

## Research Article

# Identifying Usability Issues in Instant Messaging Apps on iOS and Android Platforms

Sergio Caro-Alvaro , Eva Garcia-Lopez , Antonio Garcia-Cabot , Luis de-Marcos , and Jose-Javier Martinez-Herraiz

Computer Science Department, University of Alcalá, Alcalá de Henares, Spain

Correspondence should be addressed to Sergio Caro-Alvaro; [sergio.caro@uah.es](mailto:sergio.caro@uah.es)

Received 19 April 2018; Revised 17 July 2018; Accepted 27 August 2018; Published 10 October 2018

Academic Editor: Salvatore Carta

Copyright © 2018 Sergio Caro-Alvaro et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rise of mobile devices and the high number of instant messaging applications available in the stores, it is necessary to evaluate the usability of such applications to provide a more satisfying user experience. To this end, this paper presents a methodical usability evaluation of instant messaging applications both in iOS and Android platforms. A predefined evaluation was used, which was created to detect the main usability issues of mobile applications, regardless of the device used and the evaluated applications. Consequently, two methods were used: the keystroke level model and the mobile heuristic evaluation. Also, the results suggest that the main problems of this type of applications are difficulties in performing tasks (some of them were not agile nor easy to complete), lack of element cohesion (some icons and buttons did not follow the style of the operating system, bad translations, and too much information on the screen), problems with the user interface (pop-ups overlapping the status bar, clipped elements, sometimes the interface did not rotate, and, in other cases, the interface changed considerably when the device was rotated), and lack of information about privacy and security features. Finally, based on the results, we propose a set of recommendations that will be helpful for developers of this kind of applications.

## 1. Introduction

Mobile devices are considered as a valuable resource in our daily lives, or even essential, since they are the most used electronic tool [1, 2]. This is also supported by the high number of active devices operating throughout the last years. Back in 2012, there were around 640 million of mobile devices [3], whereas, by the end of 2017, there were reported 2,890 million worldwide smartphones [4].

The number of devices and available mobile applications (apps) seem to have been growing exponentially and simultaneously in recent years. The Apple App Store reported an increase of more than 400,000 available apps between 2009 and early 2012 [5], while Google Play reported that there were more than 700,000 available apps on late 2012 [6]. In July 2015, the available apps increased up to 1,600,000 in Google Play and up to 1,500,000 in the Apple App Store [7]. As of January 2017, there were 2,200,000 apps in the Apple

App Store [8]. By the end of December 2017, there were reported 3,500,000 apps in Google Play [9]. This means that there are many alternative apps to perform a given task, so users are supposed to choose the best app, motivated by several facts like ease of use, low learnability curve, or low time consumption [10].

Mobile instant messaging (MIM) apps are becoming highly popular in recent years, as a technological (and free, in some cases) evolution of short messages (namely, SMS) [11–13], providing a mean to enhance social interrelationships [12]. According to a research performed in late 2013 [14], only in China there were about 1.48 billion accounts in instant messaging (IM) services. For instance, “WeChat” and “WhatsApp” apps had around 600 million registered users each; whereas 7 billion daily messages are sent in “Line” and 10 billion messages were sent daily in “WhatsApp.” In 2017, there are reported 1.82 billion of IM app users worldwide, and it is expected to grow up to 2.48 billion users in 2021

[15]. Thus, IM apps are more likely to be used by a larger mass of users and, consequently, they are getting an increasing importance. Therefore, an analysis which is putting all these elements together, alongside with user-based interaction elements, becomes mandatory in order to improve the users' quality usage of these tools.

HCI field is focused on how people interact with computers in order to create better products [16]. To this end, HCI is widely applied to try to understand users' perceptions while using some specific software products. In other words, it seeks to improve the User eXperience (UX). As defined by the International Organization for Standardization (ISO), UX is defined as "*person's perceptions and responses resulting from the use and/or anticipated use of a product, system or service*" [17]. In other words, UX is focused on the human emotions while using a specific software product, trying to produce high-quality experiences to the user [18, 19].

Given that UX is a broader term (applied to almost all experiences involving the usage of a product), in HCI domain, more focused, the usability engineering is known to be part in software life cycle, connected to the design phase. Thus, it is a field of study to improve products' designs. But these improvements are not only related to usability. They could be also applied to the UX, given the UX definition by ISO, where UX and usability are connected: "*Note 3 to entry: Usability, when interpreted from the perspective of the users' personal goals, can include the kind of perceptual and emotional aspects typically associated with user experience. Usability criteria can be used to assess aspects of user experience*" [17].

Usability, a software quality attribute, is defined by ISO as "*the effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments*" [20], whereas Nielsen defines it as "*a quality attribute that assesses how easy user interfaces are to use*" [21]. Usability is recognized as the key of success for software [22], despite the fact that it is not considered as fully suitable for a better UX [23]. Anyhow, why usability, in any kind of software, is so important? It is simple [21, 24]. If users face any kind of setback, they are prone to leave. Mainly, because there are plenty of alternatives at their disposal. In words of [25], a poor usability leads to a hugely lower performance of the user with the given software. These ideas could be summarized as follows: users do not tolerate UX problems anymore. All of these could lead to the user-centered design (UCD), a UX methodology applied to increase product usability. UCD focus on the design process, from the point of view of user requirements and goals [26]. UCD promotes prioritizing design prototyping in order to meet users' expectations, rather than software features.

When it comes to evaluate the usability of a given software product, there is a plethora of alternatives, usually divided into three groups [27, 28]: empirical methods, inspections methods, and inquiry methods. Empirical methods (for example, user performance tests or beta tests) consist on retrieving user's experiences with the use of the product. Inspection methods (like heuristic evaluations, expert reviews, or cognitive walkthroughs) involve usability

experts to analyze usability features of a given product user interface. Finally, inquiry methods (for example, user satisfaction questionnaires, field observations, or interviews) try to obtain users' opinions and requirements based, mainly, on observations and by talks with users.

Nevertheless, the main limitations to measure usability in mobile applications are the special characteristics of mobile devices [29]: small screen size, input capabilities, network limitations, working with batteries, and variety of hardware and software elements, among others. Briefly, that is why traditional methodologies do not work well with mobile devices and, thus, new (or, at least, different from PC approaches) evaluations must be applied to mobile applications, in order to obtain optimal usability results.

On the other hand, creating an IM app and getting people to use it is not a complicated process. What it is a difficult task is to keep users to continue using the service because the cost of switching between applications is very low [13, 30]. Therefore, this is why it is so vital to evaluate and improve the usability of this type of apps.

In this paper, a methodical usability methodology is used to evaluate MIM apps, both on iOS and Android platforms, and to suggest a list of recommendations that will do well to improve the usability of this type of mobile applications. Firstly, the activities that characterize common interactions in an IM app are identified. Then, efficiency is measured in terms of the number of interactions to complete the activities. Finally, it is performed a heuristic evaluation with mobile experts.

The rest of the paper is structured as follows: Section 2 presents the theoretical description underlying the methodical methodology. Section 3 presents the results for the methodical evaluation performed on mobile IM applications. Also, a statistical analysis is applied to determine which values are statistically significant. Section 4 discusses results and related work and Section 5 proposes a list of usability recommendations. Finally, Section 6 presents some conclusions.

## 2. Material and Methods

As far as the authors know, there are no so many methodologies created to evaluate the usability of certain types of mobile apps. There are, indeed, a lot of research to evaluate specific under-test apps, that is, the app is already selected before the evaluation. However, when facing studies like the one presented here, where the apps to evaluate are elected based on a series of formal conditions (i.e., apps are selected within the evaluation process), possible methodologies to choose are reduced.

In the work of Moumame [31], although being reported to be a methodology to evaluate usability of any kind of mobile apps regardless to the device, in fact it is an user satisfaction questionnaire (based, however, on an ISO standard) after some tasks are completed by participants. Those tasks are supposed to be the most frequent and used tasks in given apps, but there is not a formal procedure to determine them. Thereby, it is quite unclear how to proceed with that methodology.

There are other types of evaluations that apply formal steps when evaluating the usability of a given app, combining both usability inspections and user tests. Namely, they are referred to as systematic usability evaluations (SUE) (defined in [32], but used within a variety of scopes). In SUE, apps are analyzed at different levels with tasks, as this paper does with IM apps. Nonetheless, the main limitation of SUE method is the same as stated previously: this type of methodologies is only created to evaluate preselected apps, but not to determine how to evaluate a specific type of app from the market.

The applied methodology uses a set of heuristics as one of the inspection methods. Due to the peculiarities of mobile devices, such as small screen size or touch screens, several studies [33–36] concluded that traditional heuristics do not work well with mobile apps and, therefore, that it is necessary to develop and use heuristics specially designed for mobile devices. There are, indeed, alternatives to analyze mobile apps with heuristics. For example, in the work of Inostroza [37], they developed a set of heuristics for smartphones and mobile apps. They reported an improvement of the original Nielsen's set, although the validation of the heuristics was not fully provided. In a recent study [38], they provided an early validation of their heuristics, but they reported more work to do to fully validate the heuristics.

Thus, it was decided to apply the methodology of Martin, Flood, and Harrison [39]. It is a versatile and methodical methodology conceived to detect the main usability issues of mobile apps. It was designed to be independent from the type of the app analyzed, in order to be adequate for the special characteristics of mobile devices. This methodology has some advantages, apart from being the one suitable for analyzing specific types of apps, starting the evaluation at the market. For example, it is independent from both the platform and domain where it is going to be applied. Also, it is economic: except the heuristic evaluation, almost all steps could be carried out by only one person. Even the experts, in the HE, are only required to analyze a bunch of apps, so their required time is low. Economy could also be seen in the definition of the main tasks, which highly reduce the number of apps to be analyzed. Not only independence and economy are advantages of this methodology but also formal steps, in the way they are defined and applied, are an advantage, as it provides seriousness to the outcome. The main limitation of the applied methodology could be seen in the lack of tests with real users. “*Heuristic Walkthroughs*” and “*Contextual Walkthroughs*” could address this issue since these techniques involve context. Nevertheless, these alternatives have been reported to be more time consuming, dreary in execution, and not covering all aspects [40]. This methodology consists of five steps, which have to be performed sequentially:

- (i) *Step 1. “Identify all potentially relevant applications.”*  
Search for applications in online stores based on a given search keyword.
- (ii) *Step 2. “Remove light or old versions of each application.”* Remove nonfully functional apps (demos or

trials) as well as old versions to evaluate only the full/current version.

- (iii) *Step 3. “Identify the main functional requirements and exclude all applications that do not offer this functionality.”* Define the essential functionalities for the type of app under evaluation, that is, MIM apps. Apps that do not meet all requirements have to be excluded. This step acts as a diagnostic method, since it is a way to classify the apps and remove those unrelated to the analyzed context.
- (iv) *Step 4. “Identify all secondary requirements.”* Determine what additional (secondary) functionalities are provided in the analyzed applications. This step does not remove any of the apps found so far.
- (v) *Step 5.* In this step, usability of the primary tasks (functionalities defined in Step 3) is measured, by using two methods:
  - (a) *Step 5-A. “Keystroke Level Modelling (KLM).”* The number of interactions to complete the main tasks is counted. This acts as a measure of efficiency (as a way to verify a good performance of the app), where fewer interactions mean better efficiency. Therefore, apps with lower total values are qualified for the next step.
  - (b) *Step 5-B. “Heuristic evaluation.”* Selected apps are evaluated according to a set of mobile usability heuristic (MUH). This evaluation consists of determining if the app met, or not, a set of guidelines (namely, heuristics). In this context, a heuristic could be generally defined as a rule of thumb applied as a criterion to evaluate user interfaces. The assistance of human mobile experts is required in order to analyze the apps using a set of heuristics that are scored in a 5-point Likert scale. It is worth mentioning that these applied heuristics are neither generic nor the traditional ones, since they do not work well with mobile devices. The set applied here were specifically created for mobile devices [29] and successfully applied in previous studies with mobile apps [39, 41–43].

### 3. Results

In this section, the results obtained in each step of the usability evaluation on iOS and Android platforms are detailed. An iPhone (with iOS 8 version) and a Samsung Galaxy Nexus (with Android 4.4 version) were used for this study. The iOS-platform data were collected during the Fall semester of 2014, whereas data for the Android platform were retrieved during the Spring semester of 2015. Figure 1 shows the number of (remaining and discarded) apps in each step of the research. Within Results section, when evidences of usability flaws are found, they are linked to the specific recommendation in Section 4, where justifications are presented.

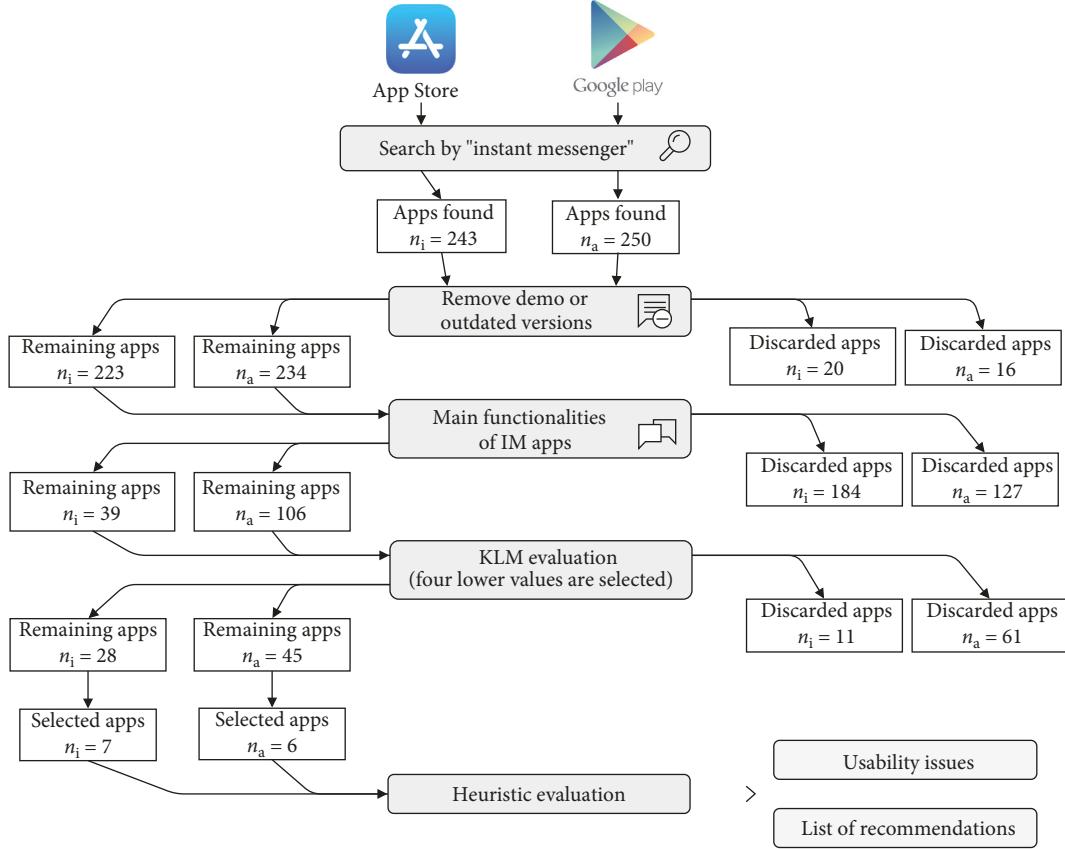


FIGURE 1: Flow diagram presenting number of apps in each stage of the research.

**3.1. Step 1: Identify All Potentially Relevant Applications.** In this first step, a search term was chosen, in such a way that included as many applications as possible. The search term selected, which was used in both markets (i.e., the App Store and the Google Play), was “instant messenger.” This keyword was derived from an analysis of the main characteristics provided in the descriptions of the most popular IM apps in the store (i.e., WhatsApp, Telegram, LINE, and Viber, among others). Thus, keyword was used to locate and retrieve information of the existing applications matching the search term. As a result, 243 applications were found on the iOS market and 250 applications were obtained on the Android platform. Hence, all these applications were potentially relevant and were used as an input for the following steps of the study.

**3.2. Step 2: Remove Light or Old Versions of Each Application.** According to the methodology followed, if an application is categorized as a demo, trial, or an old outdated version, it cannot be analyzed because it is not fully functional. These applications were thus removed from the original list of potential applications. As a result, twenty iOS applications (8%) and sixteen Android applications were discarded (6%).

In agreement with the results, in the iOS platform, all discarded applications were apps with limited functionality. However, in the Android platform, eight applications

offered limited functionality, three were old versions, two were beta versions, and two were modules for another application. Also, one app was discarded from the Android list because it was removed from the store in the meantime.

**3.3. Step 3: Identify the Primary Operating Functions and Exclude All Applications That Do Not Offer This Functionality.** In this step, the main functionalities that all IM apps should provide were extracted from the literature. According to [44–46], it could be assumed that sending and reading instant messages are essential features for this type of apps. Indeed, IM apps also use contacts [44, 46], so adding new contacts should be essential for the user to communicate with other (new) contacts. For security and privacy warranties, deleting contacts [47] or blocking features [45, 48] should also be implemented. Finally, for privacy or managing internal storage, in some cases, the users may want to delete specific conversations [49, 50]. Therefore, IM applications should provide features to perform the following activities:

- (i) **Task 1 (T1): send a message.** Send an instant message to a specific contact.
- (ii) **Task 2 (T2): read and Reply.** Read an incoming instant message and reply to it.
- (iii) **Task 3 (T3): add a contact.** Add a new contact to the agenda.

- (iv) Task 4 (T4): delete (or block) a contact. Delete a specific contact (or block it, if deletion is not supported).
- (v) Task 5 (T5): delete chats. Clear specific conversations.

Once the main functionalities were defined, the applications that did not meet all the features were then removed. Thereby, each application was tested to check its main functionalities and, as an output, a new list of IM applications was obtained. As a result, 184 applications (82.51%) were discarded in the iOS platform, so 39 apps (17.49%) remained for the next step. As for the Android platform, 127 applications (54.51%) were discarded, so 106 apps (45.49%) remained.

Taking a closer look to Figure 2, it compares the apps that met each given number of functionalities (5 to 0) on both platforms. On Android, nearly a half of the apps were fully functional and about a half of them were discarded. On iOS, more than 80% of the apps were discarded, whereas less than 20% of the apps were fully functional. It should be highlighted that, on both platforms, between 45% and 55% of the apps did not meet any of the main functionalities, as they were apps designed for IM apps (like text editors, photo enhancers, or emoticons), but not really IM apps. Furthermore, for iOS platform, it is also notable that the number of apps that met 5 functionalities (17.57%) is very similar to those which met 3 functionalities (16.67%), since the latter were social networks' IM apps without possibility to manage contacts. In conclusion, there were almost three (2.72) times more IM applications (i.e., apps meeting all the 5 functionalities) on the Android platform (45.49%) than on the iOS platform (17.57%).

It should be noted that more than a half (55.41%) of the iOS apps on the first step did not meet any of the five defined functionalities. This was motivated by the fact that, given the selected search term, the iOS market retrieved multiple apps related to IM apps that were not really IM apps, for example, modules for social networks or multimedia plugins (text visual enhancers, photo/video editors, and emoji).

Furthermore, while carrying out this step, several applications were also removed because their (bad) performance hindered a full evaluation, for example, unrecoverable errors (crashes), registrations that did not work, or apps that could not be opened. In total, 24 apps (from the 127 discarded) were removed in Android for these reasons and 5 apps (from the 184 discarded) were removed in iOS due to these causes.

Within this initial inspection of the main functionalities in the raw list of apps, several usability problems were discovered. Hence, three main issues could be highlighted:

- Using too small font size texts that made impossible to use the app. Figure 3 shows an example of a pop-up in which texts and input messages were very difficult to read.
- Annoying and continuous pop-ups and notifications that, in some case, blocked the app or even the device. Figure 4 shows four examples of apps that made an indiscriminate use of pop-ups/notifications.

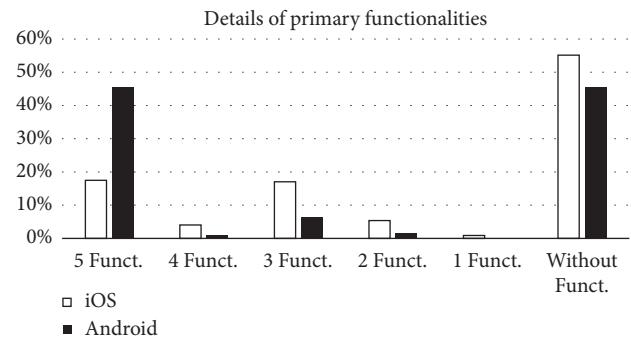


FIGURE 2: Number of main functionalities met by the apps found in both platforms.

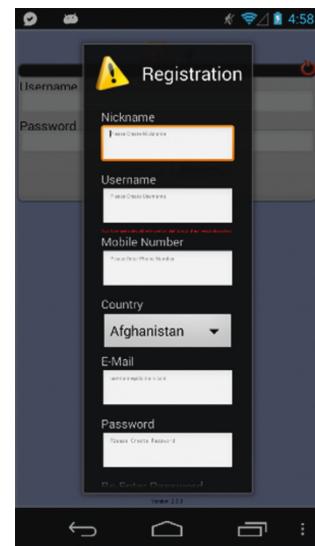


FIGURE 3: Details of too small texts.

- Clipped texts that were not properly displayed hinder the readability of the text. In this respect, Figure 5 shows examples of clipped texts (headers of the input texts and text in the buttons) that were difficult to read.

**3.4. Step 4: Identify All Secondary Functionalities.** Within this step, all the applications were tested in order to discover which other additional (secondary) functionalities they included. Secondary functionalities are additional features, which were not identified in the literature review, so they could not be considered as main functionalities. This process determines which additional features are the most common in IM applications. This step does not entail removing applications. As an input, 39 applications were tested on the iOS platform, whereas 106 applications were analyzed on Android.

On this ground, most secondary functionalities were identical in both platforms (Figure 6), and variations were found only in the percentage of each functionality (considerably lower in the Android platform).

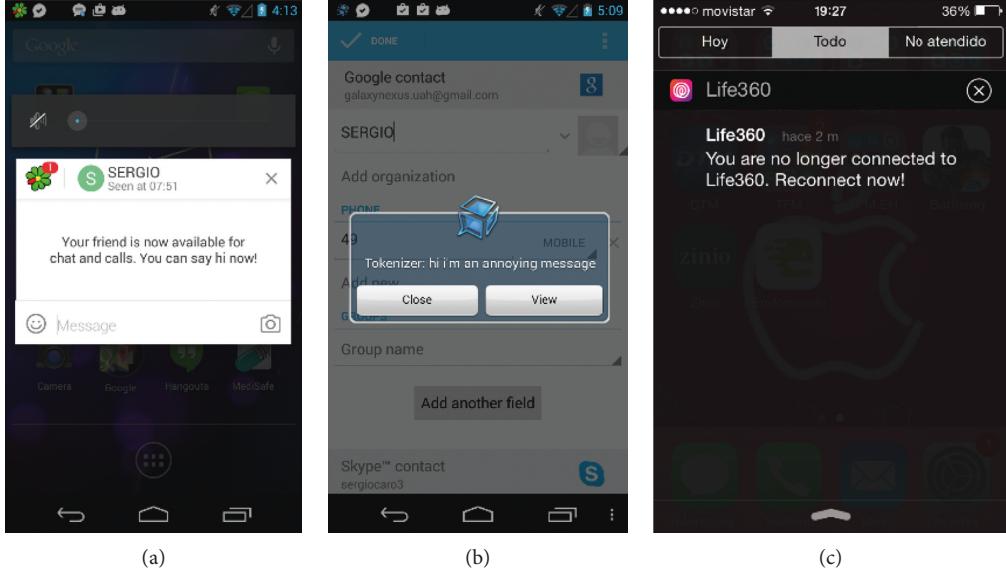


FIGURE 4: Examples of annoying and continuous messages.



FIGURE 5: Example of clipped texts.

Looking at the results, for the iOS platform, the most common functionalities were as follows: (1st) set a profile avatar (74.36%), (2nd) search in contacts (69.23%), (3rd) send a photo (66.67%), (4th) send a video (64.1%), (5th) set group chats (64.1%), and (6th) block contacts (53.85%), among others. Likewise, as for the Android platform, secondary functionalities were as follows: (1st) send a photo (37.74%), (2nd) search in contacts (34.91%), (3rd) set a profile avatar (30.19%), (4th) send a video (30.19%), (5th) set group chats (30.19%), and (6th) block contacts (28.30%), among others.

**3.5. Step 5-A: Keystroke Level Model.** In this step, the number of interactions required to complete each of the main functionalities (those defined in Section 3.3) was counted for each of the applications that was on the list after completing

Step 3 (identification of the main features). In KLM, as defined in the literature [51, 52], each tap on the screen (including each kind of interaction with the device, e.g., tap, pitch, scroll, swipe, drag, or the use of hardware buttons) was considered as one interaction, and the counting started when the app was just opened (indeed, the app was reopened for each task). Also, for sameness reasons, keyboard interactions (e.g., introducing the message to be sent) were determined to act as a unique interaction. It should be noted that, while analyzing these apps, some of them were discarded from the evaluation set, given their bad performance, crashes, and critical errors. For the iOS platform, 11 apps (out of the 39 selected apps) were discarded. Therefore, 28 apps were evaluated in this step. For the Android platform, 61 apps (out of the 106 selected apps) were discarded for the same reason. Hence, 45 apps were analyzed in this step.

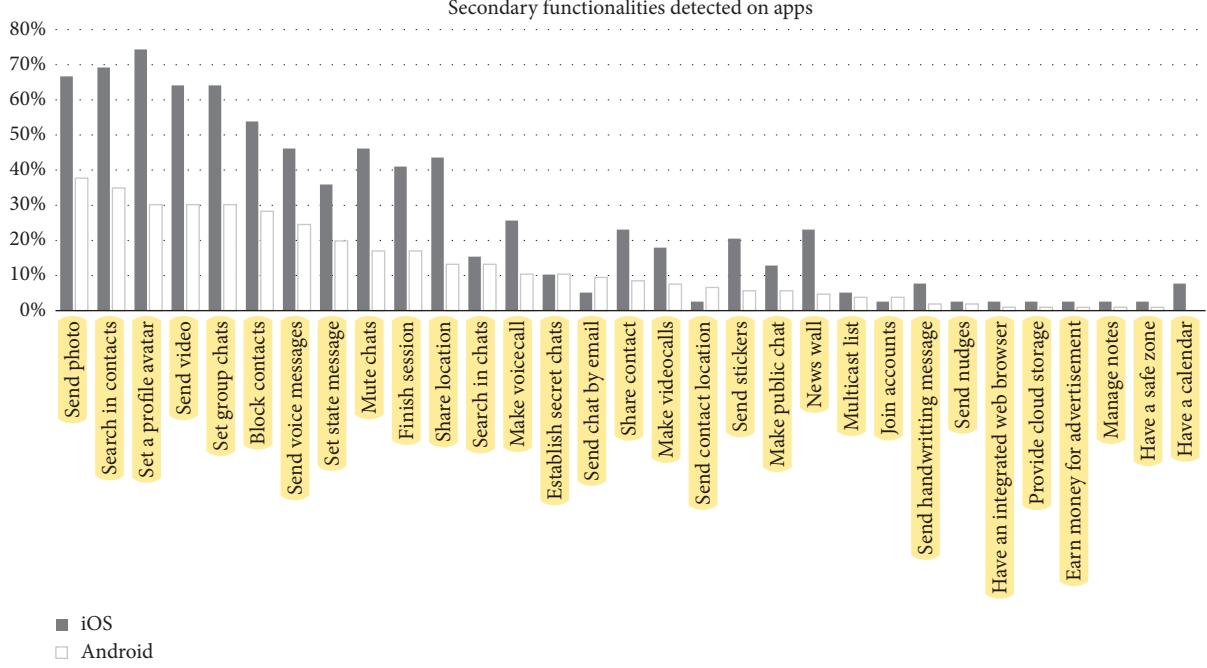


FIGURE 6: Secondary functionalities found on iOS and Android apps.

On the one hand, looking to the iOS results, the total average number of interactions was 29.86 (Table 1). When analyzing each task individually, task 1 (send a message) required an average of 6.53 interactions, task 2 (read and reply) needed an average of 5.75 interactions, task 3 (add a contact) required an average of 6.25 interactions, task 4 (delete or block a contact) needed an average of 5.96 interactions, and task 5 (delete chats) required an average of 5.35 interactions.

On the other hand, the total average number of interactions on Android was 29.4 (Table 1). Breaking down by each task, it should be noted that task 1 required 6.26 interactions on average, task 2 needed an average of 5.46 interactions, task 3 required an average of 7.62 interactions, task 4 needed 5.73 interactions on average, and task 5 required an average of 4.31 interactions.

Although the iOS apps required more number of interactions than the Android apps, it is worth pointing out that there were only 1.55% more interactions in iOS than in Android. Therefore, despite of being different platforms, it can be seen that the results were, actually, quite similar: the average number of interactions in iOS was around 6 interactions for all tasks, whereas for Android, the average value of interactions was between 4 and 7 interactions. Based on the compilation of the results, the main tasks should be agile to complete, avoiding any unnecessarily deep navigation. In practice, each task should require, as reported by [53],  $7 \pm 2$  interactions (see Recommendation #1 in Section 4). Taking a closer look at each task, some points about their performance could be remarked:

- (1) *Task 1: sending a new message.* Although both platforms presented, on average, a similar number of interactions when starting a new conversation, 4% more interactions (on average) were detected in iOS than in Android. In essence, the evaluated apps with

TABLE 1: KLM results for MIM apps on iOS and Android platforms. This table shows the mean number of interactions taken for each of the main functionalities.

Platform\Task	Mean					Total
	T1	T2	T3	T4	T5	
iOS	6.54	5.75	6.25*	5.96	5.36*	29.86
Android	6.27	5.47	7.62*	5.73	4.31*	29.40

\*Statistical significant difference was found when comparing both platforms.

fewest interactions could provide these lower values by showing the keyboard automatically when accessing to new chats (see Recommendation #2).

- (2) *Task 2: reply to an incoming message.* Similar to the first task, this one returned similar results for both platforms, although apps in iOS platform needed 5% more interactions than those in Android platform. Measurements showed between 4 and 6 interactions on average. These results were expected, given that almost all apps presented the active conversations grouped in a specific section.
- (3) *Task 3: adding a contact.* In essence, this task required 21.9% more interactions in the Android platform than those in the iOS platform. When examining the way this feature is implemented, it became apparent that there are multiple scenarios: it varies from 3 to 11 interactions. In short, some apps used the internal agenda of the device, whereas other apps established a custom contact list. As a result, the latter (usually) requires fewer interactions to achieve this task. Apart from this, the number of interactions was observed to escalate quickly when the app required more information to register a new contact

- (extra fields as, e.g., the name and the phone number) than the average (typically, only the username). Clearly then, this substantial difference is statistically significant ( $p = 0.0057$ ,  $W = 795$ ), and it is probably caused by the different ways in which platforms manage the task (see Recommendation #3). For example, in Android apps, the user must close the application, open the agenda, add there the contact, and then return to the application. This process is much easier in iOS apps for the user because communication with the agenda is done within the application without having to access the phonebook.
- (4) *Task 4: delete/block a contact.* Based on the results, very similar implementations of this feature were found for all apps. Hence, no major deviations were detected. Furthermore, this task required 4% more interactions, on average, in the iOS platform than those in the Android platform.

- (5) *Task 5: delete a chat.* After analyzing the results, this task is more agile to complete in Android apps, requiring 24.2% more interactions in the iOS platform than those in the Android platform. The apps needed 4 to 7 interactions in the iOS platform and between 3 and 7 interactions in Android to complete this task. This difference in the number of interactions is statistically significant ( $p = 0.0002$ ,  $W = 1354.5$ ) depending on how it is implemented. If the chat is deleted by selecting the element-to-delete (i.e., sliding in iOS or long press in Android), then less interactions are required. On the other hand, more interactions are required to delete the chat if the user has to press a top button to enable the selection of the elements to be deleted.

It should be highlighted that, during this step, several functionalities in different apps returned unrecoverable errors that resulted in forced closing (or freezing) of the app (see Recommendation #4). Other apps seemed to be developed for a screen size different to the one that was used for the study (and no information is shown in the store by the vendors about it), resulting on an interface that did not fit well to the screen.

Based on the methodology carried out, not all applications moved to the next step: only the four lowest values of keystrokes representing the top-ranked applications were used in the heuristic evaluation (step 5-B). It should be highlighted that if two or more apps returned the same total number of keystrokes, they were all taken as one to the next step.

Based on the results, on the iOS platform, the following seven apps were chosen: "surespot encrypted messenger" (21 interactions), "hike messenger" (24 interactions), "Hush-HushApp" (24 interactions), "Kik Messenger" (26 interactions), "Touch" (26 interactions), "Hiapp Messenger" (26 interactions), and "WhatsApp Messenger" (27 interactions).

Hence, the following six apps were selected on the Android platform: "surespot encrypted messenger" (19 interactions), "ZOHIB messenger" (23 interactions),

"Yak Messenger" (24 interactions), "Cnectd Messenger - Chat & Text" (24 interactions), "Kik Messenger" (25 interactions), and "HushHushApp" (25 interactions).

**3.6. Step 5-B: Mobile Usability Heuristics.** At this point, the final step (in the form of a heuristic evaluation) was performed on the selected apps. Six experts in mobile technologies evaluated all apps using a set of usability guidelines by applying scores on a 5-point Likert scale (Table 2) [39, 54]. To do this, six experts evaluated the iOS apps and five experts evaluated the Android apps (see Table 3 for further details). All of them made the evaluation individually in a stand-alone environment. All the experts were aged between 18 and 24 years with, at least, a bachelor's degree and more than three years of background experience with mobile devices and mobile apps.

This analysis consisted of checking whether the app met (or not) a set of guidelines. Also, apart from reporting flaws with the scale shown in Table 2, experts were required to provide a justification (i.e., the usability issue) for the score given to each guideline. Thus, the evaluation was rated by considering eight heuristics and their corresponding subsections (namely, subheuristics) (see Table 4) [29].

Based on the experimental evaluation, the results of the heuristic evaluation are shown in Table 5 (iOS platform) and Table 6 (Android platform). Results of each heuristic for a given app (i.e., each cell in the table) were calculated as the average of all expert ratings for all subheuristics. The MEAN<sub>H</sub> column shows the average value for each heuristic.

With the results in hand, it could be seen that heuristics with lower values (i.e., more usable) had, mostly, no problems or cosmetic problems. On the other hand, heuristics with higher values (i.e., less usable) presented issues in the form of minor and major problems. From these results, it could be observed that such problems (minor and major) affect the functionality of the application in a regular use. However, the other issues (cosmetic problems) mean small obstacles in the interface, which does not affect, at all, the regular use of the application.

At this point, the results of each heuristic could be analyzed further in detail:

(i) *Heuristic A: visibility of system status and losability/findability of the mobile device.* The main issue of this section was, in words of the experts, problems related to pop-ups and panels hiding (fully or partially) the top status bar that were found in some apps (Figure 7). This was found uncomfortable by experts, considering that the user loses the context of the real world and the current status of the device (e.g., battery percentage or network conditions) (see Recommendation #5).

Furthermore, experts detected that some apps did not support the feature of recovering the user's account when the access is done from another device, or when the same device has been formatted

TABLE 2: The 5-point Likert scale used within the heuristic evaluation.

Rating	Description
0	I do not agree that this is a usability problem at all Cosmetic problem only. Need not be fixed unless extra time is available on project
1	Minor usability problem. Fixing this should be given low priority
2	Major usability problem. Important to fix, so should be given high priority
3	Usability catastrophe. Imperative to fix this before product can be released

(see Recommendation #6). Thus, this forces the users to register again in the system with a different ID, which implies losing all contacts, configuration settings, and conversations.

- (ii) *Heuristic B: match between system and the real world.* Our experts found problems to adapt and limit the information on the screen. These issues were reported on both platforms (Figure 8). One of the main limitations of the mobile devices is the reduced display size available to show contents, which could vary from one device (or OS) to another. Thus, the interface should be carefully planned to adapt contents to the available space and set boundaries to the amount of information that will be placed (see Recommendation #7).
- (iii) *Heuristic C: consistency and mapping.* Some apps presented some panels and lists (like dropdown menus) with a number of items that could not possibly fit the screen (Figure 9) or with options that were initially not displayed (Figure 10). In addition, layout designs (as well as UI elements) should be made in line with the target OS, which reduces the learning curve of the app (see Recommendation #7), distribution of OK/Cancel buttons in dialogs (reported by Nielsen [55] as a problem), or the confusion made with the usage of nonstandard UI elements [56]. Besides, experts also highlighted problems with respect to clarity and organization of the displayed information (such as lack of consistency or too much information on the screen) on the iOS platform.
- (iv) *Heuristic D: good ergonomics and minimalist design.* Experts emphasized that more than eight interactions affect the ability to follow the flow of task actions, which is directly related to the learnability skill (see Recommendation #1). Likewise, when the app is used in a different language from the one in which it was created, translations were sometimes clipped. Experts pointed out that this way it is difficult to follow flow of events. In some cases, main tasks could be achieved with more or less obstacles, but when trying to change the app settings is where good translations make the difference (see Recommendation #8).

(v) *Heuristic E: ease of input, screen readability, and glancability.* Experts found several issues while operating with chats (Task 1, Task 2, and Task 5). Mainly, experts observed apps that placed individual and group chats in the same frame, without providing visual clues to distinguish them. Whereas individual chats are intended to be a one-to-one conversation, group chats represent one-to-many conversations. Consequently, experts did not talk about the chat-window itself, but how the chats are shown in the list of active chats. Our recommendation could be twofold: either placing individual and group chats in different sections or displaying visual elements on chat entries that would help on differentiating them (see Recommendation #9).

- (vi) *Heuristic F: flexibility, efficiency of use, and personalization.* While performing Task 1, when starting a new chat, the experts found that showing automatically the keyboard greatly reduced the number of interactions. In addition, while performing Task 3, experts agreed that specifying the ID of the new user (for instance, username, e-mail, or phone number) should be enough to register a new contact, whereas other contact parameters should be optional. This would make adding a new contact a more effective task (see Recommendation #1). Moreover, while operating with chats (Task 1, Task 2, and Task 5), experts tried to turn the device horizontally to facilitate interaction with the app, but they discovered that not all apps were prepared; that is, the user interface was not adapted when the device was rotated (Figure 11) (see Recommendation #10).
- (vii) *Heuristic G: aesthetic, privacy, and social conventions.* In essence, experts found out that the design of mobile apps should not only focus on the available features (and, if any, a minimal UI design) but also consider how the UI will look (in terms of how the layout is reorganized) on different screen sizes and device rotation positions (i.e., vertical and horizontal positions) (see Recommendation #10). Therefore, when it comes to security and privacy aspects, several apps showed a lack of information about these terms that, for experts, was a serious problem because messages could be sent using not reliable channels and there was not enough information about what the system does with the user information (see Recommendation #11).
- (viii) *Heuristic H: realistic error management.* It is interesting to address that some unrecoverable errors that produced a forced closing of the app were also found. Along with this, experts found this issue very disappointing, causing frustration feelings when an unexpected forced close of the app lost some information and requires reopening the app (see Recommendation #4).

TABLE 3: Individual experience of each expert.

iOS	Experience	Age	Gender	Android	Experience	Age	Gender
Expert #1	3 years (ICT company)	24	M	Expert #1	4 years (ICT company)	24	M
Expert #2	3 years (app developer)	22	M	Expert #2	3 years (IT consulting)	23	F
Expert #3	3 years (ICT company)	23	F	Expert #3	3 years (app developer)	21	F
Expert #4	3 years (app developer)	21	M	Expert #4	4 years (ICT company)	24	M
Expert #5	4 years (ICT consulting)	24	M	Expert #5	3 years (app developer)	23	M
Expert #6	4 years (ICT company)	24	F				

TABLE 4: Heuristic subsections.

A	Visibility of system status and losability/findability of the mobile device	A1	The battery status is visible
		A2	The network status is visible when transmitting data
		A3	The time is visible when entering data
		A4	The previously logged data and personal settings can be recovered if the device is lost
B	Match between system and the real world	B1	The information appears in a natural and logical order
		B2	The information is presented clearly
		B3	You can see where everything is that you might need
C	Consistency and mapping	C1	It is easy to see how to do tasks like sending a message, adding a contact or delete a contact
		C2	There are no objects on the interface that you would not expect to see
D	Good ergonomics and minimalist design	D1	The screens are well designed and clear
		D2	The dialogues do not contain information that is irrelevant or rarely used
E	Ease of input, screen readability, and glancability	E1	It is easy to input the numbers
		E2	It is easy to see what the information on each screen means
		E3	You can easily navigate around the app
		E4	The screens have a “back” button
		E5	The user can get crucial information “at a glance”
F	Flexibility, efficiency of use, and personalization	F1	The user can personalize the system sufficiently
		F2	The system allows efficient input of data
G	Aesthetic, privacy, and social conventions	G1	The design looks good
		G2	There are suitable provisions for security and privacy (e.g., transmission of data is encrypted.)
		H1	Users can recover from errors easily (if something goes wrong you can get back to where you were easily.)
H	Realistic error management	H2	If data is input incorrectly, it can be edited
		H3	There is an undo button, where appropriate

TABLE 5: Results of the heuristic evaluation for the iOS platform apps.

Heuristic\app	Hiapp messenger	Hike messenger	HushHushApp	Kik messenger	Surespot	Touch	WhatsApp messenger	MEAN <sub>H</sub>	Std
A	0.00	1.00	0.92	2.63	2.54	0.00	0.08	1.02	1.14
B	1.17	2.39	0.61	1.61	2.17	2.00	0.28	1.46	0.80
C	1.67	1.75	0.67	2.08	1.75	1.67	0.92	1.50	0.51
D	1.42	2.08	1.00	1.17	0.58	1.92	0.67	1.26	0.58
E	0.70	1.57	0.23	1.33	1.03	1.07	0.13	0.87	0.54
F	0.50	0.67	1.17	1.25	1.50	2.08	0.42	1.08	0.60
G	1.25	2.33	0.50	1.67	0.50	3.17	1.58	1.57	0.96
H	0.89	1.44	0.39	1.00	0.61	0.89	0.00	0.75	0.46
							Mean	1.19	0.31

As a final recall for all heuristics, results (Figure 12) are significantly better (i.e., lower values) on the Android platform than on the iOS platform, except when it is referred to the interface design and completion of tasks. On average, heuristics on the Android platform is around 0.87 (it does

not get to be categorized as a cosmetic problem), while this value rises to 1.19 on the iOS platform (categorized between a cosmetic problem and a minor problem).

It is noteworthy that apps that performed well in the KLM analysis (low number of interactions) underperformed in the

TABLE 6: Results of the heuristic evaluation for the Android platform apps.

Heuristic\app	Cnectd	Hush hush app	Kik messenger	Surespot	Yak	Zohib	Mean <sub>h</sub>	Std
A	0.05	0.00	0.00	0.60	1.00	0.00	0.28	0.43
B	0.60	0.40	1.47	1.27	1.27	0.53	0.92	0.46
C	1.50	0.50	2.70	1.30	1.40	1.50	1.48	0.71
D	0.60	0.10	2.10	0.90	0.20	0.30	0.70	0.75
E	0.44	0.28	0.96	0.96	0.56	0.72	0.65	0.28
F	0.80	0.00	0.50	1.90	0.00	1.00	0.70	0.72
G	3.30	0.10	2.10	1.20	2.30	2.30	1.88	1.10
H	0.27	0.33	0.00	0.00	0.33	1.20	0.36	0.44
						<b>Mean</b>	0.87	0.55



FIGURE 7: The top status bar is hidden when a left panel is shown.

heuristic evaluation (a larger number of usability problems were found). To our best knowledge, it could be suggested using both methods (KLM and heuristic evaluation) for an optimal usability evaluation of mobile applications.

#### 4. Recommendations

At this point, in this section, our findings could be summarized in the following set of recommendations:

- (i) **Recommendation #1: main features should be easy to access.** This recommendation results in making simple and effective functionalities with a fewer number of interactions. Especially, the main functions of chatting and adding/deleting contacts should be highly intuitive, or at least clear instructions on how to perform these tasks should be provided. Experts determine that this would make

tasks easier and faster to complete. Thus, and according to Miller's report [53],  $7 \pm 2$  interactions should be optimum for main IM features.

- (ii) **Recommendation #2: automatic display of the keyboard at new chats.** When sending a message that initiates a new chat with a contact, display automatically the keyboard. Experts rated this positively. They pointed out that, in a new chat, it is inherent to display the keyboard, so it is better to show it automatically.
- (iii) **Recommendation #3: add a new contact only with the ID.** When adding a new contact, to make it an agile task, specifying the ID (username, phone number, e-mail, etc.) of the new contact should be enough information to register the contact. All other (extra) information (e.g., contact details) should be optional and it should be possible to add



FIGURE 8: Examples of bad adaption of the information to the screen.

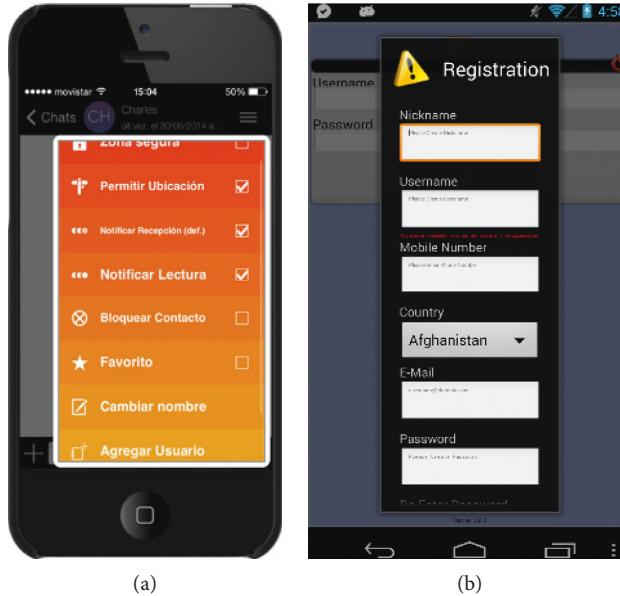


FIGURE 9: Panels filled with options. (a) Dropdown menu with too much options that requires sliding and (b) a pop-up with too much text inputs for a regular screen that requires sliding and reduces the font size to the minimum.

it later, if needed. Experts determined this to make tasks easier and faster to complete.

- (iv) ***Recommendation #4: do not tolerate unrecoverable errors.*** Unrecoverable errors should be avoided as far as possible. This is an obvious recommendation, but such problems were found in some of the analyzed apps. It is highly preferable to close a window (or a pop-up) with an error report than

an unexpected shutdown or a freezing of the app; thus, “help users recognize, diagnose, and recover from errors” [37]. Likewise, as detailed by Nielsen [57], error messages should be explicit, human-readable, polite, precise, constructive, and highly noticeable. The application should also be tested on different end-target devices/OS to avoid anomalies derived from selling or



FIGURE 10: Initially (a) no options are shown, but (b) hidden content appears after sliding the screen.



FIGURE 11: Example of user interface before (a) and after (b) the device is rotated.

publishing the app for a different device to that used in the design and implementation of the app (variation in screen size or different versions of the operating system, among others). Experts suggested this recommendation after finding unrecoverable errors in their analysis, which were highly negative.

- (v) **Recommendation #5: keep the top status bar always visible.** While using the app, whenever possible, avoid panels or pop-ups that overlap totally or partially the top status bar (battery,

time/date, and network indicators). The exception is when an app is specifically designed to be used always in full-screen mode.

- (vi) **Recommendation #6: provide account recovery features.** Whenever possible, the app should implement methods that allow the user to restore the account (in the same or a different device) if the device is lost or formatted. At least the account details (profile) and contacts should be recovered. Retrieving messages may not be necessary, since

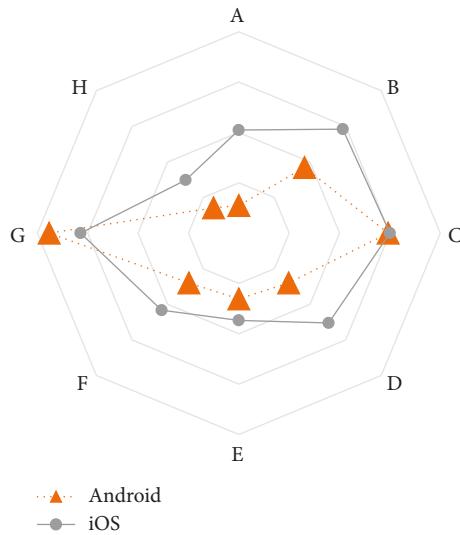


FIGURE 12: Results of heuristic evaluation on Android and iOS platforms, showing a comparison of the mean values of each heuristic.

servers, for privacy reasons, usually do not store messages for long periods of time. Experts realized that not all the apps have implemented this feature, so users who change from one device to another could not be able to retrieve their account details.

(vii) ***Recommendation #7: UI adapted to and limited by the operating system (OS).*** Content displayed should be adapted and limited to the screen size and the OS. The available space on the screen is not very large, so clipped elements are quite common when translations are applied, making content unreadable. As seen in [58], hyperlinks showed in texts should be as short as possible, remarking this need of content adapted to the available space. In addition, as seen in [59], each OS presents its own UI design alternatives. To improve the user experience and to keep similarity to the host system, icons, buttons, and other elements should be in line with the operating system.

(viii) ***Recommendation #8: avoid half translations.*** For those apps aimed at a target audience with a different language, half translations should be avoided because some users do not understand other languages. This recommendation, along with consistency in the design of the interface, is important so that texts do not appear as trimmed or misplaced; or even suffering from UI changes [59]. Experts pointed out that internationalization of the interface texts is highly related to the consistency of the app, also important due to the globally dissemination of this kind of apps.

(ix) ***Recommendation #9: provide visual distinction between individual and group chats.*** Individual and group chats should be differentiated, for example, using multiple tabs or icons. Experts pointed out that group and individual chats are slightly

different, in terms of the number of people involved and the content of such chats. Therefore, a distinction should be made to speed up visualization of information.

(x) ***Recommendation #10: design the interface carefully and accurately.*** The development of an app should be focused on the user interface in order to ensure that the content is appropriately displayed in different screen sizes, screen resolutions, and orientations of the device (portrait and landscape). The app should show the same minimum content, regardless of screen rotation or size, to ensure a satisfactory user experience. Extra content may vary for devices with larger screens or if the device is placed horizontally (some apps change text for icons when rotating the device because there is less space to be used). This has been reasoned as a way to force an improvement of the interface, in which most of the apps have been discovered by the experts as a main failure. In addition, enable using the device in landscape orientation, which is especially important while writing messages (in fact, in all keyboard interactions). Experts pointed out that using the device horizontally greatly improves the ease of use of the app. Also, note that not all analyzed apps facilitate this action because several did not adapt the interface when the device was rotated.

(xi) ***Recommendation #11: provide security mechanisms and information to the user.*** Since messages usually travel through unsecure channels in this type of apps [60, 61], the application should provide mechanisms to ensure the encryption of information. It should be noted that, in several number of available mobile apps, there are reported scarce security methods implemented and a lack of information to the user [62]. Privacy policies should be clearly stated so that the user knows how private information is managed and displayed. Experts pointed out that there was few or none information about this subject, which has been measured as a very important feature for this kind of apps.

## 5. Discussion

Generally speaking, this paper presented a research that aims to broaden our knowledge of the usability issues of mobile instant messaging apps. The methodology used can evaluate mobile apps in terms of efficiency, which is a common metric to measure usability [63] and usability inspection, aiming to detect potential flaws in the apps. Nonetheless, it could be highlighted as a limitation that satisfaction, one of the main (emotional) metrics of usability, was not measured here, given that real users were not surveyed in this study.

In particular, the methodical evaluation [39] carried out here to detect usability issues was previously tested successfully on two types of apps: diabetes management apps

[42, 43] and spreadsheet apps [41]. Additionally, these authors applied this methodical evaluation on previous studies in IM environments [64, 65]. The results of these studies (as well as this study about IM apps) show that it is necessary to apply both KLM and heuristic evaluation methods in order to detect more usability problems than when only one of them is used. The KLM is an objective method (e.g., error management dialogs adds, irrevocably, interaction costs compared with those apps that do not apply error management), whereas the heuristic evaluation is a subjective method. When both methods are applied, more reliable results could be obtained.

Besides the limitation of the context, which is addressed by the methodology, other limitation of these kind of studies (i.e., methodical evaluations) could be, for example, that several steps turned into exhausting stages when there are lots of elements (mobile apps, in this context) to filter out. Other issues are, for example, time-sensitive results or the fact that not all usability issues can be detected [66]. Moreover, only iOS and Android platforms were evaluated in this study. Although they are the most used mobile platforms [67], other platforms may return different results, due to their software and hardware capabilities.

Although it is not clear in which degree a usability expert could provide better heuristic evaluation results than a context-related professional (as the ones selected in this study) [68–70], authors would like to stress this as a possible limitation of the study, given that experts chosen for this evaluation were experts in mobile technologies, rather than in heuristic evaluations.

It is interesting to compare this paper with three mobile IM prototype apps that were developed by other authors: Perttunen et al. for a PDA (in 2005) [71], by Inbar and Zilberman for several platforms (in 2008) [72] and by Nawi et al. for Android (in 2012) [73]. At the first glance, as in our study, their usability evaluations came out with problems associated to the Nielsen's "Visibility of System Status" web-heuristic, in some aspects quite similar to heuristic A of our applied methodology. In our study, the problems detected were related to the status bar (it was hidden in some applications while performing different actions on the UI) (as reflected in our Recommendation #5). Perttunen et al.'s issues pointed to the need of visualizing indicators related to availability (user status) and message transmission (delivery information), but this study was only focused on sending messages under diverse user status conditions. Inbar and Zulberman's study reported usability issues before the IM app was created. They were related to user status and fast ways to initiate chats. The small number of issues found may be because the usability analysis was made with case studies examples. As well as in our study, in the 2012 prototype (Nawi et al.'s), the evaluation also found problems with the UI, although the authors suggested that this was because the application was just a prototype, not a final version.

Although the previous studies applied usability evaluations to mobile IM apps, the most similar study to this paper was an evaluation made in 2013: a MIM usability evaluation on Android devices [74] using cognitive

walkthrough in a laboratory environment with the assistance of six evaluators. Due to, in the authors' opinion, the impossibility to target all existing IM apps on markets, they used a review on the best Android platform applications (based on users' opinions) published by the "PC Magazine" website [75]. Authors chose the best three apps of the review on the "Communication" category, that is, "WhatsApp," "Skype," and "GO SMS Pro." The usability evaluation was then performed by creating and analyzing tasks (like the methodology applied in this study): chatting, file transfer, contact features (addition, update, and visualization), and profile status. The results of this evaluation are aligned with our study, since they conclude that there were similar buttons and icons (in some cases they were the same visual element) with different actions and, as previously said in this paper, this issue leads user to confusion (as shown in our Recommendation #7 and, also, reported by Nielsen when a GUI element looks like a clickable element but it does not initiate any action [56]). However, the study also found usability issues of these apps like, for example, users could not select multiple emoticons at once; there was no confirmation message when the user sends a file or an inefficient "Search" feature, among other problems. Nevertheless, these problems were not found in our study, may be due to the fact that tasks were defined taking into account features offered by the applications (related mainly to the chats section), whereas in our study, tasks were defined after a literature review and trying to identify what is an IM application based on its main and necessary tasks. Their decision has the clear disadvantage of not covering all possible dimensions and functionalities of IM apps. It is true that, if our evaluation would have included other or more tasks, more diverse problems could have been discovered.

Meanwhile, in Mendoza's book [76], a series of advices are given as mobile design patterns based on the author's experience, since (desktop) web UX is not the linear mobile UX. In general terms, mobile experiences should keep in mind some performance scenarios: short, easy, OS-centered, and consistent navigation, as well as small and clean layouts with larger and concise UI elements. Extra use of options, images, and texts slow down the experience. Screen-rotation adaptation does not only mean adjusting width and height dimensions of the contents but also produces experiences according to the new orientation. Mobile experiences are expected to be fast, while loading times and many screens could be seen as unsuccessful experiences. All of this is, as well, aligned with our Recommendation #10.

In words of Swierenga [77], although there are guidelines in iOS and Android for consistent UI designs, there is a lack of usability guidelines for mobile apps. Hence, these authors proposed a set of usability guidelines to be used in outdoor tourism apps, mainly, providing high quantities of information to the user. Essentially, their first set of recommendations is thought to be for broad kind of apps: navigation elements at the bottom of the interface, scrolling as least as possible, oversized and simple clickable elements, well-defined titles for all UI elements (aligned with our Recommendation #10), and fewer number of interactions to

get desired contents (fully aligned with our Recommendation #1). Finally, they provided a set of recommendations specially created for outdoor environments, such as, download contents to the device, increase readability with bigger font size and contrast, and produce versions for both smartphones and tablets, given the different screen sizes among these kind of devices (aligned with our Recommendations #7 and #10).

In addition to it, Shitkova [24] came out with the same point: there is a lack of usability guidelines for mobile apps in the scientific literature. To cope with it, the authors created a list of 39 usability guidelines for mobile websites and applications. The guideline is divided into five sections: layout, navigation, design, content, and performance. Some of their guidelines could be arranged with our usability recommendations: easy navigation elements with few options, number of clicks as few as possible (both aligned with our Recommendation #1), self-explanatory titles for all UI elements, optimized UI, contents, and functionalities similar in different device versions of the app (all of them aligned with our Recommendation #7), easy-to-understand UI elements, avoid table layouts, contents ordered by importance, confirmation dialogues, consistent, uniform, and simple UI (all of them aligned with our Recommendation #10).

Nonetheless, other authors [78] pointed out the convenience of creating generic usability guidelines, instead of app-specific guidelines. However, authors admitted a two-fold consequence: generic guidelines could not be, may be, used in specific kinds of apps; and, also, specific guidelines could not be generalized to broader domains. Authors present a compilation of the main usability guidelines present in the scientific literature. It was expected that our most specific recommendations for IM apps, that is, Recommendations #2, #3, and #9 were, indeed, not present in that collection. However, we presented other generic-like recommendations which are not present, either, in that collection of guidelines. Although a bit specific, but certainly for a huge number of apps, our Recommendation #6 (account recovery) was not shown there. Similarly, our generic Recommendation #8 (avoid half translations) was not specifically covered. It is partially covered, with the “Do not use objects with different meanings” and “use terms related to the real world” guidelines, but not explicitly controlled. Finally, our generic Recommendation #11 (security mechanisms and information) was not presented, in any way, in any of the usability guidelines.

To conclude this section, these advices, our recommendations (see previous section), and other kinds of guidance found on the literature demonstrate what this paper has been remarking since the beginning: app designers are not applying these methods, methods that only want to improve user’s mobile experiences, and it is noticeable for the flaws in the apps. Also, reducing learnability curves in the usage of these apps reflects in a benefit for the users.

## 6. Conclusions

With the high widespread adoption of smartphones and the fast proliferation of instant messaging applications (IM), the

usability evaluation of this type of mobile applications is required. Usability is in the form of a User eXperience (UX) attribute. In a different way from PC contexts, addressing the usability of mobile apps require taking into account multiple factors: small screen size, limited capabilities, network limitations, or input restrictions, among other factors.

To summarize, the presented paper has followed a methodical evaluation to detect the main usability issues of instant messaging apps both on Android and iOS platforms (an iPhone and a Google Nexus were used for this study). This sequential approach followed five steps: (1) identification of potentially relevant apps, (2) discard demos and old versions of apps, (3) identification of main functionalities and exclusion of apps not offering all of these functionalities, (4) identification of secondary functionalities, and (5a) keystroke level modelling (KLM) to measure time to complete the main functionalities and (5b) heuristic evaluation (HE) to detect usability problems along with 6 experts in mobile technologies.

Within the first two steps of the evaluation, after reviewing literature, it was concluded that an IM application should have at least the following main features/functionalities: (T1) sending an instant message to a particular contact, (T2) reading and replaying an incoming message, (T3) adding a new contact, (T4) deleting (or blocking) a contact, and (T5) deleting a chat. In Step 3, all apps from online market stores (App Store and Google Play) that did not meet all of these features were excluded and the list of relevant applications was obtained. Hence, only 39 apps met these features on iOS, whereas 106 apps met these features on Android.

In relation to the results of the KLM test, consisted in counting the number of interactions of the main features for each app, returned that the average total number of interactions of an IM application is quite similar on both platforms: 29.86 on iOS and 29.4 on Android.

Regarding the HE, the collaboration of several experts in mobile technologies was required to determine which usability heuristics were met when examining applications. These heuristics are divided into subheuristics, which were evaluated using a 5-point Likert scale. An in-depth analysis of the results suggested that both platforms have significant issues related to a bad interface design, clarity and organization of the displayed information, lack of security/privacy information, and problems to complete the main tasks.

Preliminary validation results of the KLM and HE suggest that both methods should be applied to detect the largest number of usability problems. Indeed, this is because applications with low scores on KLM (theoretically the best applications) had worse results in HE (i.e., more problems were detected in better-rated apps on KLM).

Finally, after the process, this paper came up with a set of usability guidelines:

- (i) Recommendation #1: main features should be easy to access
- (ii) Recommendation #2: automatically display of the keyboard at new chats

- (iii) Recommendation #3: add a new contact only with the ID
- (iv) Recommendation #4: do not tolerate unrecoverable errors
- (v) Recommendation #5: keep the top status bar always visible
- (vi) Recommendation #6: provide account recovery features
- (vii) Recommendation #7: UI adapted to and limited by the OS
- (viii) Recommendation #8: avoid half translations
- (ix) Recommendation #9: provide visual distinction between individual and group chats
- (x) Recommendation #10: design the interface carefully and accurately
- (xi) Recommendation #11: provide security mechanisms and information to the user

All these recommendations should be considered for the benefit of the users, since an improved usability of the application involves a higher number of downloads and active users. This, depending on the business model of the app, could also lead to a higher number of economic benefits. Based on the presented findings, these recommendations call attention to the usability of mobile applications, both to interface designers and to companies to focus their development efforts on user-based and user-experience designs. As most of the technology of the twentieth century, the aim is to make tech world closer and easier for the user.

Finally, as a future work, our plan is to create a prototype following the previous recommendations and perform the same methodical analysis in order to compare the results with the previous ones, as well as carrying out an experiment with real users, which will allow us to validate the guidelines proposed. In addition to this, evaluating UX aspects over this type of apps could answer several questions, like how, where, and why users use certain IM apps over other, may be usability friendly ones, alternatives.

## Data Availability

The source data used to support the findings of this study are included within the article. Additionally, the source data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The authors would like to thank the support of TIFYC and PMI research groups. This research was funded by the FPU research staff education program of the “University of Alcala.”

## References

- [1] K. Mullan and J. Wajcman, “Have mobile devices changed working patterns in the 21st century? A time-diary analysis of work extension in the UK,” *Work, Employment and Society*, 2017.
- [2] S. Ozawa and E. T. Ualesi, “A pilot study of mobile learning in higher education in Samoa,” in *Mobile Learning in Higher Education in the Asia-Pacific Region: Harnessing Trends and Challenging Orthodoxies*, A. Murphy, H. Farley, L. Evelyn Dyson, and H. Jones, Eds., Springer Singapore, Singapore, pp. 557–579, 2017.
- [3] P. Farago, *iOS and Android Adoption Explodes Internationally, The Flurry Blog*, 2013.
- [4] statista, “Smartphones worldwide installed base from 2008 to 2017 (in millions),” 2017, <https://www.statista.com/statistics/371889/smartphone-worldwide-installed-base/>.
- [5] F. Cuadrado and J. C. Dueñas, “Mobile application stores: success factors, existing approaches, and future developments,” *IEEE Communications Magazine*, vol. 50, no. 11, pp. 160–167, 2012.
- [6] B. Womack, “Google says 700,000 applications available for android,” October 2012, <https://www.bloomberg.com/news/articles/2012-10-29/google-says-700-000-applications-available-for-android-devices>.
- [7] statista, “Number of apps available in leading app stores as of July 2015,” 2015, <http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>.
- [8] Apple-Inc, “App Store shatters records on New Year’s Day,” January 2017, <https://www.apple.com/newsroom/2017/01/app-store-shatters-records-on-new-years-day/>.
- [9] statista, “Number of available applications in the Google Play Store from December 2009 to December 2017,” 2018, <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>.
- [10] N. Tillmann, M. Moskal, J. de Halleux, M. Fahndrich, and S. Burckhardt, “TouchDevelop: app development on mobile devices,” in *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering*, November 2012.
- [11] K. Church and R. de Oliveira, “What’s up with whatsapp?: comparing mobile instant messaging behaviors with traditional SMS,” in *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services*, August 2013.
- [12] S. H. Hsieh and T. H. Tseng, “Playfulness in mobile instant messaging: examining the influence of emoticons and text messaging on social interaction,” *Computers in Human Behavior*, vol. 69, pp. 405–414, 2017.
- [13] Y. Sun, D. Liu, S. Chen, X. Wu, X.-L. Shen, and X. Zhang, “Understanding users’ switching behavior of mobile instant messaging applications: an empirical study from the perspective of push-pull-mooring framework,” *Computers in Human Behavior*, vol. 75, pp. 727–738, 2017.
- [14] C. Clifford, “Top 10 apps for instant messaging,” *Entrepreneur (Technology)*, p. 1, 2013 December 11, 2013.
- [15] statista, “Number of mobile phone messaging app users worldwide from 2016 to 2021 (in billions),” 2018, <https://www.statista.com/statistics/483255/number-of-mobile-messaging-users-worldwide/>.
- [16] A. Dix, *Human-Computer Interaction, Encyclopedia of Database Systems*, Springer, Berlin, Germany, 2009.
- [17] I. Dis, *Ergonomics of Human System Interaction-Part 210: Human-Centred Design for Interactive Systems*, International

- Standardization Organization (ISO), , Geneva, Switzerland, 9241-210, 2010, 2009.
- [18] E. L-C. Law, V. Roto, M. Hassenzahl, A. P. O. S. Vermeeren, and J. Kort, "Understanding, scoping and defining user experience: a survey approach," in *Proceedings of the SIGCHI conference on human factors in computing systems*, April 2009.
- [19] D. Norman and J. Nielsen, "The definition of user experience (UX)," June, 2018, <https://www.nngroup.com/articles/definition-user-experience/>.
- [20] E. Din, *Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)-Part 11: Guidance on usability*, International Organization for Standardization, 9241-11, 1998.
- [21] J. Nielsen, *Usability 101: Introduction to Usability*, 2003.
- [22] L. Rampersad, S. Blyth, E. Elson, and M. M. Kuttel, "Improving the usability of scientific software with participatory design: a new interface design for radio astronomy visualisation software," in *Proceedings of the South African Institute of Computer Scientists and Information Technologists*, pp. 1-9, ACM, Thaba'Nchu, South Africa, September 2017.
- [23] J. Sauro and P. Zarolia, "SUPR-Qm: a questionnaire to measure the mobile app user experience," *Journal of Usability Studies*, vol. 13, no. 1, pp. 17-37, 2017.
- [24] M. Shitkova, J. Holler, T. Heide, N. Clever, and J. Becker, "Towards usability guidelines for mobile websites and applications," in *Proceedings of Wirtschaftsinformatik*, Osnabrück, Germany, March 2015.
- [25] P. Brauner, A. Calero Valdez, R. Philipsen, and M. Ziefle, "On studying human factors in complex cyber-physical systems," in *Proceedings of Mensch und Computer 2016–Workshopband*, Aachen, Germany, September 2016.
- [26] K. Vredenburg, J.-Y. Mao, P. W. Smith, and T. Carey, "A survey of user-centered design practice," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, April 2002.
- [27] N. Jacobsen, *Usability Evaluation Methods*, The Reliability and Usage of Cognitive Walkthrough and Usability Test, 1999.
- [28] J. Lazar, J. H. Feng, and H. Hochheiser, *Research Methods in Human-Computer Interaction*, Wiley Publishing, Hoboken, NJ, USA, 2010.
- [29] E. Bertini, T. Catarci, A. Dix, S. Gabrielli, S. Kimani, and G. Santucci, "Appropriating heuristic evaluation for mobile computing," *International Journal of Mobile Human Computer Interaction (IJMHCI)*, vol. 1, no. 1, pp. 20-41, 2009.
- [30] H. Hong and D. Xu, "An empirical study of mobile social app continuance intention: integrating flow experience and switching costs," *International Journal of Networking and Virtual Organisations*, vol. 17, no. 4, pp. 410-424, 2017.
- [31] K. Moumane, A. Idri, and A. Abran, "Usability evaluation of mobile applications using ISO 9241 and ISO 25062 standards," *SpringerPlus*, vol. 5, no. 1, p. 548, 2016.
- [32] M. Matera, M. F. Costabile, F. Garzotto, and P. Paolini, "SUE inspection: an effective method for systematic usability evaluation of hypermedia," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 32, no. 1, pp. 93-103, 2002.
- [33] A. de Lima Salgado and A. P. Freire, *Heuristic Evaluation of Mobile Usability: A Mapping Study*, *Human-Computer Interaction. Applications and Services*, Springer, Berlin, Germany, 2014.
- [34] G. Joyce and M. Lilley, *Towards the Development of Usability Heuristics for Native Smartphone Mobile Applications, Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience*, Springer, Berlin, Germany, 2014.
- [35] E. Garcia-Lopez, L. de-Marcos, A. Garcia-Cabot, and J.-J. Martinez-Herraiz, "Comparing zooming methods in mobile devices: effectiveness, efficiency, and user satisfaction in touch and nontouch smartphones," *International Journal of Human-Computer Interaction*, vol. 31, no. 11, pp. 777-789, 2015.
- [36] E. Garcia-Lopez, A. Garcia-Cabot, and L. de-Marcos, "An experiment with content distribution methods in touchscreen mobile devices," *Applied ergonomics*, vol. 50, pp. 79-86, 2015.
- [37] R. Inostroza, C. Rusu, S. Roncagliolo, V. Rusu, and C. A. Collazos, "Developing SMASH: a set of SMArtphone's usability heuristics," *Computer Standards & Interfaces*, vol. 43, pp. 40-52, 2016.
- [38] D. Quinones, C. Rusu, S. Roncagliolo, V. Rusu, and C. A. Collazos, "Developing usability heuristics: a formal or informal process?," *IEEE Latin America Transactions*, vol. 14, no. 7, pp. 3400-3409, 2016.
- [39] C. Martin, D. Flood, and R. Harrison, "A protocol for evaluating mobile applications," in *Proceedings of the IADIS International Conference on Interfaces and Human Computer Interaction*, July 2011.
- [40] A. Dhouib, A. Trabelsi, C. Kolski, and M. Neji, "A classification and comparison of usability evaluation methods for interactive adaptive systems," in *Proceedings of 9th International Conference on Human System Interactions (HSI)*, Portsmouth, UK, July 2016.
- [41] D. Flood, "A systematic evaluation of mobile spreadsheet apps," in *Proceedings of IADIS International Conference Interfaces and Human Computer Interaction*, Rome, Italy, July 2011.
- [42] E. Garcia, C. Martin, A. Garcia, R. Harrison, and D. Flood, "Systematic analysis of mobile diabetes management applications on different platforms," in *Proceedings of Information Quality in e-Health*, pp. 379-396, Graz, Austria, November 2011.
- [43] C. Martin, D. Flood, D. Sutton, A. Aldea, R. Harrison, and M. Waite, "A systematic evaluation of mobile applications for diabetes management," in *Proceedings of Human-Computer Interaction—INTERACT*, pp. 466-469, Springer, Lisbon, Portugal, September 2011.
- [44] B. A. Nardi, S. Whittaker, and E. Bradner, "Interaction and outeraction: instant messaging in action," in *Proceedings of 2000 ACM Conference on Computer Supported Cooperative Work*, December 2000.
- [45] A. Tomar and A. Kakkar, "Maturity model for features of social messaging applications," in *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*, Noida, India, October 2014.
- [46] D. Cui, "Beyond "connected presence": multimedia mobile instant messaging in close relationship management," *Mobile Media & Communication*, vol. 4, no. 1, pp. 19-36, 2016.
- [47] R. E. Grinter and L. Palen, "Instant messaging in teen life," in *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, November 2002.
- [48] C. Lewis and B. Fabos, "Instant messaging, literacies, and social identities," *Reading Research Quarterly*, vol. 40, no. 4, pp. 470-501, 2005.
- [49] B. S. Gerber, M. R. Stolley, A. L. Thompson, L. K. Sharp, and M. L. Fitzgibbon, "Mobile phone text messaging to promote healthy behaviors and weight loss maintenance: a feasibility study," *Health informatics journal*, vol. 15, no. 1, pp. 17-25, 2009.

- [50] M. Rost, C. Kitsos, A. Morgan et al., “Forget-me-not: history-less mobile messaging,” in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 1904–1908, ACM, Santa Clara, CA, USA, May 2016.
- [51] D. Kieras, *Using the Keystroke-Level Model to Estimate Execution Times*, University of Michigan, Ann Arbor, MI, USA, 2001.
- [52] A. D. Rice and J. W. Lartigue, “Touch-level model (TLM): evolving KLM-GOMS for touchscreen and mobile devices,” in *Proceedings of the 2014 ACM Southeast Regional Conference*, pp. 1–6, ACM, Kennesaw, Georgia, March 2014.
- [53] G. A. Miller, “The magical number seven, plus or minus two: some limits on our capacity for processing information,” *Psychological review*, vol. 63, no. 2, pp. 81–97, 1956.
- [54] J. Nielsen, “Enhancing the explanatory power of usability heuristics,” in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, April 1994.
- [55] J. Nielsen, “OK-cancel or cancel-OK? The trouble with buttons,” May 2008, <https://www.nngroup.com/articles/ok-cancel-or-cancel-ok/>.
- [56] J. Nielsen, “Top 10 application-design mistakes,” February 2008, <https://www.nngroup.com/articles/top-10-application-design-mistakes/>.
- [57] J. Nielsen, “Error message guidelines,” June 2001, <https://www.nngroup.com/articles/error-message-guidelines/>.
- [58] E. Garcia-Lopez, A. Garcia-Cabot, C. Manresa-Yee, L. de-Marcos, and C. Pages-Arevalo, “Validation of navigation guidelines for improving usability in the mobile web,” *Computer Standards & Interfaces*, vol. 52, pp. 51–62, 2017.
- [59] M. Rauch, “Mobile documentation: usability guidelines, and considerations for providing documentation on Kindle, tablets, and smartphones,” in *Proceedings of IEEE International Professional Communication Conference (IPCC)*, Cincinnati, OH, USA, October 2011.
- [60] M. La Polla, F. Martinelli, and D. Sgandurra, “A survey on security for mobile devices,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 446–471, 2013.
- [61] D. Christin, A. Reinhardt, S. S. Kanhere, and M. Hollick, “A survey on privacy in mobile participatory sensing applications,” *Journal of Systems and Software*, vol. 84, no. 11, pp. 1928–1946, 2011.
- [62] S. M. Dye and K. Scarfone, “A standard for developing secure mobile applications,” *Computer Standards & Interfaces*, vol. 36, no. 3, pp. 524–530, 2014.
- [63] R. Baharuddin, D. Singh, and R. Razali, “Usability dimensions for mobile applications—a review,” *Research Journal of Applied Sciences, Engineering and Technology*, vol. 5, pp. 2225–2231, 2013.
- [64] S. Caro-Alvaro, A. Garcia, E. Garcia, L. de-Marcos, and J.-M. Gutierrez-Martinez, “Mobile instant messaging apps: usability evaluation on iOS and android platforms and recommendations for developers,” in *Proceedings of Information Systems Development: Advances in Methods, Tools and Management (ISD2017 Proceedings)*, September, 2017.
- [65] S. Caro-Alvaro, E. Garcia-Lopez, A. Garcia-Cabot, L. de-Marcos, and J.-M. Gutierrez-Martinez, “A systematic evaluation of mobile applications for instant messaging on iOS devices,” *Mobile Information Systems*, vol. 2017, Article ID 1294193, 17 pages, 2017.
- [66] J. Bastien, “Usability testing: a review of some methodological and technical aspects of the method,” *International Journal of Medical Informatics*, vol. 79, no. 4, pp. e18–e23, 2010.
- [67] M. E. Joorabchi, A. Mesbah, and P. Kruchten, “Real challenges in mobile app development,” in *Proceedings of ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, October 2013.
- [68] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, April 1990.
- [69] A. Holzinger, “Usability engineering methods for software developers,” *Communications of the ACM*, vol. 48, no. 1, pp. 71–74, 2005.
- [70] H. M. Salman, W. F. W. Ahmad, and S. Sulaiman, “Usability evaluation of the smartphone user interface in supporting elderly users from experts’ perspective,” *IEEE Access*, vol. 6, pp. 22578–22591, 2018.
- [71] M. Perttunen, J. Riekki, K. Koskinen, and M. Tahti, “Experiments on mobile context-aware instant messaging,” in *Proceedings of the 2005 International Symposium on Collaborative Technologies and Systems*, May 2005.
- [72] O. Inbar and B. Zilberman, “Usability challenges in creating a multi-IM mobile application,” in *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services*, September 2008.
- [73] M. Nawi, N. Haron, and M. Hasan, “Context-aware instant messenger with integrated scheduling planner,” in *Proceedings of the International Conference on Computer & Information Science (ICCIS)*, June 2012.
- [74] D. Jadhav, G. Bhutkar, and V. Mehta, “Usability evaluation of messenger applications for Android phones using cognitive walkthrough,” in *Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction*, September 2013.
- [75] M. Eddy, “Best 100 android apps,” 2012, <http://www.pcmag.com/article2/0,2817,2393097,00.asp>.
- [76] A. Mendoza, *Mobile User Experience: Patterns to Make Sense of it All*, Newnes, Oxford, UK, 2013.
- [77] S. J. Swierenga, D. B. Propst, J. Ismirle, C. Figlan, and C. K. Coursaris, “Mobile design usability guidelines for outdoor recreation and tourism,” Edited by F. F.-H. Nah, Ed., in *Proceedings HCI in Business: First International Conference, HCIB 2014, Held as Part of HCI International 2014*, pp. 371–378, Springer International Publishing, Heraklion, Crete, Greece, June 2014.
- [78] N. Ahmad, A. Rextin, and U. E. Kulsoom, “Perspectives on usability guidelines for smartphone applications: an empirical investigation and systematic literature review,” *Information and Software Technology*, vol. 94, pp. 130–149, 2018.

