

# Comparative Evaluation? Yes, But With Which Alternative UI?

Hayet Hammami <sup>(1)(2)</sup>  
<sup>(1)</sup> Univ. Tunis ElManar, Faculty  
of Science of Tunis.  
CRISTAL Laboratory

Gaëlle Calvary <sup>(2)</sup>  
<sup>(2)</sup> Univ. Grenoble Alpes, CNRS,  
Grenoble INP, LIG, F-38000  
Grenoble France  
FirstName.Name@imag.fr

Meriem Riahi <sup>(3)</sup>  
<sup>(3)</sup> National Higher Engineering  
School of Tunis  
Taha Hussein Avenue, Tunis.  
Tunisia  
meriem.riahi@ensit.rnu.tn

Faouiz Moussa <sup>(4)</sup>

Sara Bouzit <sup>(2)</sup>  
<sup>(4)</sup> CRISTAL Laboratory  
National School of Computer  
Sciences, Mannouba University,  
Tunisia  
faouzimoussa@gmail.com

**User's feedback provides valuable information suitable to help designers to improve their work. In this paper, we present a study on user's feedback when evaluating a User Interface (UI) by comparison. Our aim is to define the properties that the alternative UI must satisfy to maximize the benefits of comparative evaluation.**

**The UIs considered in the study were designed using the CAMELEON Reference Framework (CRF), covering variations at each level of abstraction. We study the impact of each variation on users' feedback. We show that when the alternative design refers to the same task model as the original one but using a different abstract UI, the number of negative returns is significantly higher, making the comparative evaluation more productive.**

*HCI. UI Design. UI Evaluation. Feedback. CAMELEON reference framework.*

## 1. INTRODUCTION

Users' feedback is important for the design process. Critiques, opinions and suggestions are valuable information to improve the design (Nguyen, 2017), (Hui, 2015). B. Gates tells that *"We improve our products, based on feedback, until they're the best"*. Traditionally when asking for users' feedback, designers present only one User Interface (UI), the one to be tested. However, as demonstrated in (Tohidi, 2006) providing several design alternatives to the assessor increases the amount of feedback and facilitates comparative reasoning. However, to the best of our knowledge, there is not yet any research studying the characteristics that the alternative design must satisfy in order to maximize the benefits of the evaluation.

This research aims to improve the comparative assessment by producing the optimal alternative design, i.e. maximizing returns to the original UI. In this paper, we report an experiment, where we use the CAMELEON Reference Framework to generate UIs depending on different classes of variations of a design and we study their impact on users' feedback through a comparative evaluation. This study is expected to support the definition of criteria that the

alternative design must meet to maximize feedback on the UI of interest.

In this paper, we describe our approach and experiment. First, we provide a discussion on related work. Then, we present preliminary studies for conducting the experiment. Finally, we report the experiment and discuss the evaluation results.

## 2. RELATED WORK

### 2.1. Testing many is better than testing one

Working with examples has proven to have several benefits for both the learning process and the outcome (Lee, 2010). Accordingly, designers often use examples for inspiration, which offers contextualized illustrations of how form and content integrate. According to (Herring, 2009), examples are crucial to design activities. They support both the generation of new ideas and the selection of interesting ones. Examples enable to identify limitation of previous designs, as well as reinterpretation and recombination of ideas (Masson, 2011).

Besides using alternative designs and examples during the design process, it has been proved that using multiple designs can also improve the results of

the design evaluation. Creating multiple prototypes facilitates comparative reasoning, grounds team discussion, and enables situated exploration (Tohidi, 2006).

In (Wiklund, 1992), Wiklund et al. studied the impact of the fidelity of software prototypes on the perception of usability. The result of their research lead to this observation:

In studies such as this one, we have found subjects reluctant to be critical of designs when they are asked to assign a rating to the design. In our usability tests, we see the same phenomenon even when we encourage subjects to be critical. We speculate that the test subjects feel that giving a low rating to a product gives the impression that they are “negative” people, that the ratings reflect negatively on their ability to use computer-based technology, that some of the blame for a product’s poor performance falls on them, or that they don’t want to hurt the feelings of the person conducting the test.

Dicks et al. (Dicks, 2002) show that when people are shown multiple prototypes, they could feel less pressured to impress the experimenters by praising a particular design. Being presented with multiple alternative designs may allow for a more accurate comparative evaluation.

Tohidi et al. (Tohidi M. W., 2006) examined the differences that would occur between a usability test that exposed users to a single design, and one where they were exposed to three different alternatives. This study showed that designs are rated higher when seen alone than they would be when seen in comparison with other designs. Additionally, the number of designs given to evaluate can influence the quantity, quality, and responsiveness of the feedback.

However, in their study, there was no discussion about how to choose the alternative designs given to users during the usability test.

In our work, we investigate the impact of the alternative design given during the comparative evaluation on the user’s feedback. We believe that the choice of this UI could remarkably influence the feedback received from users.

## 2.2. CAMELEON reference framework

CAMELEON is a Reference Framework (CRF) for the development and execution of UIs in multiple contexts of use, a context of use being defined as a triplet <User, Platform, Environment>. It structures the design process into four levels of abstraction for ensuring UIs consistency by design, and thereby

saving costs of development and maintenance (Calvary, 2003). The four levels of abstraction are (Figure1):

- **Task and domain** is the top level that describes the users tasks the interactive system must support, together with the information (the domain concepts) that are manipulated by these tasks,
- **Abstract User Interface (AUI)** makes design decision about grouping and navigation,
- **Concrete User Interface (CUI)** makes design decisions about rendering. It defines how the UI is perceived and can be manipulated by users,
- **Final User Interface (FUI)** is the running UI. Design decisions are about the programming or mark-up language to be used.

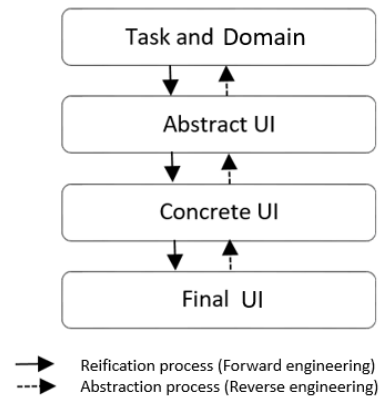


Figure 1. A simplified version of the CAMELEON Reference Framework

These four levels of abstraction are structured with a relationship of reification (going from an abstract level to a concrete one) and/or abstraction (going from a concrete level to an abstract one) (Calvary, 2003).

## 3. PRELIMINARY STUDIES

We conducted an experiment in which participants had to evaluate different designs. Each design is generated depending on a different variation of the first three abstract levels of CRF: task and domain, AUI and CUI.

Our goal is to identify which UI influences user’s feedback the most and then to define the criteria that alternative UIs must meet to maximize returns on the UI produced by the designer.

We assume that these variations affect the user’s feedback as follows:

**H1:** The choice of the alternative design conditions the user's return to the original one.

**H2:** The comparative evaluation will be more productive with a design that has the same task model but a different abstract UI.

### 3.1. Method

We start by creating different designs for the same application following CRF. Therefore, we first define the variations:

- At the Task and concepts level, there are three classes of possible variations:
  - Structure of tasks, e.g. by factorizing tasks and/or concepts,
  - Operators between tasks, e.g. by replacing a choice operator by a sequence one,
  - Task decorations, e.g. by declaring a task as being frequent.
- At the Abstract UI level, there are two classes of variations:
  - Grouping, e.g. by putting together all frequent tasks to separate them from non-frequent ones,
  - Navigation, e.g. by launching the interactive system on the dialog space devoted to frequent tasks and by forcing non-frequent tasks to be accessible through frequent tasks.
- At the Concrete UI level, there are two classes of variations:
  - Interactors, e.g. by preferring graphical widgets (radio buttons, check boxes, etc.) to vocal interaction,
  - Parameters, e.g., by setting colour, size, position, etc. of graphical widgets.

We did not consider variations at the Final UI level of abstraction as we decided to work with paper-based prototypes, quick and thereby inexpensive to make.

### 3.2. Case study

The case study is about checking and managing security of houses remotely. The motivation is that such an application is widespread, and thereby easy to explain.

Three main tasks were proposed to the user: **(1)** control the access to the house, **(2)** control the security cameras and **(3)** manage the alarm system. Controlling the access to the house allows the user to remotely lock or unlock the doors and other entries of the house. The user can watch the feed from the security cameras (in real time or recorded), send or

delete them. Finally managing the alarm system allows the user to program it and to stop the alarm when triggered.

### 3.3. Design alternatives

The Task models below (figures 2 and 5) are designed using Flexilab, a multimodal editor created by (N. Hili, 2015).

Figures 3 and 4 present two alternatives of AUI related to Task model 1. Figure 6 proposes an AUI for Task model 2.

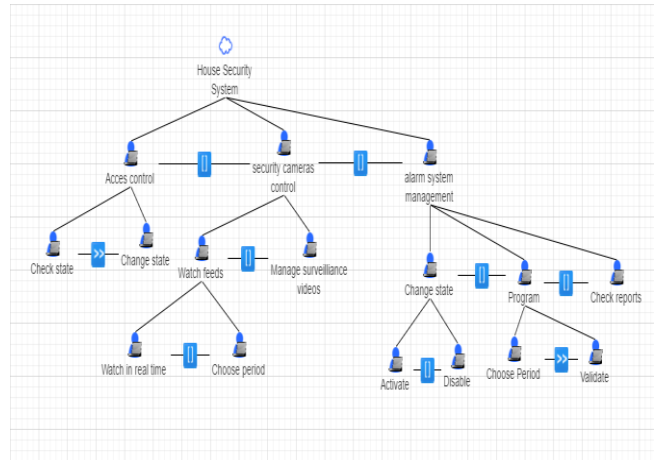


Figure 2. Task model 1

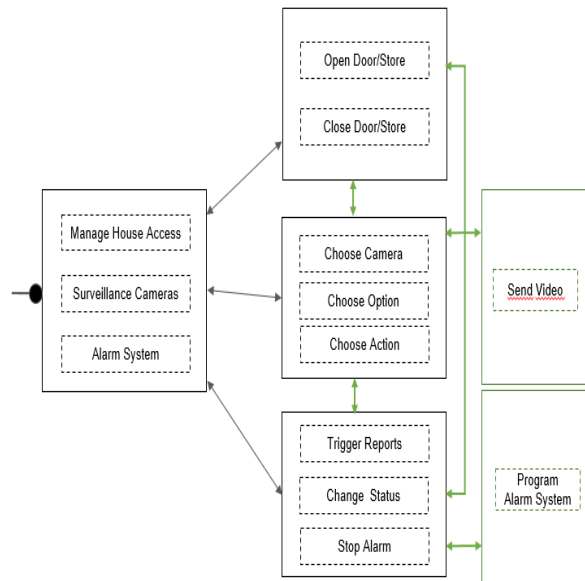


Figure 3. AUI\_1

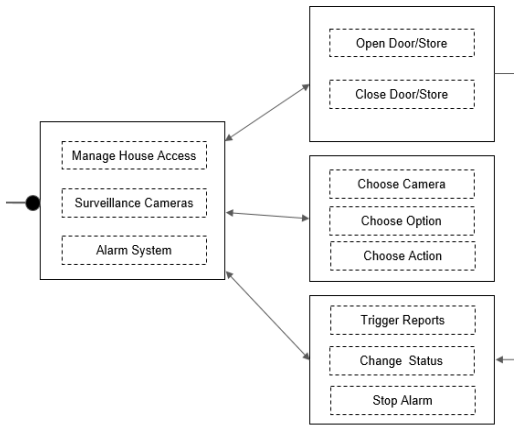


Figure 4. AUI\_2

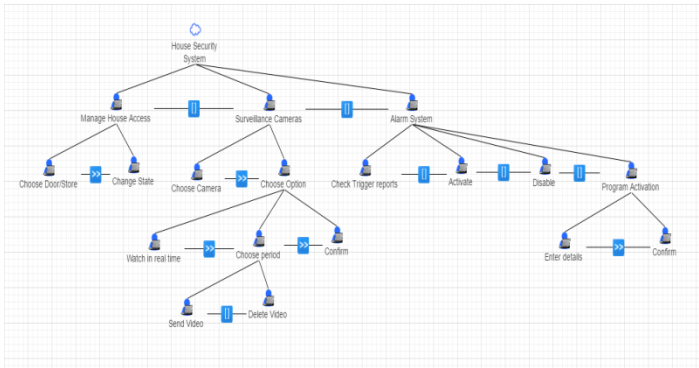


Figure 5. Task model 2

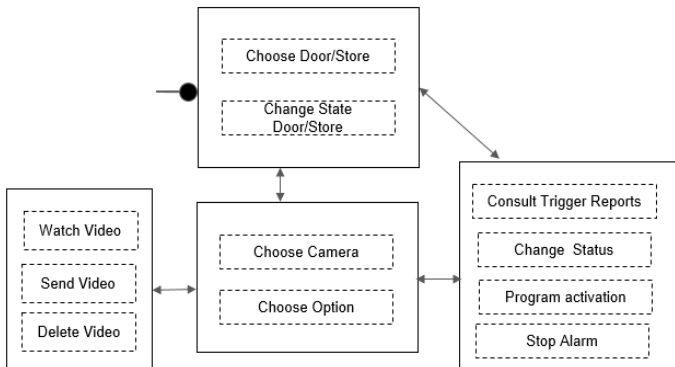


Figure 6. AUI\_3

### 3.4. Participants

We recruit a total of 28 participants: PhD students, recent graduated doctors, master students, engineers and other students in computer science.

## 4. EXPERIMENT

We first start by briefly introducing the study. We then explain the security house application and its main functionalities. Finally, we give each participant two designs and a questionnaire. We do not enforce any time limit; the participants take their time to observe each UI.

The questionnaire given to each participant within the two designs is divided in two parts. The first part is composed of 11 questions based on a 5-point Likert scale. This first part is meant to rate the design. These questions are about three main aspects: Content (organization), navigation (structure/navigation tool) and design (visual). In the second part of the evaluation, the participants are asked to give their opinions concerning these six dimensions of design evaluation: navigation, aesthetics, readability, consistency, exportability and learnability.

We divide the participants into 3 groups. The first group is asked to evaluate UI\_1 with UI\_2. The second group evaluates UI\_1 with UI\_3, and the third group evaluates UI\_1 with UI\_4. The UIs are selected as presented below:

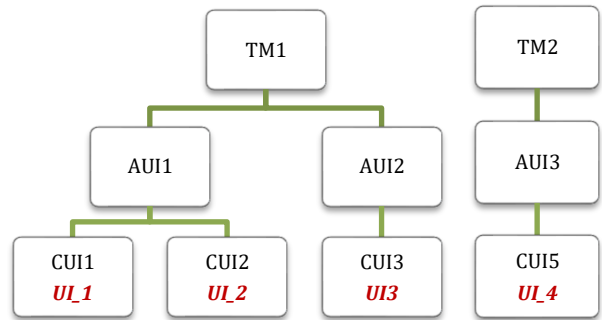


Figure 7. Choice of the alternative design

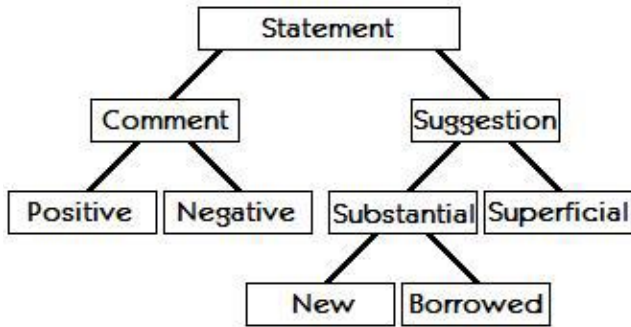
The aim of the experiment is (1) to identify which variant gives more feedback, and (2) to see the impact of each abstraction level variations on the user's feedback, in order to define which alternative design to use during UI evaluation.

## 5. EVALUATION RESULTS

### 5.1. Categorization of user's feedback

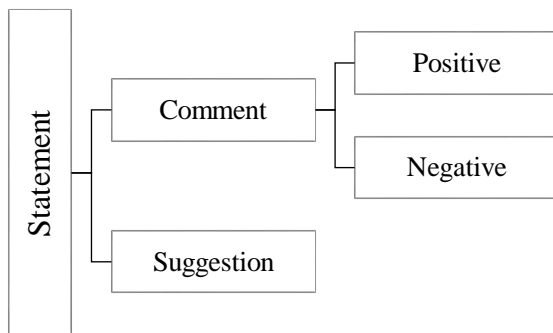
In order to classify the users' statements (critique, opinion and suggestion), we use the taxonomy elaborated in (Tohidi, 2006) but we adapt it to our needs. In their work, Tohidi et al. divided the user's statements as shown in Figure 8 where comments are facts or personal opinions about the design. **Suggestions** are propositions for change to improve

the current design. The comments were either **positive** or **negative**. As for the suggestions there were classified as **substantial** or **superficial**. Substantial suggestions include ideas for improvement that were original (**new**), as well as ideas **borrowed** from other interfaces.



**Figure 8.** Categorization of User Feedback according to Tohidi et al.

In our study, we only consider “comments” and “suggestions”, comments being classified as either “positive” (Easy and convenient navigation), or “negative”, (there is too much information in the interface).



**Figure 9.** Categorization of User Feedback

## 5.2. Impact on user ratings depending on the choice of the alternative design

In order to assess the impact of comparing UI\_1 to different designs (same task model but different AUI; same task model and AUI, but different CUI; different task models), we first calculate the average overall score of UI\_1 based on the participant’s rating given in the questionnaire.

Then, for each different prototype, we compare the score assigned to UI\_1 when seen with each of the other design. Finally, we calculate the number of statements about UI\_1 each time when given with a different UI.

The first observation is that the choice of the second UI conditions the user returns and opinion to the first one. For example, when comparing UI\_1 to UI\_2, a user did not comment about the interface navigation, but when comparing UI\_1 to UI\_3, the user started criticizing the navigation or the widgets.

The number of statements when comparing UI\_1 to UI\_3 was higher than when comparing UI\_1 to UI\_2, and UI\_4. Also, the average score given to UI\_1 when seen with a design with a different AUI was lower than when seen with the other designs (Table 1).

These results support our hypotheses: (1) the comparative evaluation is more productive with a UI that has the same task model but a different abstract UI; (2) the number of suggestions to improve is significantly higher when comparing UI\_1 to UI\_3 than when comparing UI\_1 to UI\_2 or UI\_4.

An observation that we did not expect is that when evaluating the UI with one design that has a different Task model, the number of positive feedback is significantly lower.

Evaluation / Statement type	UI_1 seen with UI_2	UI_1 seen with UI_3	UI_1 seen with UI_4
<b>Number of statement</b>	18	<b>29</b>	17
<b>Positive comments</b>	5	<b>4</b>	2
<b>Negative comments</b>	4	<b>14</b>	8
<b>Suggestions</b>	9	<b>11</b>	7
<b>Average score</b>	0.68	<b>0.58</b>	0.63

**Table 1.** Impact on user ratings depending on the choice of the alternative design

## 6. CONCLUSION AND FUTURE WORK

As demonstrated in (Tohidi, 2006), user’s feedback is affected by the number of design alternatives they are exposed to. In this paper, we report a study to observe the user’s feedback depending on the alternative UI given for comparison. The aim of this research is to define the criteria that the alternative UI must meet to maximise returns on the original one. We used the CRF to characterise the UIs variations.

We observe that the user's opinion about the UI under study is remarkably affected by the choice of the alternative design presented. Analyzing the feedback, we found that the AUI variants affect the users' feedback the most in terms of rating (score) and of the number of statements.

In the next step, we will explore further criteria, and once well defined, we will develop a tool for generating the *best* alternative UI for supporting comparative evaluation at low-cost and for high-benefit.

## 7. REFERENCES

- A.Xu, Brian P. Bailey.2012. What Do You Think? A Case Study of Benefit, Expectation,and Interaction in a Large Online Critique Community. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*.295-304
- B.Lee, Savil Srivastava, Ranjitha Kumar, Ronen Brafman, Scott R Klemmer.2010. Designing with Interactive Example Galleries. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. (CHI '10). 2257-2266
- ConcurTaskTrees. Retrieved April 2017 from <https://en.wikipedia.org/wiki/ConcurTaskTrees>.
- Dicks, R. S.2002. Mis-Usability: On the Uses and Misuses of Usability Testing. In *Proceedings of the 20th annual international conference on Computer documentation*.
- D.Masson, Alexandre Demeure, Gaelle Calvary.2011. Examples Galleries Generated by Interactive Genetic Algorithms .In *proceedings of the Second Conference on Creativity and Innovation in Design*. 61-71
- D.T. Nguyen, Thomas Gancarz, Felicia Ng, Laura A. Dabbish, Steven P. Dow.2017. Fruitful Feedback:Positive Affective Language and Source Anonymity Improve Critique Reception and Work. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*.1024-1034
- G.Calvary Joëlle Coutaz, David Thevenin, Q. Limbourg, L. Bouillon, Jean Vanderdonckt.2003. A Unifying Reference Framework for multi-target user interfaces.*Interacting With Computers* Vol. 15/3 . 289-308.
- G.John, S. Design Prototypes.1990. A Knowledge Representation Schema for Design.*AI Magazine: Volume 11 Issue 4, Winter 1990*.
- G.Meixner, Gaëlle Calvary, Joëlle Coutaz Introduction to Model-Based User Interfaces Retrieved January 2014 from <https://www.w3.org/TR/mbui-intro/>.
- S.R. Herring, C.C. Chang, J. Krantzler, and B.P.Bailey .2009. Getting inspired!: understanding how and why examples are used in creative design practice. In *Process of the 27th international conference on Human factors in computing systems*. (ACM 2009). 87-96.
- J.Manuel, Juan M. González Calleros,Gerrit Meixner, Fabio Paternò, Jaroslav Pullmann, Dave Raggett, Jean Vanderdonckt. Model-Based UI XG Final Report. Retrieved on April 2010 from <https://www.w3.org/2007/uwa/editors-drafts/mbui/Model-Based-UI-XG-FinalReport.html>.
- J.Hui, Amos Glenn, Rachel Jue, Elizabeth Gerber, Steven Dow. 2015. Using Anonymity and Communal Efforts to Improve Quality of Crowdsourced Feedback.In *Proceeding of Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP-15)*
- K.Luther, Amy Pavel, Wei Wu, Jari-lee Tolentino, Maneesh Agrawala, Björn Hartmann, Steven Dow 2014. CrowdCrit: Crowdsourcing and Aggregating Visual Design Critique. In *Proceedings of the companion publication of the 17th ACM conference on Computer supported cooperative work & social computing*.21-24
- M.Tohidi, William Buxton, Ronald Baecker, Abigail Sellen. 2006. Getting the Right Design and the Design Right:Testing Many Is Better Than One. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'2006)*.1243-1252
- M.Tohidi, William Buxton, Ronald Baecker, Abigail Sellen. 2006. User Sketches: A Quick, Inexpensive, and Effective way to Elicit More Reflective User Feedback. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles (NordiCHI'2006)*.105-114
- N.Hili, Y.Laurillau, S.Dupuy-Chessa, G.Calvary. 2015. Innovative Key Features for Mastering Model Complexity: FlexiLab, a Multimodel Editor Illustrated on Task Modeling. *Proceedings of the 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*.
- M. E. Wiklund, Christopher Thurrott, Joseph S. Dumas. 1992. Does the Fidelity of Software Prototypes Affect the Perception of Usability? In *proceedings of. Human Factors Society 36th Annual Meeting, (1992)*, 399-403.