note: The dialog we are asking you about is the one that immediately preceded the dialog in which you were asked to grant or deny access to your {permission protected resource}. Free response.

A.3 Demographics

We would like to ask you for some demographic information.

Mobile OS: What operating system are you using on your (primary) mobile phone? (i) Android (ii) iOS (iPhone) (iii) Windows (Windows Phone) (iv) Other.

Gender: Which gender do you identify most with? (i) Male (ii) Female (iii) Prefer not to say (iv) Other.

Age: In what year where you born? Drop-down list.

Education: What is the highest degree or level of education you have completed? (i) Some school, no degree (ii) High school graduate (iii) College, no degree (iv) Bachelor's degree (v) Master's degree (vi) Professional degree (vii) Doctorate degree.

Computer Science Background: Are you studying or have you been working in any of the following areas: information technology, computer science, electronic data processing, electrical engineering, communications technology, or similar? (i) Yes (ii) No.

Privacy Concerns (Priv$_{Conc}$): (i) Compared to others, I am more sensitive about the way mobile apps handle my personal information (ii) To me, it is the most important thing to keep my privacy intact from mobile apps (iii) In general, I am very concerned about threats to my personal privacy.

Prior Privacy Experience (Prior$_{Exp}$): (i) How often have you personally experienced incidents whereby your personal information was used by some mobile app without your authorization? (ii) How much have you heard or read during the last year about the use and potential misuse of the information collected from mobile apps? (iii) How often have you personally been the victim of what you felt was an improper invasion of your privacy from a mobile app?

B Participant Demographics

Number of Participants	473	
Gender		
Male	296	62.6%
Female	174	36.8%
Other	3	0.6%
Age		
18–23	20	4.2%
24–30	128	27.1%
31–40	184	38.9%
41–50	78	16.5%
51 and over	63	13.3%
Education		
Up to high school	54	11.4%
Professional school degree	6	1.3%
Some college (no degree)	83	17.5%
Bachelor's degree	243	51.4%
Graduate degree	87	18.4%
Mobile OS		
Android	330	69.8%
iOS	134	28.3%
Other	9	1.9%
Computer Science Background		
Yes	176	37.2%
No	297	62.8%

C Model fit

We statistically compared all steps in the model building process using the akaike information criterion (AIC) and the likelihood-ratio tests. The model that described our data best and had the lowest AIC score was selected as the final model. To remain consistent with the theoretical design of our study, we included the variables of interest (step 3) for the Des$_{\text{Control}}$ model even if this step was not significant. Table 2 present the goodness of fit, marginal R^2 and conditional R^2 for each step in the model building process of all outcome variables.

Table 2: Goodness of fit for final models

Decision Model	AIC	LogLik	Df	Pr(>Chisq)	Marginal R^2	Conditional R^2
simple regression	2328.14	-1163.07				
step 1: multilevel base (app and user as random effects)	1955.97	-974.98	2	<0.001		0.590
+ step 2: variables from previous work	1487.60	-734.80	6	<0.001	0.462	0.733
+ step 3: variables of interest: timing and rationales	1449.35	-713.68	2	<0.001	0.483	0.765
+ step 4: interaction(timing:rationales)	1451.35	-713.68	1	0.986	0.483	0.765
$\text{Des}_{\text{Inform}}$ Model						
simple regression	6290.77	-3143.39				
step 1: multilevel base (app and user as random effects)	6013.84	-3002.92	2	<0.001		0.354
+ step 2: variables from previous work & Decision	5746.44	-2862.22	7	<0.001	0.180	0.430
+ step 3: variables of interest: timing and rationales	5647.17	-2810.59	2	<0.001	0.207	0.470
+ step 4: interaction(timing:rationales)	5633.44	-2802.72	1	<0.001	0.211	0.476
$\text{Des}_{\text{Satis}}$ Model						
simple regression	5500.03	-2748.02				
step 1: multilevel base (app and user as random effects)	4921.63	-2456.82	2	<0.001		0.533
+ step 2: variables from previous work & Decision	4704.05	-2341.02	7	<0.001	0.194	0.544
+ step 3: variables of interest: timing and rationales	4702.12	-2338.06	2	0.052	0.196	0.546
+ step 4: interaction(timing:rationales)	4695.43	-2333.72	1	<0.01	0.198	0.549
$\text{Des}_{\text{Control}}$ Model						
simple regression	6343.33	-3169.67				
step 1: multilevel base (app and user as random effects)	5350.00	-2671.00	2	<0.001		0.676
+ step 2: variables from previous work & Decision	5245.12	-2611.56	7	<0.001	0.134	0.677
+ step 3: variables of interest: timing and rationales	5243.57	-2608.78	2	0.062	0.136	0.679
+ step 4: interaction(timing:rationales)	5243.39	-2607.69	1	0.139	0.136	0.679
$\text{Clarity}_{\text{Post}}$ Model						
simple regression	7775.5	-3885.75				
step 1: multilevel base (app and user as random effects)	7401.07	-3696.54	2	<0.001		0.314
+ step 2: variables from previous work	6561.61	-3270.80	6	<0.001	0.470	0.512
+ step 3: variables of interest: timing and rationales	6424.99	-3200.50	2	<0.001	0.502	0.559
+ step 4: interaction(timing:rationales)	6418.44	-3196.22	1	<0.01	0.504	0.562

D User study apps

App	Perm.	Perm. purpose	Goal to use the app *(You want to use this app to . . .)*	Rationale message *(This app requires access to your. . .)*
TextDrive[1]	contacts	visible	block phone calls of some contacts while you're driving.	contacts to display caller names, and block selected contacts.
Conference Caller[1]	phone	main	have a conference call with your work colleagues.	phone to make conference calls.
SContact[1]	phone	hidden	exchange contact information with your business partners.	phone to read device id to uniquely identify your device.
Meteor[2]	location	visible	compare network speed of different locations.	location to show your network accesses on map.
Wifi Time Tracker[2]	location	main	keep track of your working hours.	location to scan for nearby Wi-Fi networks.
AmazeVPN[2]	storage	hidden	use vpn while browsing.	photos, media, and files to manage cache of app data on SD card.
EOS[3]	location	visible	order some delicious sandwiches.	location to find EOS restaurants nearby, and show your location on map.
Cookiegazm[3]	location	main	order food from Cookiegazm.	location to find Cookiegazm restaurants nearby.
Pancakes[3]	storage	hidden	find the next stampede pancake breakfast event.	photos, media, and files to manage cache of app data on SD card.
FaceSwap [4]	mic.	visible	record a video of you and your friend with your faces swapped.	microphone to record face swapped videos with audio.
Beauty Cam[4]	camera	main	take a beautiful selfie.	camera to display stickers on camera view, and take selfies.
Free Fonts for Samsung[4]	phone	hidden	get new fonts for your phone.	phone to read device id to uniquely identify your device.
All Meter[5]	mic.	visible	measure the sound level of your voice.	microphone to measure sound levels in dB.
Loopback[5]	mic.	main	measure the round-trip latency of your voice.	microphone to measure round-trip audio latency.
Tractor Guide[5]	storage	hidden	mark which field areas you have already covered with fertilizer.	photos, media, and files to manage cache of app data on SD card.
Belize Radio World[6]	mic.	visible	record your own channel.	microphone to record your own audio program.
Strobily[6]	camera	main	make your phone's flashlight sync to music.	camera to turn on flashlight.
Cambodian Radio[6]	location	hidden	listen to music.	location for targeted advertisement.
NT Hunting Mate[7]	storage	visible	report an illegal hunting activity.	photos, media, and files to store uncompleted reports.
GoldHunt Free[7]	location	main	find a hidden geocache nearby.	location to show your location on map, and find unfound caches nearby.
Trout Fly Fishing[7]	storage	hidden	learn how to tie a fly.	photos, media, and files to store cache of app data for better performance.
My Weirton[8]	location	visible	report a pothole in Weirton.	location to find reported issues nearby, and show current location on map.
SkyPointer[8]	location	main	find the current position of the ISS in the sky.	location to autocomplete your current location and coordinates.
Monroeville Chamber[8]	storage	hidden	find opening times of the museums in Monroeville.	photos, media, and files to download app content to SD card.
Vehi Care[9]	storage	visible	backup your vehicle's data.	photos, media, and files to store backups of your car data to SD card.
OpenMBTA[9]	location	main	find the closest train station nearby.	location to find train stations nearby, and show your current location on map.
ELCO Chevrolet Cadillac[9]	storage	hidden	buy a used car.	photos, media, and files to download app content to SD card.
Dinosaur Photo Wallpapers[10]	camera	visible	take a selfie with a dinosaur frame.	camera to display dinosaur frames on camera view, and take pictures.
Ice Cream Wallpapers[10]	storage	main	set an ice cream wallpaper as your phone's background.	photos, media, and files to download wallpapers to SD card.
Roses[10]	phone	hidden	send a rose picture to your friend.	phone to read device id to uniquely identify your device.

[1]commun., [2]connection, [3]delivery service, [4]design and art, [5]measurem. tools, [6]music and sound, [7]outdoor activities, [8]places and stars, [9]vehicles and transport., [10]wallpapers