

## Developing SMASH: A set of SMARTphone's uSability Heuristics



Rodolfo Inostroza<sup>a</sup>, Cristian Rusu<sup>a,\*</sup>, Silvana Roncagliolo<sup>a</sup>, Virginica Rusu<sup>b</sup>, César A. Collazos<sup>c</sup>

<sup>a</sup> Pontificia Universidad Católica de Valparaíso, Av. Brasil No. 2241, Valparaíso 2340000, Chile

<sup>b</sup> Universidad de Playa Ancha de Ciencias de la Educación, Av. Playa Ancha No. 850, Valparaíso 2340000, Chile

<sup>c</sup> Universidad del Cauca, FIET-Sector Tulcan Calle 5 No. 4–70, Popayán 190003, Colombia

### ARTICLE INFO

#### Article history:

Received 18 April 2013

Received in revised form 10 August 2015

Accepted 18 August 2015

Available online 22 August 2015

#### Keywords:

Usability

Smartphones

Heuristic evaluation

Usability heuristics

Experimental validation

### ABSTRACT

The smartphone market is nowadays highly competitive. When buying a new device, users focus on visual esthetics, ergonomics, performance, and user experience, among others. Assessing usability issues allows improving these aspects. One popular method for detecting usability problems is heuristic evaluation, in which evaluators employ a set of usability heuristics as guide. Using proper heuristics is highly relevant. In this paper we present SMASH, a set of 12 usability heuristics for smartphones and mobile applications, developed iteratively. SMASH (previously named TMD: Usability heuristics for Touchscreen-based Mobile Devices) was experimentally validated. The results support its utility and effectiveness.

© 2015 Elsevier B.V. All rights reserved.

### 1. Introduction

According to Bevan [1], usability is the extent to which a user can easily interact with a product, and it measures how easy to operate, learn and (even) memorize such products are. Many authors consider usability a key requirement for the design of new products, and highly important to people when choosing a product [2,3,4]. In practice, usability goes beyond that. Usability depends on the task, the user's goals and the context of use. It is not necessarily related to ease of use, although the latter is one of its key aspects. In some cases users might be willing to learn how to use a product for months or even years if it supports their activities. In other cases, users do not want to spend more than a couple of minutes learning how to use a tool, and this is the case of mobile applications. The first impression and experience of use is critical and might determine the product's success [5].

Usability evaluation provides developers with insights on the degree to which a product enables a user to achieve specific goals, the efficiency in achieving these goals, the easy to learn and satisfaction [6]. As stated by Lewis [7], there are two major concepts of usability, commonly referred to as summative and formative. While summative usability focuses on the use of metrics associated with meeting global task and product goals (i.e., measurement-based usability), the focus of formative usability is the detection of usability problems and the design of interventions to reduce or eliminate their impact. The latter is also the focus of this work.

Heuristic evaluation is one of the most widely used usability evaluation methods. It normally requires the participation of 3 to 5

evaluators [8]. When selecting the set of heuristics to be used, there are (mainly) two choices: generic or specific heuristics. Specific heuristics may be comparatively difficult to understand and apply, but they can potentially detect more specific usability issues related to the application's domain. Generic heuristics are usually easy to understand and apply, but they can miss specific (domain-related) usability issues [9].

In this paper we present SMASH, a set of 12 SMARTphone's uSability Heuristics that was developed through an iterative process. Our work builds on an early (abbreviated) version of SMASH, previously called TMD (usability heuristics for Touchscreen-based Mobile Devices) [9]. We have performed several additional validation experiments, such as heuristic evaluation and inquiry tests.

This paper is structured as follows: Section 2 introduces main concepts related to smartphones, usability, usability heuristics, and usability evaluation challenges. Section 3 describes the process of developing SMASH; a special emphasis is made on the last iteration, as the earlier ones were presented in previous publications. Section 4 presents the results of the latest experiments. We detail our proposed SMASH usability heuristics set in Section 5. Finally, conclusions and future work are presented in Section 6.

### 2. Theoretical background

#### 2.1. Smartphones

A mobile device is a small electronic appliance, with some processing capabilities, with permanent or intermittent connection to a network [9]. There are several taxonomy proposals [10,11]. Schiefer and

\* Corresponding author. Tel.: +56 32 2274096; fax: +56 32 2273859.  
E-mail address: [cristian.rusu@ucv.cl](mailto:cristian.rusu@ucv.cl) (C. Rusu).

Decker [12] classify mobile devices according to the following criteria: (1) size and weight, (2) input modes, (3) output modes, (4) performance, (5) kind of usage, (6) communication capabilities, (7) type of operating system and (8) expandability. Smartphones are any mobile device that has a touch-sensitive display (touch-screen) and groups a large range of functionalities, including communication capabilities.

## 2.2. Usability

One of the first authors who attempted to define usability was Miller [13]. In the early 1980s the term usability was used as substitute for the terms user friendliness and ease of use [14,15]. Shackel [16] defines usability of a product as “the capability to be used by humans easily and effectively”. Shackel and Richardson [17] define usability as: “the capability in human functional terms for a system to be used easily and effectively by the specified range of users, given specified training and user support, to fulfill the specified range of tasks, within the specified range of scenarios”.

Bevan et al. [14] summarize four different perspectives that have influenced the way usability was defined: (i) a product-oriented perspective emphasizes that usability can be measured with regard to the ergonomic characteristics of a product. (ii) in a user-oriented perspective, usability is represented by means of the user's mental effort in product usage and his/her attitude towards the product. (iii) in a performance-oriented perspective, usability is described in terms of the interaction of the user with the product, while (iv) a context-oriented perspective emphasizes that usability is dependent on the user group that is studied, the tasks those users are performing, and the environment in which the tasks are completed [18].

The requirements proposed by Bevan et al. [14] are satisfied by ISO 9241-210 [19] which defines usability as “extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use”.

The ISO/IEC 25000 series of standards was developed to replace and extend ISO/IEC 9126 and ISO/IEC 14598. The main goal of the ISO/IEC 25000 SQuaRE (Systems and software Quality Requirements and Evaluation) standard is to organize, enhance and unify concepts relevant to two main processes: software quality requirements specification and systems and software quality evaluation, supported by a systems and software quality measurement process [20]. Usability is considered in the whole standard, but it is especially mentioned in ISO/IEC TR 25060:2010 (Common Industry Format (CIF) for Usability – General Framework for Usability-related Information) and ISO/IEC 25062:2006 (Common Industry Format (CIF) for usability test reports). The ISO 20282 is based on ISO 9241-11 [21]. It states that all interactive products will have a user interface, and the quality of the user interface can have significant effects on achieving the user's goal(s) [22]. On the other hand, Nielsen suggests that usability is composed of a set of paradigms, principles and attributes [23].

From above it is apparent that usability is difficult to define. For more than 30 years there has been a lack of a clear and generally accepted definition [24]. The measurement of usability is complex because usability is not a specific property of a person or thing. According to Lewis “you cannot measure usability with a simple ‘usability’ thermometer” [25]. Usability is an emergent property dependent on interactions among users, products, tasks, and environments.

## 2.3. Challenges of usability evaluation in smartphones

When evaluating usability in smartphones it is important to consider the challenging aspects of such products. According to Heo et al. [26],

mobile devices are portable communication and information systems. Their design is influenced by three main aspects:

- They are mainly used on the user's hands.
- They are operated in a wireless way.
- They support the addition of new applications and Internet connection.

Lee et al. [27] introduce additional important aspects to consider:

- They have small screen size to display huge amounts of information at the same time.
- Buttons usually have multiple functionalities.
- The devices have limited processing, power and memory capabilities.

Considering the latest, the trend of mobile multimedia resources is to get “bigger” in size and consume more memory. Some smartphones do not even have external memory slots, making cloud-storage services very appealing.

As we can see, usability evaluation in smartphones can be difficult if the inappropriate tools are used. Reviewing the literature, we did not find a specific usability heuristics set for smartphones.

## 2.4. Related work

Salazar et al. [28] performed a systematic literature review of usability heuristics for mobile phones. The scope of the review mainly considers both feature phones and smartphones, but no specific touch phones heuristics were found beside our proposal [29]. Salgado and Freire [30] present a mapping study of heuristic evaluation of mobile usability, which also considers our work.

Bertini et al. [31] developed a set of 8 usability heuristics for mobile phones. They validate the proposal performing a heuristic evaluation against Nielsen's traditional heuristics [23]. The heuristics consider ergonomic factors along with performance, security and (even) robustness of the device. Some of the heuristics overlap, which may lead to redundancy or mistakes.

Cunha et al. [32] propose a set of specific usability heuristics for modern mobile devices. The set consists of 11 heuristics and is actually an extension of Nielsen's set of usability heuristics [23]. The heuristics “Consistency and standards” and “Help and documentation” have definitions quite different from Nielsen's (traditional) heuristics. Neto and Pimentel [33] use a set of heuristics that appears to be an update of the usability heuristics described in Cunha et al. [32].

Korhonen [34] defines a set of 12 usability heuristics oriented to mobile gaming. Despite the specificity of this set of heuristics, it can also be applied to mobile applications. Ponnada and Kannan [35] make use of the Korhonen's set in their work. They validate the set of heuristics through a heuristic evaluation.

Väänänen-Vainio-Mattila and Wäljas [36] propose a set of mobile web services user experience heuristics. The initial set is composed of 7 heuristics, which were later increased to 9 [37].

Most of the above-mentioned sets of heuristics are based on Nielsen's (traditional) heuristics; changes were made in heuristics' names, definitions and/or order, which may affect evaluators' memorability.

## 3. Method and materials

We developed SMASH (SMARtphone's uSability Heuristics) using a 6-steps methodology proposed by Rusu et al. [38]:

- STEP 1: An exploratory stage, to collect bibliography related to smartphones, their characteristics, as well as general and/or related (if available) usability heuristics.

- STEP 2: A descriptive stage, to highlight the most important characteristics of the previously collected information, in order to formalize the main concepts associated with the research.
- STEP 3: A correlational stage, to identify the characteristics that usability heuristics for smartphones should have, based on traditional heuristics and case studies analysis.
- STEP 4: An explicative stage, to formally specify the set of usability heuristics, using a standard template.
- STEP 5: A validation (experimental) stage, to check the proposed set of usability heuristics against traditional (Nielsen's) heuristics by experiments, through heuristic evaluations performed on selected case studies, complemented by user tests.
- STEP 6: A refinement stage, based on the feedback from the validation stage.

The template used at STEP 4 was the following one:

- ID, Name and Definition: Heuristic's identifier, name and definition.
- Explanation: Heuristic's detailed explanation, including references to usability principles, typical usability problems, and related usability heuristics proposed by other authors.
- Examples: Examples of heuristic's violation and compliance.
- Benefits: Expected usability benefits, when the heuristic is accomplished.
- Problems: Anticipated problems of heuristic misunderstanding, when performing heuristic evaluations.

SMASH was developed through five iterations. In a first iteration (1) a literature review was made and (2) a guided inspection was conducted in order to identify usability issues based on Nielsen's usability heuristics and experts' opinion. For the first experiment, two devices were inspected: (1) Samsung Galaxy S1 running Android and (2) Nokia X6 running Symbian. After identifying usability issues and positive aspects, and mapping them to Nielsen's heuristics, the issues that could not be associated to Nielsen's heuristics were carefully analyzed. A preliminary set of 11 usability heuristics for Touchscreen-based Mobile Devices (TMD) was developed.

In a second iteration the initial set of 11 TMD heuristics was experimentally validated. Heuristic evaluations were performed on a Blackberry Storm 2 (9550) device running Blackberry OS. The following applications were analyzed: (1) menu, (2) name and address book, (3) calendar, (4) messaging and (5) camera. Evaluations were carried out by two separate groups of evaluators, in equal conditions. Each group was composed by two evaluators, of similar level of expertise. One group used only the TMD set, while the other group used only Nielsen's heuristics. Usability problems found by the two groups were then compared. A total of 53 usability problems were identified by the four evaluators. More usability problems were captured using TMD, than using Nielsen's usability heuristics. The usability problems identified by the group of evaluators using TMD were qualified as more severe (an average severity of 3.19, while the group using Nielsen's heuristics scored an average severity of 2.33). In a rough evaluation, it appears that the set of 11 TMD heuristics works better than Nielsen's heuristics. After analyzing the results along with experts' opinion, a set of 12 (refined) TMD usability heuristics was formalized using the previously described template [29].

In a third iteration, the TMD set of 12 usability heuristics was validated through a heuristic evaluation of a Samsung Galaxy Ace (GT-S5830L) device running Android OS (v. 2.3.4). The same applications as in the previous iteration were analyzed. This time, each group was composed by three evaluators. One group used the updated set of TMD, while the other group used only Nielsen's heuristics. A total of 37 usability problems were identified by the six evaluators. Once again, more usability problems were captured using TMD, than using Nielsen's heuristics. However, the usability problems identified by the group of evaluators

using TMD were qualified as less severe, getting an average severity of 1.67, while the group using Nielsen's heuristics scored an average severity of 2.02 [9].

In a fourth iteration, the TMD set of 12 usability heuristics was reviewed by several members of the "UseCV" Research Group, from the School of Informatics Engineering of the Pontifical Catholic University of Valparaíso — Chile. Each heuristic's definition was checked for understandability, clarity and consistency. The TMD set was refined based on experts' feedback [39].

A fifth iteration consisted of the following:

- A heuristic evaluation of Dropbox™.
- An inquiry test.

Dropbox™ is a file hosting service that offers cloud storage, file synchronization, personal cloud, and client software. Cloud storage is one of the most popular types of mobile cloud applications. Cloud-computing refers to the computer cluster which allows users to have immediate access to all their applications and files on any networking computer without restrictions by the desktop [40,41]. Gradually users have accepted the idea of cloud-computing, and have begun to move from traditional information systems to the "cloud" [42].

A total of 27 evaluators participated in the heuristic evaluation of Dropbox™ and were inquired afterward. All participants were students of the "Mobile Interfaces Design" undergraduate course, at the School of Informatics Engineering of the Pontifical Catholic University of Valparaíso — Chile. With the results of the heuristic evaluation, a statistical study was conducted. The number of usability issues associated to each heuristic was revised. The average severity and standard deviation were also studied.

The inquiry test participants were divided into two categories: (1) evaluators with No Previous Experience (NPE) and (2) evaluators with Previous Experience (PE). The NPE category contains 11 of 27 participants who had never performed a heuristic evaluation before. The PE category includes 16 of 27 participants, who had participated in few previous heuristic evaluations, some of them involving the TMD usability heuristics. The inquiry test results were analyzed using the ANOVA method; it analyzes the variance of one factor in order to check for significant differences among the opinion of the 27 participants.

#### 4. Analysis and results

Iterations 1 to 4 were briefly described in Section 3; the outcomes were published in previous works [9,29,39]. The results of the fifth iteration are described below.

##### 4.1. Heuristic evaluation

A total of 27 participants performed a heuristic evaluation of Dropbox™. As Apple iPhone is unfamiliar to most of the evaluators, two commonly used platforms were selected: Android and Windows Phone. The protocol of the heuristic evaluation proposed by Nielsen was followed [43]. The set of 12 TMD heuristics was used to inspect the application. Experiments' details are presented in Table 1.

Table 2 shows the number of problems by heuristics/experimental groups, the average severity and the associated standard deviation. Severity was estimated on a 0 (low) to 4 (high) scale.

The average number of detected usability problems among experimental groups is 28.33. The highest average number of usability problems was associated to heuristic TMD4 — consistency and standards (5.67). Several heuristics had no associated usability problems, in some experiments: TMD3, TMD5, TMD6, TMD7, TMD8, TMD9, TMD10, TMD11 and TMD12. This may be explained due to issues with heuristics' definition, low evaluators' experience or evaluators' subjectiveness.

**Table 1**  
Experiments and devices.

Experimental group	Number of participants	Device manufacturer	Device model	Operating System (O.S.)	Version of O.S.
1	5	LG	Optimus L3	Android	2.3.6
2	4	Samsung	Galaxy Fame	Android	4.1.2
3	6	Nokia	Lumia 505	Windows Phone	7.8
4	4	Samsung	Galaxy Note 10.1	Android	4.1.2
5	5	Motorola	RAZRi XT890	Android	4.1.2
6	3	Samsung	Galaxy Ace	Android	2.3.7

The heuristic with the highest associated severity average is TMD8 — efficiency of use and performance, with a score of 2.91 of a maximum of 4.00. The lowest severity average score is 1.45, associated to heuristic TMD9 — esthetic and minimalist design.

The standard deviation of severity among experiments is relatively stable at  $\approx 0.79$ . The heuristic with the highest standard deviation of severity is TMD5 — error prevention, with a score of 0.89. The lowest score is 0.66, associated to TMD10 — help users recognize, diagnose, and recover from errors.

In general, the experiments also showed the effect of inexperienced evaluators, unfamiliar with the set of TMD heuristics, leading to no problems associated to some heuristics and relatively high standard deviation. The study showed no major differences in number of problems detected, severity or standard deviation related to the number of participants, or the used device/platform.

#### 4.2. Inquiry test

After the heuristic evaluation was performed, an inquiry test was applied to all 27 evaluators as explained above. The survey was designed in order to capture the evaluators' perception about four dimensions: Utility (D1), Clarity (D2), Ease of use (D3), and Need of additional evaluation elements — “checklists” (D4). A five-point Likert scale was used. It allowed grading each heuristic, in a poor (1) to high (5) level.

Results were analyzed using the ANOVA method. It included the analysis of variance of one factor in order to look for differences among the opinion of the 27 participants. The ANOVA method used the following hypothesis (along with its opposite):

- $H_0$ : There are no significant differences among the opinion of the participants.
- $H_1$ : There are significant differences among the opinion of the participants.

Table 3 presents the p-value of the analysis of variance about the four dimensions mentioned above (D1, D2, D3 and D4). The significance level was considered  $\alpha = 0.05$ . In all cases, the p-value is lower than 0.05 hence  $H_0$  hypothesis is rejected. It can be concluded that there are significant differences between evaluators' perception on TMD heuristics Utility (D1), Clarity (D2), Ease of use (D3) and Need of additional evaluation elements — “checklists” (D4).

Additional analysis of variance was performed, considering separately the two categories of evaluators: (1) NPE — No Previous Experience and (2) PE — Previous Experience ones.

In the case of PE evaluators the p-value is lower than 0.05 for all dimensions; hence  $H_0$  hypothesis is rejected. This allows concluding that there are significant differences between evaluators' perception on all dimensions, despite the evaluators' previous experience.

**Table 2**  
Number of problems per heuristics/experimental groups, severity average and standard deviation.

TMD usability heuristics	Number of problems per experimental group						Average number of problems per experimental group	Severity average	Standard deviation
	Experimental group 1 (LG Optimus L3)	Experimental group 2 (Samsung Galaxy Fame)	Experimental group 3 (Nokia Lumia 505)	Experimental group 4 (Galaxy Note 10.1)	Experimental group 5 (Motorola XT890)	Experimental group 6 (Samsung Galaxy Ace)			
TMD1 — Visibility of system status	2	2	1	9	2	2	3.00	2.25	0.87
TMD2 — Match between system and the real world	1	3	5	6	1	5	3.50	2.08	0.69
TMD3 — User control and freedom	5	6	3	3	0	2	3.17	2.12	0.75
TMD4 — Consistency and standards	6	7	7	8	5	1	<b>5.67</b>	2.12	0.83
TMD5 — Error prevention	2	0	2	2	2	4	2.00	2.18	<b>0.89</b>
TMD6 — Minimize the user's memory load	3	1	1	2	1	0	1.33	2.29	0.79
TMD7 — Customization and shortcuts	8	1	0	1	4	0	2.33	2.48	0.82
TMD8 — Efficiency of use and performance	6	0	1	4	1	1	2.17	<b>2.91</b>	0.84
TMD9 — Esthetic and minimalist design	0	1	6	1	0	0	1.33	<b>1.45</b>	0.69
TMD10 — Help users recognize, diagnose, and recover from errors	0	0	3	2	1	0	1.00	2.45	<b>0.66</b>
TMD11 — Help and documentation	1	4	0	1	0	0	1.00	2.04	0.77
TMD12 — Physical interaction and ergonomics	8	0	0	1	1	1	1.83	1.50	0.76
Total number of problems per experimental group	42	25	29	40	18	16	Average number of problems among experimental groups: <b>28.33</b>		



**Table 3**Analysis of variances in D1, D2, D3 and D4 (significance level  $\alpha = 0.05$ ).

Dimension	p-Value (All 27 evaluators)	p-Value (11 NPE evaluators)	p-Value (16 PE evaluators)
Utility (D1)	9.38E-14	0.0098	8.49E-14
Clarity (D2)	6.48E-13	0.0004	1.04E-10
Ease of use (D3)	4.67E-10	<b>0.0868</b>	2.96E-10
Need of additional evaluation elements – “checklists” (D4)	1.13E-20	2.82E-10	4.36E-12

In the case of NPE evaluators only the p-value for Easy of use (D3) is higher than 0.05. In this particular case the  $H_0$  hypothesis is not rejected; this would imply that there is no evidence of significant differences between NPE evaluators' perception on Easy of use (D3). Likely the lack of experience prevents evaluators perceiving the ease of use of the heuristics.

The average perception of each TMD usability heuristics is presented in Table 4. In general, heuristics obtained high grades (greater than 3.00).

The dimension Ease of use (D3) obtained the lowest value (3.57). This is probably because even novice evaluators are familiar with more generic (Nielsen's) usability heuristics, but not with specific ones, in this case TMD. On the other hand, even if heuristics are perceived as quite useful (D1), clear (D2), and easy to use (D3), evaluators agree with the need of additional evaluation elements – “checklists” (D4), as a support in the discovery of usability problems (average score 4.00).

The correlation coefficients between averages (see Table 5) show that:

- There is a positive (and significant) correlation between Clarity (D2) and Ease of use (D3) with a score of 0.64; when a heuristic is perceived as clear, it is also perceived as easy to use, and vice versa.
- There is a positive correlation between Clarity (D2) and Utility (D1) with a score of 0.53; when a heuristic is perceived as clear, it is also perceived as useful, and vice versa.
- There is a positive correlation between Utility (D1) and the Need of additional elements – “checklists” (D4) with a score of 0.49; when a heuristic is perceived as useful evaluators feel the need of an associated checklist.
- There is a negative correlation between Ease of use (D3) and the Need of additional elements – “checklists” (D4) with a score of  $-0.37$ ; when a heuristic is perceived as easy to use, it does not need additional elements.

**Table 4**

Average perception scores for TMD usability heuristics in the four evaluated dimensions.

Heuristic	Utility (D1)	Clarity (D2)	Ease of use (D3)	Need of additional elements – “checklists” (D4)
TMD1	3.81	4.22	4.04	4.04
TMD2	4.04	4.26	4.04	4.04
TMD3	4.52	4.26	3.67	4.11
TMD4	4.33	4.19	4.07	4.04
TMD5	4.44	4.11	3.30	4.19
TMD6	4.07	3.93	3.15	4.11
TMD7	3.89	3.93	3.41	4.22
TMD8	4.11	3.89	3.04	4.30
TMD9	3.81	3.89	3.52	3.67
TMD10	4.22	4.19	3.33	3.93
TMD11	3.67	4.07	3.85	3.37
TMD12	3.63	3.78	3.44	4.04
Average	4.05	4.06	<b>3.57</b>	4.00

**Table 5**

Correlation coefficients between average perception scores.

	Utility (D1)	Clarity (D2)	Ease of use (D3)	Need of additional elements – “checklists” (D4)
Utility (D1)	1.00	0.53	−0.12	0.49
Clarity (D2)	0.53	1.00	<b>0.64</b>	−0.04
Ease of use (D3)	−0.12	<b>0.64</b>	1.00	− <b>0.37</b>
Need of additional elements – “checklists” (D4)	0.49	−0.04	− <b>0.37</b>	1.00

## 5. SMASH: a set of usability heuristics for smartphones

Based on the experimental results described in the previous section, the usability heuristics for Touchscreen-based Mobile Devices (TMD) were refined and renamed as SMARtphone's uSability Heuristics (SMASH). The new name highlights the focus on smartphones, rather than other touchscreen-based mobile devices. Each heuristic is described below with its ID, name, definition, explanation, examples, benefits and problems associated with misinterpretation.

### 5.1. SMASH1: visibility of system status

#### 5.1.1. Definition

The device should keep the user informed about all the processes and state changes through feedback and in a reasonable time.

#### 5.1.2. Explanation

Through the interaction with the device, the user should be able to perform different tasks. These actions could lead to a state change of the system, which should be communicated to the user in some way. Also, there are other events that are not triggered by user interaction, but they require later response, i.e.,: phone calls, video calls, text message reception, clock alarm, low battery notification, among others.

Some of the most common existing specific feedback types on mobile devices are: (1) Sound: (1.a) call ringtone, (1.b) message tone, (1.c) email tone, (1.d) low battery tone, (1.e) camera shutter tone; these sounds should be recognizable by the user (by default, according to conventions) and customizable. (2) Lights: (2.a) under the buttons, (2.b) notification lights, (2.c) camera led flash. (3) Graphic information: (3.a) static icons, (3.b) animated icons, (3.c) text notifications, (3.d) alert messages (i.e.,: push messages with distinctive icon(s) and text), (3.e) error messages (same structure as alert messages but different icon(s) and text). (4) Vibration.

#### 5.1.3. Examples

Fig. 1 shows an example of feedback, in this case, a status bar informing the user of the progress of the upload of a file.

#### 5.1.4. Benefits

- Better experience of use: precise and adequate feedback allows the user to react properly to events.
- Better knowledge/awareness of system status: precise and adequate feedback allows the user to know if there is any significative system status change.

#### 5.1.5. Problems

An aspect to consider when applying “Visibility of system status” heuristic is to distinguish between lack of effective feedback (as non-implemented) and lack of feedback due to performance issues. Many processes running on the device and/or battery running low can affect the performance, which can lead to “disabled” feedback.

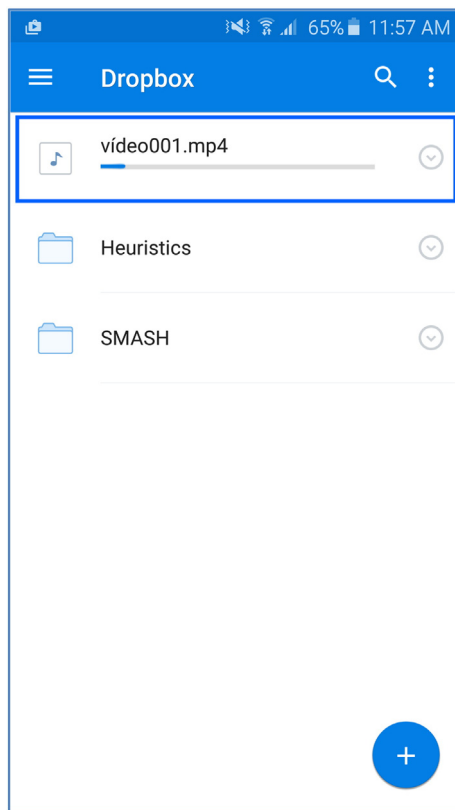


Fig. 1. SMASH1: visibility of system status.

## 5.2. SMASH2: match between system and the real world

### 5.2.1. Definition

The device should speak the users' language instead of system-oriented concepts and technicalities. The device should follow the real world conventions and display the information in a logical and natural order.

### 5.2.2. Explanation

Devices provide different interaction modes; more than just touching the keys on a keyboard. Nowadays, touchscreen-based mobile devices have particular characteristics that allow the user to interact with them in novel ways, such as: (1) Touch screen: it allows direct manipulation of displayed objects using the fingers or stylus. The user can drag, press, pinch, among other gestures. (2) Proximity sensor: used to deactivate the backlight when the user places the device to his/her ear. (3) Accelerometer: it detects variations of the acceleration when moving the device. (4) GPS: it allows determining global position coordinates automatically.

Through these new interaction modes, the user can perform tasks in a more intuitive way, by mimicking real world interaction rules. As an example, when scrolling down a long list, if the user "swipes" it with certain speed, the list will continue moving, mimicking the effect of inertia. It is expected that every interaction should depict a response similar to that expected in real world. Also, the language (text or icons) should be related to the real world and/or recognizable concepts.

### 5.2.3. Examples

Fig. 2 shows an example of match between system and the real world. The screen rotates when the user rotates the device.

### 5.2.4. Benefits

- Lower entry barrier for new users: when using recognizable concepts, the system can be more friendly and intuitive for new users.
- Minimize errors: when using recognizable concepts, the misinterpretation rate can be minimized; this leads to fewer errors.

### 5.2.5. Problems

When applying this heuristic, it should not be confused with "Consistency and standards". If some parts of the system are in a different language, it is not an issue related to "Match between system and the real world"; it is a consistency related issue.

## 5.3. SMASH3: User control and freedom

### 5.3.1. Definition

The device should allow the user to undo and redo his/her actions, and provide clearly pointed "emergency exits" to leave unwanted states. These options should be available preferably through a physical button or equivalent.

### 5.3.2. Explanation

When the user commits a mistake by introducing text, modifying configuration options, or just reaching an undesired state, the system should provide proper "emergency exits". These exits should easily allow the user to go from an undesired state to a desired one. Also, the user should be allowed to undo and redo his/her actions in a simple and intuitive way. On the other hand, the user should also be able to easily manage the applications that are running on the device, and the resources in use. When using the data network, the user should be able to control the amount of data that is being transmitted, and the associated time.

### 5.3.3. Examples

Fig. 3 shows an example of user control, allowing the user to easily manage the running applications and close them with a swipe.

### 5.3.4. Benefits

- Better experience of use: when having control over the system and a greater sense of freedom, the user perceives the product in a positive way, improving his/her experience of use.
- Better efficiency of use: when having control over the system and a greater sense of freedom, the user can improve the efficiency by controlling the resources in use.
- Better sense of ownership: when having control over the system and a greater sense of freedom, the user feels a strong sense of ownership by having almost total control over his/her device.

### 5.3.5. Problems

This heuristic should not be confused with the concept of flexibility and efficiency of use. Even though some effects of control are related to better efficiency (as explained in Benefits), they are dissimilar concepts. This heuristic of "User control and freedom" aims to repair or solve errors, give the user the chance to undo or redo his/her actions and have control over the resources of the device.

## 5.4. SMASH4: consistency and standards

### 5.4.1. Definition

The device should follow the established conventions, allowing the user to do things in a familiar, standard and consistent way.

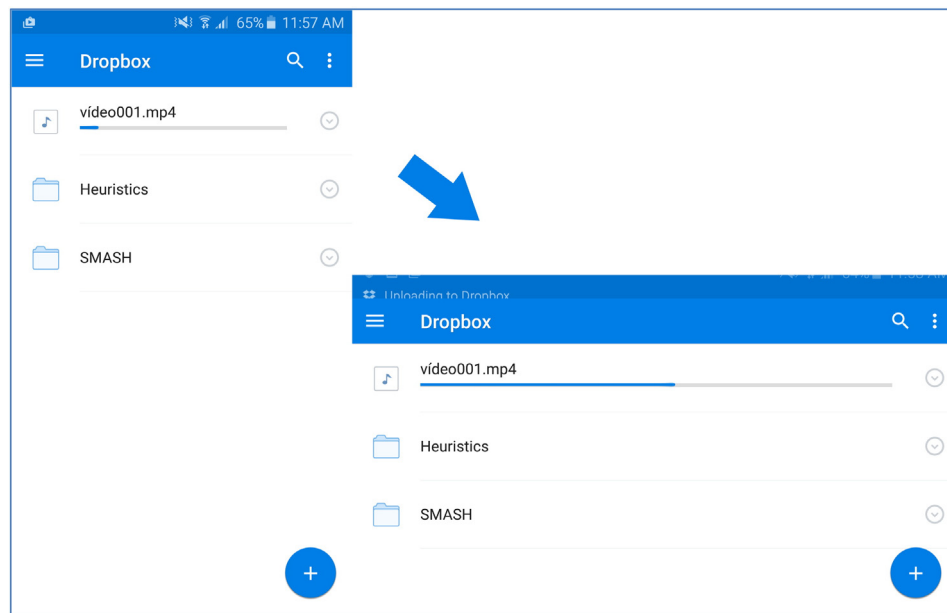


Fig. 2. SMASH2: match between system and the real world.

#### 5.4.2. Explanation

Many times, different parts of the system that are related and should be similar have different design or logic. In general, every concept presented in a contrasting way to the user's conception of the concept produces confusion in some degree. This confusion might lead to a decreasing efficiency of use or a low satisfaction, among other side effects. Considering all this, it is expected that the system should follow

standards and conventions in order to achieve an intuitive and easy-to-use interface.

Standards and conventions may be followed in three different dimensions: (1) From one part of the system to another, e.g., scrolling down lists of elements. (2) From one device to another of the same manufacturer (or related), e.g., pressing the power button once, the user can select a different sound profile (Nokia devices). (3) From one device to another with the same Operating System, e.g., every Android device has a “back” and a “home” button. Also, the system should be consistent in terms like: language, iconography, sounds, interaction, and so on. Same things should be the same.

#### 5.4.3. Examples

Fig. 4 shows an example of consistency and standards where the icons for photos, favorites and notifications are well known (standard).

#### 5.4.4. Benefits

- Lower entry barrier for new users: when following conventions and standards, a new user of a particular device can take advantage of its own experience using other devices, in order to use the new product in an intuitive way.
- Minimize errors: when having an intuitive and consistent system, the error rate can be minimized.

#### 5.4.5. Problems

When talking about consistency, one classical aspect is language consistency. Sometimes there are words that do not have a suitable translation, especially when related to technology. These words might be hard to translate, or even lose sense. Therefore, keeping a couple of words in another language may not be a “consistency” issue, but keeping a whole paragraph definitely is.

#### 5.5. SMASH5: error prevention

##### 5.5.1. Definition

The device should hide or deactivate unavailable functionalities, warn users about critical actions and provide access to additional information.

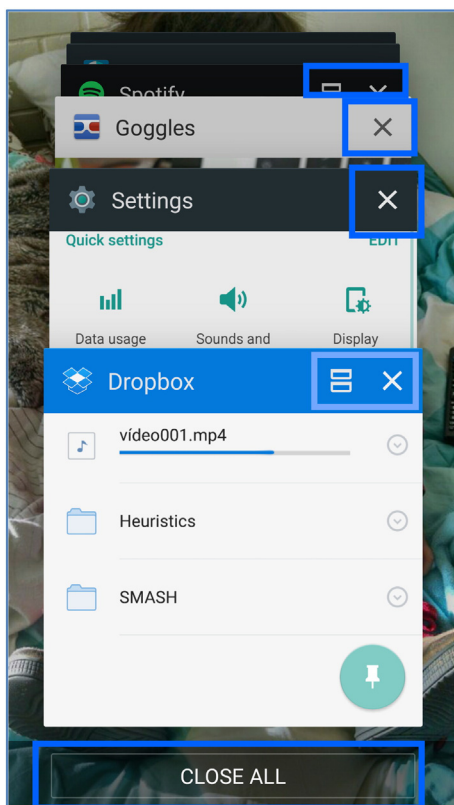


Fig. 3. SMASH3: user control and freedom.

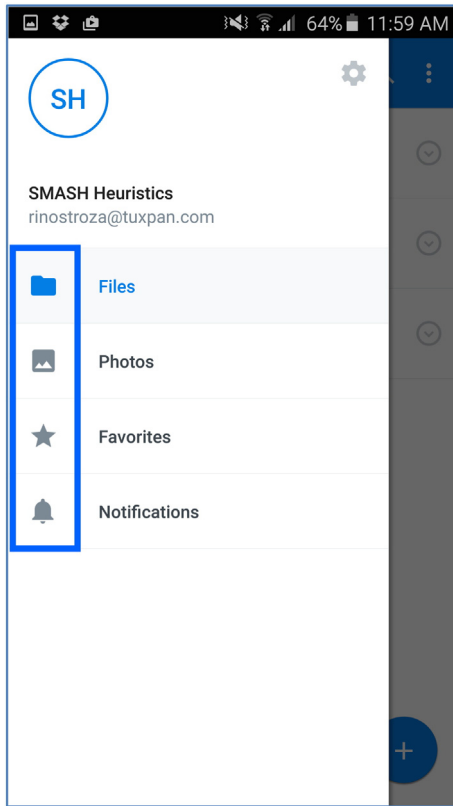


Fig. 4. SMASH4: consistency and standards.

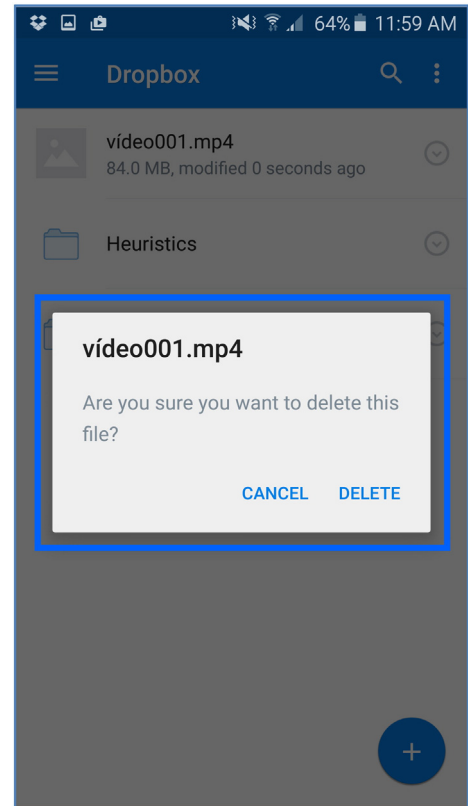


Fig. 5. SMASH5: error prevention.

### 5.5.2. Explanation

The device should try to be explicit regarding every option and functionality. Considering a small screen size, this could be a big challenge. In this way, icons play a very important role. Sadly, sometimes a small image is not enough to describe in detail a function or alike, and in order to fix this, the system should provide additional information on user's demand. The information should be displayed clearly, trying to avoid long dialog sequences. Also, the user should be warned, especially when some actions might have undesired effects. The potentially dangerous options should be placed in deeper menu levels (therefore, assigning a physical button to one of these options is not recommended).

On the other hand, on the physical aspect, the device should assure correct functioning. Actions like removing the battery pack, SIM card or memory card might abruptly stop the device and lose non-saved data. From this point of view, these actions should have some degree of security in order to avoid involuntary side effects.

### 5.5.3. Examples

Fig. 5 shows an appropriate example of warning messages for critical actions, in this case deleting a file from Dropbox™.

### 5.5.4. Benefits

- Better efficiency of use: by preventing errors, the user will lose less time solving them, which can lead to greater efficiency.
- Minimize errors: preventing errors leads to fewer errors.

### 5.5.5. Problems

This concept should not be confused with solving or repairing errors. If the user can trigger an error, that is a prevention issue.

## 5.6. SMASH6: minimize the user's memory load

### 5.6.1. Definition

The device should offer visible objects, actions and options in order to prevent users from having to memorize information from one part of the dialog to another.

### 5.6.2. Explanation

Human short-term memory is limited, so the user should not be forced to remember information from one part of the system to another. The instruction on how to use the system should be visible or easy to get; complex instructions should be simplified. When talking about mobile devices, the limited display size puts designers in a rough position regarding which elements of the interface should be hidden or minimized. In this way, it is important that sensitive information should be placed in a visible spot. Users should not write text from one part of the system to another; in these devices it is better to select and copy than write.

### 5.6.3. Examples

Fig. 6 shows an example of heuristic compliance. The Dropbox™ application shows a header with the name of the current folder, even if the user scrolls down the screen.

### 5.6.4. Benefits

- Reduce mental effort: when reducing memory load, the user's mental effort is also reduced.

### 5.6.5. Problems

For this particular heuristic, the evaluator might find some problems related to error prevention. The main issue here is to remark that this heuristic is strongly related to information overload. It is not about the



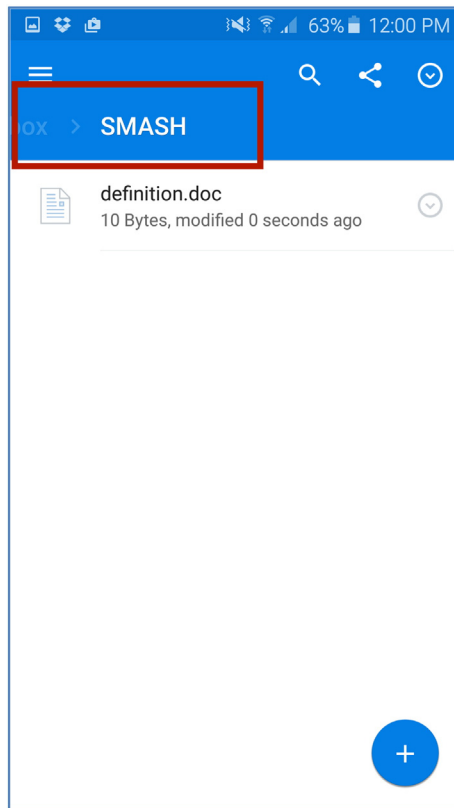


Fig. 6. SMASH6: minimize the user's memory load.

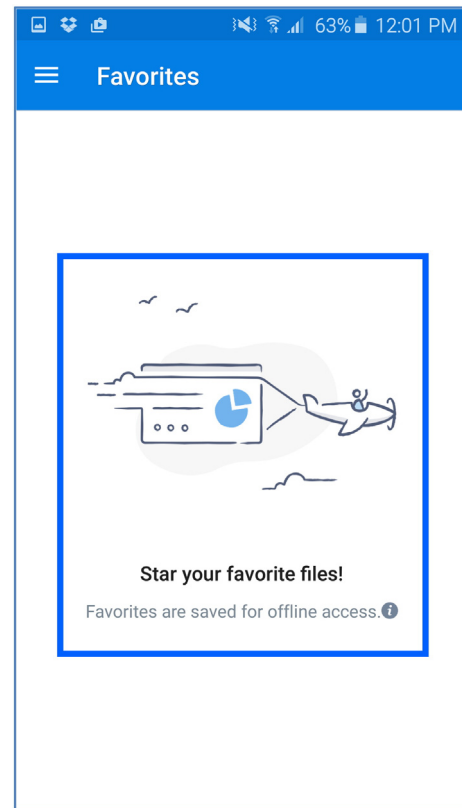


Fig. 7. SMASH7: customization and shortcuts.

information availability rather than the amount of information that the user needs to memorize in order to properly use the system.

#### 5.7. SMASH7: customization and shortcuts

##### 5.7.1. Definition

The device should provide basic and advanced configuration options, allow definition and customization of shortcuts to frequent actions.

##### 5.7.2. Explanation

Nowadays, a mobile device is almost an extension of its owner's body. It groups a lot of daily essential functionalities. From this perspective, customization and shortcut creation to these essential functionalities are some of the most frequent actions on a mobile device. Each user has his/her own needs and trying to satisfy all of them with a standard menu or interface can be a challenge. In this way, letting users to create their own shortcuts and customize most parts of the system may help. Through the access to advanced configuration options, expert users can improve their efficiency of use, and new users can get a deeper feel of ownership.

##### 5.7.3. Examples

Fig. 7 shows the action of creating a shortcut in compliance to the heuristic.

##### 5.7.4. Benefits

- Better sense of ownership: being able to customize his/her device, the user can have a deeper sense of ownership.
- Better efficiency of use: through the use of shortcuts and access to configuration options, the user can achieve a higher level of efficiency. The system can put the most frequent functionalities closer to the user and adjust itself to the user's work style.

#### 5.7.5. Problems

Clearly, there is a limit in terms of customization. Esthetic modifications might be possible in most parts of the system, but some things are not achievable. Significant modifications should be analyzed case by case.

#### 5.8. SMASH8: efficiency of use and performance

##### 5.8.1. Definition

The device should be able to load and display the required information in a reasonable time and minimize the required steps to perform a task. Animations and transitions should be displayed smoothly.

##### 5.8.2. Explanation

The processing power of most touchscreen-based mobile devices has increased exponentially in past years, even having devices with four (or more) processors. The match between hardware capabilities and software needs is not always the best. It is expected for basic software to be compatible with the hardware, especially with processing capabilities, in order to prevent black screens and long waiting times. Also, the animations, effects and transitions should be displayed fluidly with no-interruptions.

Another critical point is the length of the sequence of steps in order to perform a task. Complex tasks, potentially dangerous ones or non-frequent ones might contain several steps as security reinforcement. Simple tasks or frequent ones should be short. If the user wants to set an alarm at 4 am, he/she does not expect a 4 steps process.

### 5.8.3. Examples

Loading screen rendering the device unusable for an extended period of time. Due to its dynamic characteristic, this heuristic cannot be illustrated by an image.

### 5.8.4. Benefits

- Better efficiency of use: lower response times and better performance lead to an efficient system.

### 5.8.5. Problems

Issues related to performance of hardware should be separated from issues related to network performance. Even though these problems affect usability, they are not part of the scope of this research, considering that they are affected by several complex factors. Regarding to the length of a sequence of steps to perform a task, the limit between normal and excessive is subjective. The evaluator should use his/her own criterion.

## 5.9. SMASH9: esthetic and minimalist design

### 5.9.1. Definition

The device should avoid displaying unwanted information overloading the screen.

### 5.9.2. Explanation

For devices with an old release date, each information unit displayed in a small screen involves lower performance. Designers should be careful when displaying information through the screen. Also, overloaded interfaces may produce stress to the user.

### 5.9.3. Examples

Fig. 8 shows an example of minimalist design with few controls for most popular functions: Dropbox (all files), Pictures, Favorites and Notifications. For further options there is a secondary menu (upper-right control).

### 5.9.4. Benefits

- Better performance: when using a minimalist design the device uses potentially fewer resources, which can lead to better performance.
- Minimize exhaustion: minimizing the amount of visual information can lead to less users stress and exhaustion.

### 5.9.5. Problems

The questions here are: Where is the limit? How to distinguish between a minimalist design and a deficient interface? Once again, a subjective criterion on the part of the evaluator is required. If the evaluator faces an overload interface, clearly there is an issue related to this heuristic.

## 5.10. SMASH10: help users recognize, diagnose, and recover from errors

### 5.10.1. Definition

The device should display error messages in a language familiar to the user, indicating the issue in a precise way and suggesting a constructive solution.

### 5.10.2. Explanation

When an error occurs, the user does not need technicalities or cryptic alert messages. The user needs clear feedback messages, in a recognizable language, with instructions on how to recover from the error.

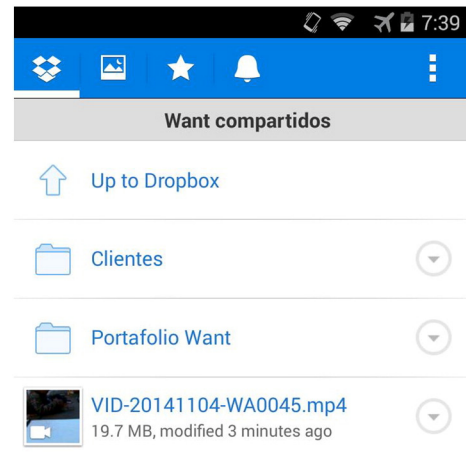


Fig. 8. SMASH9: esthetic and minimalist design.

### 5.10.3. Examples

Fig. 9 shows a usability issue related to SMASH10. The error message uses a system-oriented term that does not help the user on how to recover from error (“process.com.android.settings has stopped”).

### 5.10.4. Benefits

- Less user frustration: an error by itself is frustrating. Help messages and instruction on how to recover from the error might lower down user frustration.

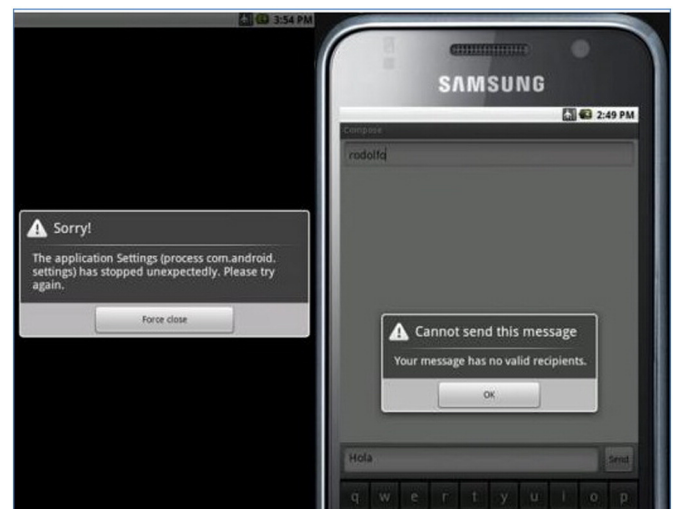


Fig. 9. SMASH10: help users recognize, diagnose, and recover from errors.

### 5.10.5. Problems

The evaluator should distinguish between prevention and help to recover from the error. The main difference here is time. If the error has not happened yet, we are talking about prevention, otherwise, it might be an issue related to this heuristic.

## 5.11. SMASH11: help and documentation

### 5.11.1. Definition

The device should provide easy-to-find documentation and help, centered on the user's current task and indicating concrete steps to follow.

### 5.11.2. Explanation

The device should provide access to detailed information about the available functionalities in a clear and simple way, from any part or state of the system where the user is located. It is recommended that this information should be included on the device. If not, the documentation should be available on a website and/or printed.

### 5.11.3. Examples

Fig. 10 shows an example of help and documentation where the help system of Dropbox™ is displayed.

### 5.11.4. Benefits

- Minimize errors: if the user has detailed information about the usage of the system, he/she will avoid running into an error.
- Increase user's knowledge of the system: documentation plays a very important role in turning a new user into an expert one.
- Better efficiency of use: through an increased knowledge of the system and fewer errors, the user might improve his/her efficiency of use.

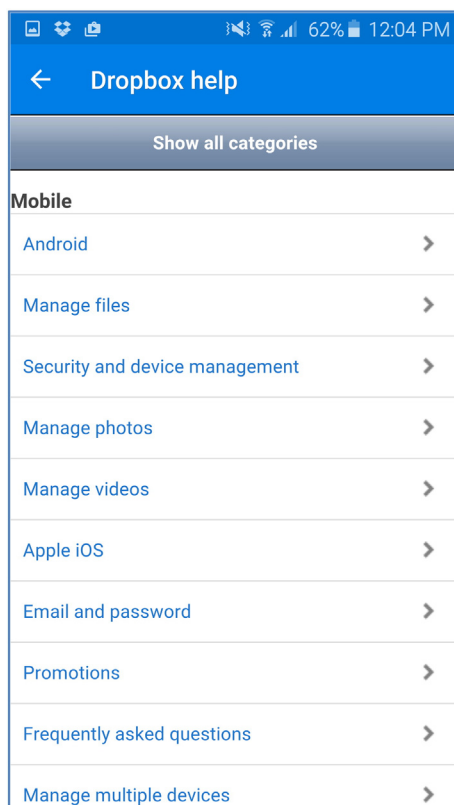


Fig. 10. SMASH11: help and documentation.

### 5.11.5. Problems

The main difficulty when applying this heuristic is how to differentiate it from error prevention. Even though documentation and help messages might prevent errors, this heuristic is rather focused on instructions on how to use the system, additional information about options and configuration, and so on.

## 5.12. SMASH12: physical interaction and ergonomics

### 5.12.1. Definition

The device should provide physical buttons or the equivalent for main functionalities, located in positions recognizable by the user, which should fit the natural posture (and reach) of the user's dominant hand.

### 5.12.2. Explanation

Mobile devices are designed as hand-held devices. From this point of view, ergonomics and comfort play a very important role in the interaction between user and device. Any product that does not have a shape, weight, dimensions or buttons' position matching the normal posture of the palm might produce exhaustion. Buttons should be placed in positions recognizable by the user based on his/her experience with other similar devices, e.g., the camera shutter button is often placed in the top-right corner of the device in landscape mode.

On the other hand, main functionalities should have assigned a physical button or the equivalent. Some examples of these functionalities are: power on/off, answer/end a phone call, volume control, camera shutter, lock/unlock device, open/close applications menu, go-back and go-home. These buttons should be labeled clearly and should not be assigned to a large number of different functionalities.

### 5.12.3. Examples

Fig. 11 shows a Samsung Galaxy S6 device that, with its 5.1 in. screen, does not fit the user's palm. The circle marks the thumb's reach area. This is a usability issue according to this heuristic.

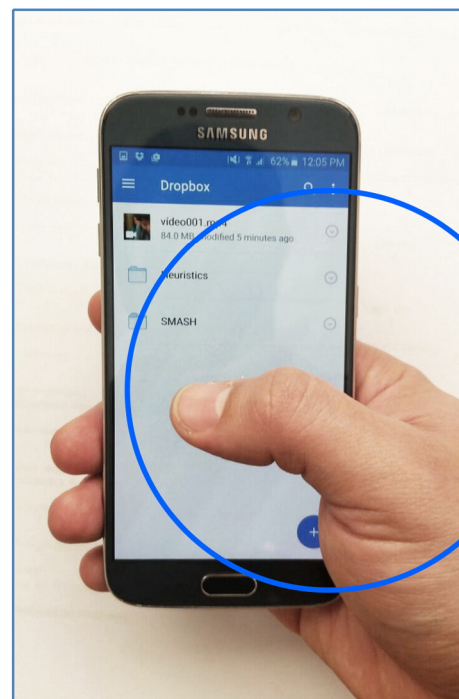


Fig. 11. SMASH12: physical interaction and ergonomics.

### 5.12.4. Benefits

- Ease of use: placing buttons in recognizable positions, clearly labeled, helps making a more intuitive product; an intuitive product does not need further instructions and it is easy to use.
- Decrease exhaustion: if the device has a shape and dimensions that match the normal posture of the palm it will decrease the hand's exhaustion and other side effects.
- Better efficiency of use: when having “shortcuts”, in this case buttons, for the most frequent functionalities, the user should improve his/her efficiency of use.

### 5.12.5. Problems

A possible problem when applying this heuristic might be the case of users with special needs. In this case, the evaluator must use his/her own criterion; there are some devices that result uncomfortable for most users and that is an issue related to this heuristic. A very important aspect to consider is “popularity”. A popular device does not necessarily have a good usability level. Each case should be analyzed separately.

## 6. Conclusions

Usability is a key differentiating factor for mobile products and applications. We try to satisfy the need of proper tools for detecting usability issues in smartphones by proposing a set of specific usability heuristics for these products.

The methodology used in the development of the heuristics has allowed us to perform an iterative process [38]. Here we present results based on five iterations. The flexibility of the methodology has supported a reliable development and refinement process, resulting in a set of 12 heuristics named SMASH: SMArtphone's uSability Heuristics (previously referred to as TMD: usability heuristics for Touchscreen-based Mobile Devices). Compared to early versions, most of the heuristics definitions are shorter. Sometimes “less is more”, especially if evaluators have to deal with unfamiliar heuristics.

Many usability heuristics sets have been proposed for specific software products; however, there is currently no clear protocol for heuristics' validation. The whole process of developing usability heuristics it is yet to be formalized. We have conducted several experiments, validating and refining the proposed set of heuristics iteratively. We have shown SMASH to be an effective and reliable tool, however further validation may still be performed.

We have identified a large amount of usability problems through the experiments conducted while developing SMASH. We have used such problems in order to develop a (preliminary) set of design patterns for smartphones and mobile applications [44]. These patterns still require validation and refinement.

## Acknowledgments

The authors would like to thank all the participants involved in the experiments that the present study required, especially the members of the “UseCV” Research Group. The work was highly supported by the School of Informatics Engineering (Escuela de Ingeniería Informática) of the Pontifical Catholic University of Valparaíso (Pontificia Universidad Católica de Valparaíso – PUCV) – Chile. Rodolfo Inostroza has been granted the PUCV 2013–2014 Graduate Scholarship.

## References

- [1] N. Bevan, International standards for HCI and usability, *Int. J. Hum. Comput. Stud.* 55 (4) (2001) 533–552.
- [2] P. Choe, C. Liao, W. Sun, Providing customisation guidelines of mobile phones for manufacturers, *Behav. Inform. Technol.* 31 (10) (2012) 983–994.
- [3] M. Kim, J. Park, Mobile phone purchase and usage behaviours of early adopter groups in Korea, *Behav. Inform. Technol.* 33 (7) (2014) 693–703.
- [4] Z. Mack, S. Sharples, The importance of usability in product choice: a mobile phone case study, *Ergonomics* 52 (12) (2009) 1514–1528.
- [5] W.H. Delone, E.R. McLean, The Delone and McLean model of information systems success: a ten-year update, *J. Manag. Inf. Syst.* 19 (4) (2003) 9–30.
- [6] A. Bruun, J. Stage, Barefoot usability evaluations, *Behav. Inform. Technol.* 33 (11) (2014) 1148–1167.
- [7] J.R. Lewis, Usability testing, in: G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics*, John Wiley & Sons, New York, NY, USA 2012, pp. 1267–1312.
- [8] J. Nielsen, Heuristic evaluation, in: J. Nielsen, R.L. Mack (Eds.), *Usability Inspection Methods*, John Wiley & Sons, New York, NY, USA 1994, pp. 25–62.
- [9] R. Inostroza, C. Rusu, S. Roncagliolo, C. Jimenez, V. Rusu, Usability heuristics validation through empirical evidences: a touchscreen-based mobile devices proposal, *Proceedings SCCC 2012, Valparaíso, Chile, IEEE 2012*, pp. 60–68.
- [10] A. Baz, I. Ferreira, M. Álvarez, R. García, *Dispositivos Móviles*, Universidad de Oviedo, Spain, 2009.
- [11] B. Ballard, *Designing the Mobile User Experience*, John Wiley & Sons, New York, NY, USA, 2007.
- [12] G. Schiefer, M. Decker, Taxonomy for mobile terminals – a selective classification scheme, in: J. Filipe, D.A. Marca, B. Shishkov, M. van Sinderen (Eds.), *Proceedings ICE-B 2008, INSTICC Press, Porto, Portugal 2008*, pp. 255–258.
- [13] R.B. Miller, Human ease of use criteria and their tradeoffs, *IBM Report TR 00.2185*, IBM Corporation, Poughkeepsie, NY, USA, 1971.
- [14] N. Bevan, J. Kirakowski, J. Maissel, What is usability? in: H.J. Bullinger (Ed.), *Proceedings of the 4th Int. Conf. on HCI, Stuttgart, Elsevier, Germany, 1991*.
- [15] F. Sarodnick, H. Brau, *Methoden der Usability Evaluation: Wissenschaftliche Grundlagen und praktische Anwendung*, Huber Hans, Bern, Switzerland, 2006.
- [16] B. Shackel, The concept of usability, *Proceedings of IBM Software and Information Usability Symposium, Poughkeepsie, NY, USA 1981*, pp. 1–30.
- [17] B. Shackel, S.J. Richardson, *Human Factors for Informatics Usability*, Cambridge University Press, New York, NY, USA, 1991.
- [18] A. Sonderegger, J. Sauer, A. Tattersall, Influencing Factors in Usability Tests: The Testing Situation, The Product Prototype, and The Test User (PhD Thesis) Université de Fribourg, Switzerland, 2010.
- [19] ISO 9241–210, Human-centered design process for interactive systems, *International Organization for Standardization 2010* (Geneva, Switzerland).
- [20] ISO/IEC 25000, Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE, *International Organization for Standardization, 2014* (Geneva, Switzerland).
- [21] ISO 9241–11, Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)-Part 11, Guidance on Usability, *International Organization for Standardization, Geneva, Switzerland, 1998*.
- [22] ISO 20282–1, Ease of Operation of Everyday Products – Part 1: Design Requirements for Context of Use and User Characteristics, *International Organization for Standardization, Geneva, Switzerland, 2006*.
- [23] J. Nielsen, *Usability Engineering*, Morgan Kaufmann, San Francisco, CA, USA, 1993.
- [24] C. Rusu, V. Rusu, S. Roncagliolo, C. González, Usability and user experience: what should we care about? *Int. J. Inf. Technol. Syst. Approach* 8 (2) (2015) 1–12.
- [25] J.R. Lewis, Lessons learned... and yet to be learned, *Int. J. Hum. Comput. Interact.* 30 (9) (2014) 663–684.
- [26] J. Heo, D. Ham, S. Park, C. Song, W.C. Yoon, A framework for evaluating the usability of mobile phones based on multi-level, hierarchical model of usability factors, *Interact. Comput.* 21 (4) (2009) 263–275.
- [27] Y.S. Lee, S.W. Hong, T.L. Smith-Jackson, M.A. Nussbaum, K. Tomioka, Systematic evaluation methodology for cell phone user interfaces, *Interact. Comput.* 18 (2) (2006) 304–325.
- [28] L.H. Salazar, T. Lacerda, J. Nunes, C. Gresse von Wangenheim, A systematic literature review on usability heuristics for mobile phones, *Int. J. Hum. Comput. Interact.* 5 (2) (2013) 50–61.
- [29] R. Inostroza, C. Rusu, S. Roncagliolo, C. Jimenez, V. Rusu, Usability heuristics for touchscreen-based mobile devices, *Proceedings ITNG 2012, Las Vegas, USA, IEEE 2012*, pp. 662–667.
- [30] A. Salgado, A. Freire, Heuristic evaluation of mobile usability: a mapping study, *Human-computer interaction. Applications and services, Lect. Notes. Comput. Sc.* 8512 (2014) 178–188.
- [31] E. Bertini, S. Gabrielli, S. Kimani, Appropriating and assessing heuristics for mobile computing, *Proceedings AVI'06, Venezia, Italy, ACM 2006*, pp. 119–126.
- [32] B.C. Cunha, O. Neto, M. Pimentel, A heuristic evaluation of a mobile annotation tool, *Proceedings WebMedia'13, Salvador, Brazil, ACM 2013*, pp. 89–92.
- [33] O. Neto, M. Pimentel, Heuristics for the assessment of interfaces of mobile devices, *Proceedings WebMedia'13, Salvador, Brazil, ACM 2013*, pp. 93–96.
- [34] H. Korhonen, Comparison of playtesting and expert review methods in mobile game evaluation, *Proceedings Fun and Games'10, Leuven, Belgium, ACM 2010*, pp. 18–27.
- [35] A. Ponnada, A. Kannan, Evaluation of mobile games using playability heuristics, *Proceedings ICACCI'12, Chennai, India, ACM 2012*, pp. 244–247.
- [36] K. Väänänen-Vainio-Mattila, M. Wäljas, Development of evaluation heuristics for web service user experience, *Proceedings CHI EA'09, Boston, MA, USA, ACM 2009*, pp. 3679–3684.
- [37] K. Väänänen-Vainio-Mattila, M. Wäljas, Developing an expert evaluation method for user experience of cross-platform web services, *Proceedings MindTrek'09, Tampere, Finland, ACM 2009*, pp. 162–169.
- [38] C. Rusu, S. Roncagliolo, V. Rusu, C. Collazos, A methodology to establish usability heuristics, *Proceedings ACHI 2011, Guadeloupe, France, IARIA 2011*, pp. 59–62.
- [39] R. Inostroza, C. Rusu, S. Roncagliolo, V. Rusu, Usability heuristics for touchscreen-based mobile devices: update, *Proceedings ChileCHI '13, Temuco, Chile, ACM 2013*, pp. 24–29.



- [40] M.A. Babar, Perspectives and reflections on cloud computing and Internet technologies from NordiCloud 2012, Proceedings NordiCloud '13, Oslo, Norway, ACM 2013, pp. 72–79.
- [41] R.K. Lomotey, R. Deters, CSB-UCC: cloud services brokerage for ubiquitous cloud computing, Proceedings MEDES '13, Luxembourg, Luxembourg, ACM 2013, pp. 100–107.
- [42] X. Li, J. Liu, J. Han, Q. Zhang, The architecture design of micro-learning platform based on Cloud Computing, Proceedings ICCS '11, Wuhan, China, ACM 2011, pp. 80–83.
- [43] J. Nielsen, R. Molich, Heuristic evaluation of user interfaces, Proceedings ACM CHI '90 Conference, Seattle, WA, USA 1990, pp. 249–256.
- [44] R. Inostroza, C. Rusu, S. Roncaliolo, V. Rusu, Design patterns for touchscreen-based mobile devices: users above all! Proceedings ChileCHI '13, Temuco, Chile, ACM 2013, pp. 50–51.



Silvana Roncagliolo is full professor at Pontificia Universidad Católica de Valparaíso (PUCV), in Chile. She has a Master of Science from Oregon State University (USA). She has been program committee member, and also chair of conferences and other academic events. Her research interests consider HCI and Usability Engineering. She is founding member of the Chilean ACM-SIGCHI Chapter and of the “UseCV” Research Group in HCI at PUCV; also she has been named IARIA fellow.



Rodolfo Inostroza is PhD candidate in Informatics Engineering at Pontificia Universidad Católica de Valparaíso (PUCV), in Chile. He is also a part-time professor at PUCV and owns a small software company (Randomstudio). His research interests focus on Human–Computer Interaction (HCI), Usability Engineering and Mobile Computing. He is founding member of the Chilean ACM-SIGCHI Chapter and of the “UseCV” Research Group in HCI at PUCV.



Virginica Rusu is professor at Universidad de Playa Ancha de Ciencias de la Educación, in Chile. She is PhD in Cybernetics and Statistics from Babes Bolyai University (Romania). She worked in several companies and research institutes in Romania. Since 1997 she is fully dedicated to academia. Her main research interests focus on qualitative and quantitative approaches in HCI, User Experience, and Service Science. She is founding member of the Chilean ACM-SIGCHI Chapter and of the “UseCV” Research Group in HCI at Pontificia Universidad Católica de Valparaíso.



Cristian Rusu is full professor at Pontificia Universidad Católica de Valparaíso (PUCV), in Chile. He is PhD in Applied Informatics from Technical University of Cluj-Napoca (Romania). He worked in several software companies and research institutes in Romania and Chile. Since 1999 he is fully dedicated to academia. He serves in editorial boards and conference program committees and disseminates Usability's importance, in Chile and Latin America. His research interests focus on HCI, Usability, User Experience, and Service Science. Former Chair of the Chilean ACM-SIGCHI Chapter, he is also the head of the “UseCV” Research Group in HCI at PUCV.



César A. Collazos is full professor at Universidad del Cauca (Colombia). He is PhD in Computer Science from Universidad de Chile. His research areas include Computer Supported Collaborative Learning, HCI, Computer Supported Cooperative Work. Author of many research papers, he serves in editorial boards and conference program committees and constantly disseminates HCI importance in Latin America.