

Elsevier Editorial System(tm) for Information Processing & Management
Manuscript Draft

Manuscript Number: IPM976R1

Title: Social and Q&A interfaces for app download

Article Type: Full Length Article

Section/Category: Regular Issue

Keywords: Web interfaces, software download, usability, Q&A, reputation system, social web

Corresponding Author: Dr. Michele Zanda, Ph.D.

Corresponding Author's Institution: University of Pisa

First Author: Michele Zanda, Ph.D.

Order of Authors: Michele Zanda, Ph.D.; Gianluca Dini, Associate Professor; Pierfrancesco Foglia, Researcher; Cosimo A Prete, Full professor

Abstract: Downloading software via Web is a major solution for publishers to deliver their software products. In this context, user interfaces for software downloading play a key role. Actually, they have to allow usable interactions as well as support users in taking conscious and coherent decisions about whether to accept to download a software product or not. This paper presents different design alternatives for software download interfaces, i.e. the interface that prompts the user if he wishes to actually complete its download, and evaluates their ability to improve the quality of user interactions while reducing errors in user decisions. More precisely, we compare Authenticode, the leading software download interface for Internet Explorer, to Question-&-Answer, a software download interface previously proposed by the authors [Dini et al., 2007]. Furthermore, we evaluate the effect of extending both interfaces by means of a reputation system similar to the eBay Feedback Forum. The results of the usability studies show that i) the pure Question-&-Answer interface is the most effective in minimizing users incoherent behaviors, and ii) the differences in reputation rankings significantly influence users. Overall results suggest guidelines to design the best interface depending on the context (brand reputation and product features).

Q&A-and social-based interfaces for software download

G. DINI, P. FOGLIA, C. A. PRETE, M. ZANDA

*Dipartimento di Ingegneria dell'Informazione
Università di Pisa, Italy*

Email: [g.dini | foglia | prete | zanda]@iet.unipi.it

Abstract. Downloading software via Web is a major solution for publishers to deliver their software products. In this context, user interfaces for software downloading play a key role. Actually, they have to allow usable interactions as well as support users in taking conscious and coherent decisions about whether to accept to download a software product or not. This paper presents different design alternatives for software download interfaces, i.e. the interface that prompts the user if he wishes to actually complete its download, and evaluates their ability to improve the quality of user interactions while reducing errors in user decisions. More precisely, we compare *Authenticode*, the leading software download interface for Internet Explorer, to *Question-&-Answer*, a *software download* interface previously proposed by the authors [Dini et al., 2007]. Furthermore, we evaluate the effect of extending both interfaces by means of a reputation system similar to the eBay Feedback Forum. The results of the usability studies show that i) the pure Question-&-Answer interface is the most effective in minimizing users incoherent behaviors, and ii) the differences in reputation rankings significantly influence users. Overall results suggest guidelines to design the best interface depending on the context (brand reputation and product features).

Keywords: Web interfaces, software download, usability, Q&A, reputation system, social web.

1. INTRODUCTION

User interfaces for software download on the Internet are the front end of online software publishers for remote users. One important component of such interfaces is the *software download interface*, namely the interface that prompts the user if he wishes to actually complete its download, after that he/she has browsed on a web site and selected the desired product to download. The design of such an interface is critical, because it usually shows product related information that should guide the user in its final choice. Software download interfaces are critical for users who may take unconscious or misguided decisions if the interfaces have not been properly designed [Allen et al., 2006; Maxon & Reeder, 2005]. Software download interfaces are critical for publishers too, if the interfaces reduce user trust and satisfaction and system usability at large [Flavian et al., 2006; Hoffman et al., 2006; Shin, 2010]. Finally, software downloading has security implications too. Brustoloni & Villamarin have shown that a poorly designed software download interface may cause *unjustified risks* in terms of breaches in a per-user derived policy adopted to classify risks [Brustoloni & Villamarin, 2007].

The Security Warning Dialog Box in the Microsoft Authenticode interface is a reference, largely-used software download interface for Internet Explorer. Prior works have highlighted weaknesses of such an interface [Dini et al., 2006], [Cranor2007], as well as the attitude of the user to seldom act appropriately when presented with such a download interface [Wood2010]. In particular, in line with [Brustoloni & Villamarin 2007], *incoherent behaviors* in users interacting with the Microsoft Authenticode interface were observed in [Dini et al., 2006], where an *incoherent behavior* consists in a download decision that is incoherent with the motivation given a-posteriori. According to [Shin 2010], the rate of online visitors who buy a product or download software is influenced, among other factors, by trust and attitude, with trust also depending on user interface design quality

[Flavian et al., 2006; Hoffman et al., 2006]. As a consequence, software download interface should have a good design that minimizes *incoherent behaviors*, increases user trust and, then, supports the user intention to purchase.

In prior works we have proposed *Question-&-Answer* (Q&A) a software download interface aimed at reducing incoherent behaviors by asking the user questions about specific product download issues, e.g., the product cost [Dini et al., 2007]. Furthermore, we have suggested the use of a *Reputation System* (RS)—a service like the eBay Feedback Forum—to increase the user trust on a given software product [Dini et al., 2013].

In this paper we experimentally compare the Authenticode and Q&A software download interfaces and evaluate the effects of their extension by means of a reputation system. The overall result is that any solution that increases users attention to the interface contents also significantly reduces incoherent behaviors. Nonetheless, the three design choices are not equivalent as they make users focus on different aspects of the proposed products. For example, RS make users to focus on the aggregated social feedback from previous users whereas Q&A makes users to focus on specific software download issue, such as the *cost* for example. Even if the minimization of incoherent behaviors through a Q&A-enhanced interface has positive effects on users trust, it causes a strong reduction in all software downloads because users tend only to accept free software, refusing software that entails a charge. So, if having a large market penetration is needed, publishers should not adopt a Q&A interface if the software offered entails a charge. However, embedding a reputation mechanism side by side to Q&A dialog reduces the relevance of cost, and reduces incoherent behaviors at an intermediate level. Showing aggregated feedback from previous users has a mitigating role between the common interface and the Q&A interface. From these findings we indicate how designers can better tackle the design tradeoffs in these user interfaces.

The paper is structured as follows. Section 2 presents the background of the problem, and shows how the macro-hypotheses under test have been developed. Sections 3 and 4 show the graphical interfaces that have been tested in the experiments and present the experimental settings. Then, the experimental results are given with a concluding discussion on the observed trends.

2. BACKGROUND AND HYPOTHESES

2.1. ISSUES IN ONLINE SOFTWARE DELIVERY

Online software delivery, like common e-commerce, must solve concerns especially regarding authentication and trust, in order to make users confident that they are not being cheated [CommerceNet, 2000; Corritore et al., 2003]. Complex Web interfaces and fraudulent software houses have caused serious problems such as spreading dialers, spywares and other threats [Shukla & Nah, 2005].

In this environment users cannot relate with a merchant and understand whether the software is in line with their expectations. In an ordinary shop, clients relate with a merchant and establish a trust relationship, and a similar relationship should be established also online [Corritore et al., 2003; Cheskin, 1999]. Easy-to-use interfaces and complete information are highly correlated to user trust [Flavian et al., 2006; Hoffman et al., 2006; Nilsson et al., 2005; Laberge & Caird, 2000; Lanford, 2006], and a trustworthy interface is more likely to make users trust a vendor on the Internet [Fogg et al., 2001].

A trustworthy, usable interface would not solve two relevant questions for users dealing with software download from the Internet: first, the need to know who really published the file offered, second, supporting the user in understanding what to expect from the executable file. As a solution to the first problem (authentication), major software vendors have developed frameworks for code signing [Jansen, 2000; Thawte, 2007], such as the Microsoft Authenticode that has been taken as

reference in the current study due to its widespread diffusion on the Internet: 53% of the market share [BMS, 2012]. By signing code, publishers seek to build a relationship of trust with users, satisfying the matter of accountability at the same time. Software publishers sign the piece of software they are releasing. The publisher's certificate, the certification authority certificate, and the signed code are then packaged together.

At the client side, the browser verifies the publisher's signature on the code, and, if verifications are successful, the Security Warning dialog box (SWDB) is presented to the user (Fig. 1). If some problems arise, the user is notified with a different dialog box. The SWDB presents the certification authority name, the software publisher name and the software name, as well as other details such as the cost and other optional information pieces. This information is shared by means of the different Authenticode versions that have been released over the years.

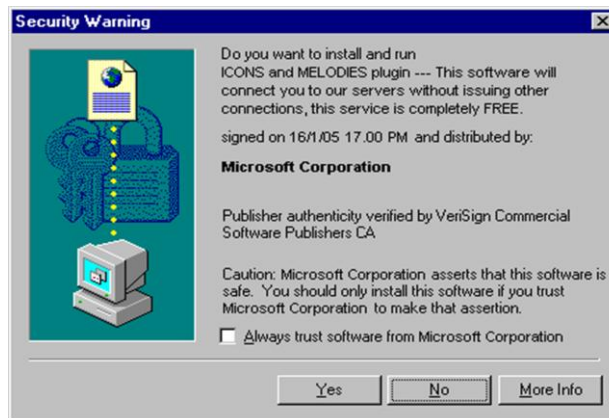


Figure 1. The Security Warning dialog box (SWDB).

The further issue to be faced in software downloading concerns the user's expectations of the file. The Authenticode framework, like other code signing frameworks, ties a publisher to a file via its certification authority, but it does not help the user understand what to expect from the file. Predictability in this scenario would be important as it influences trust [Corritore et al., 2003]. Current software download frameworks do not give any safe information about what and how the executable file will behave once run on a computer system. Given that predictability cannot be fully ensured with the above approaches, in recent years Reputation Systems (RS) have emerged as a method for fostering trust amongst strangers cooperating online by gathering, distributing, and aggregating the feedback of previous consumers [Resnick et al., 2000].

2.2. INTERACTION PARADIGMS FOR SOFTWARE DOWNLOAD

Deciding whether to accept or refuse a software download is a context-dependent decision: "In such situations, an application usually needs user input, because the application cannot determine automatically all the context relevant to the decision" [Brustoloni & Villamarin, 2007]. For instance, receiving a software should be allowed if the sender is a friend, and should be denied if the sender is a malicious user. Based on [Brustoloni & Villamarin, 2007] classification of interaction paradigms that support such decisions, in this paper we analyze the *warn-and-continue* and the *context sensitive guidance*. A context sensitive guidance interface asks the user to provide context information necessary for a security decision, while in the *warn-and-continue* approach the application warns the user of the risk and asks whether the user wants to accept it or not. Interfaces for software download (e.g.: the Authenticode Security Warning Dialog Box) follow the *warn-and-continue* approach.

In [Dini et al., 2006] users claimed that they wished to download free software only, but they largely accepted software from Well-Known (WK) publishers, showing that the brand effect was more important than cost. Thus, users did not focus their attention on the whole interface, but they

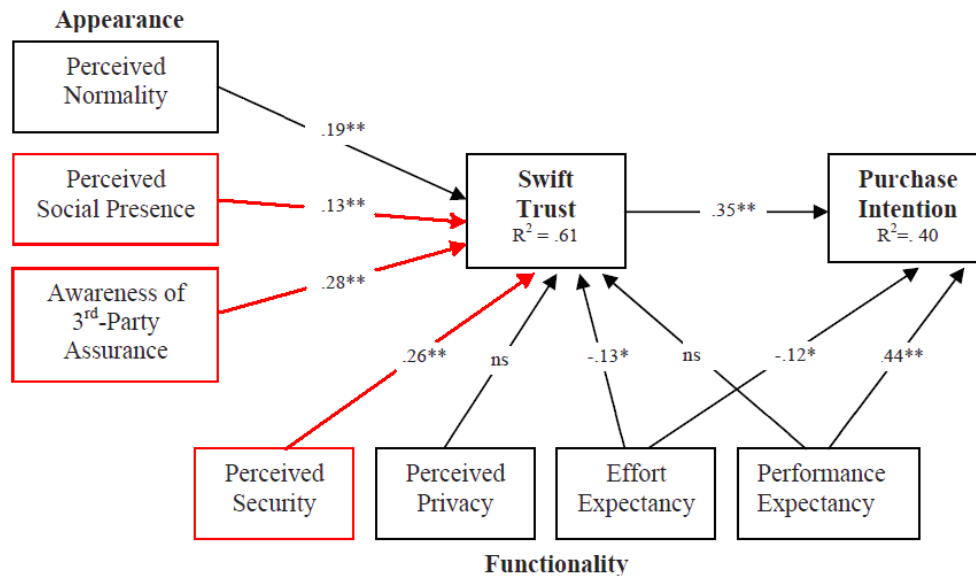
accepted or refused software downloads according to the publisher name, not the item cost. In addition, the Authenticode Security Warning Dialog Box prevented Common Name (CN) publishers with little or null brand name from having their software downloaded. Similar considerations were done by [Brustoloni & Villamarin, 2007], [Wood2010] highlighting the problems deriving from the *warn-and-continue* approach: users click “continue” automatically, click through warnings due to habit [Chia et al., 2010], do not pay attention to the whole information in the interface, and run into *incoherent behaviors* [Dini et al., 2007] or *unjustified risks* (breaches in the policy adopted to classify risks) [Brustoloni & Villamarin, 2007]. As a countermeasure to these problems, [Brustoloni & Villamarin 2007] propose the adoption of polymorphic dialogs that change the form of the required user inputs, forcing users to pay attention to security decisions. While [Dini et al. 2007] tested a Question-&-Answer (Q&A) interface designed to increase user attention on the contents of the interface. Both the designs proved effective in reducing *unjustified risks* or *incoherent behaviors* compared to other interaction paradigms.

2.3. TRUST AND REPUTATION SYSTEMS

Trust for [Mayer et al. 1995] is the willingness of a trustor to be vulnerable to the actions of a trustee based on the expectation that the trustee will perform a particular action important to the trustor, irrespective of the ability to monitor or control the trustee. Trust is driven by many variables, and upon deciding whether to accept or refuse a software download, the user should be able to predict if the software meets with his/her expectations. Predictability influences trust: “predictability is a trustor’s expectation that an object of trust will act consistently based on past experience” [Corritore et al., 2003]. To better understand and solve trust issues, [Egger 2000] proposed a model of trust for electronic commerce, which includes interface properties such as usability and familiarity, and cooperation between consumers. Cooperation is relevant also for [Olson 2000], who states that trust deals with behavior: “people learn to trust others by noting their behavior”. In the online domain, the end user has limited possibilities to fully try a software product or a service before buying, while the software publisher knows what he gets as long as he receives money. “The inefficiencies resulting from this information asymmetry can be mitigated through trust and reputation” [Josang et al., 2007]. Reputation Systems help end users in predicting whether a product can be trusted or not. Enforcing a trust belief in the end users towards a remote software provider is a subjective phenomenon influenced by a set of factors, and in absence of personal experience, trust must be based on third party reviews. For [Josang 2007], reputation is a collective measure of trustworthiness based on reviews from members in a community.

2.4. FOLLOWING PROVEN RESEARCH MODELS

[Lin et al. 2009] validate a research model wherein purchase intention and swift trust are influenced by a set of parameters, including perceived social presence, awareness of 3rd party assurance, and perceived security (Fig. 2). Swift trust refers to trust formed quickly in new or transitory relationships [Meyerson et al., 1996]. According to Lin, “social presence refers to consumers’ perception that there is personal, sociable, and sensitive human contact and/or peer community on the Web site”. This type of reassurance for online entities is similar to the reassurance provided by brick and mortar businesses [Gefen & Straub, 2004]. The model by Lin et al. infers users opinions and intentions by looking at the answers from a survey. The users answers are analyzed to derive how “Purchase intention” is influenced by a set of factors. The problem with survey-only studies is that relevant aspects of the user interaction are not differentiated qualitatively. In an everyday Internet scenario users can make incoherent decisions, which happen to differ significantly from the desired behaviors. For instance, it is clear that asking a user “Do you feel safe to provide your login and password to a phishing Website?” will cause a negative answer. Though in real life the user could behave differently because of badly designed interactions, fraudulent designs, lack of attention.



** $p < 0.01$; * $p < 0.05$; ns: non-significant

Figure 2. Model of swift trust in web vendor [Lin et al., 2009].

The results of this paper help to understand how interfaces can be better designed to meet the users needs, in a safe way. Thus, our study is orthogonal to the above research model: we investigate what happens in some building blocks of the above model, when real interfaces must be designed. The block “Social presence” suggests the adoption of a Reputation System as aggregator of reputation scores from a community, also to improve the efficiency in the process of software downloading. [Resnick et al. 2000] state that these mechanisms can help people make decisions about who to trust and provide incentive for honest behavior, while deterring dishonest parties from participating. Despite many problems in theory [Resnick et al., 2000], in practice RSs proved to work rather well as a means to provide relevant information on the quality of merchants and products in auction sites. Successful examples are represented by iTunes and its reputation system for both desktop and mobile users, by the eBay auction site, or online book sellers like Amazon [Chia et al., 2010; Dellarocas & Resnick, 2003; Chen & Liu, 2011]. RSs have been proposed as a means to identify inauthentic content in file sharing applications [Walsh & Sirer, 2006], and similarly they could be effective in helping users to identify deceptive software publishers.

The model by Lin also includes “Perceived security” which, as shown in Section 2.1, can be managed by designing a *context sensitive* interface like the Q&A. The Q&A proved effective in minimizing incoherent decisions, thus increasing perceived security in the interaction paradigm, as also observed in questionnaire answers [Dini et al., 2007]. In our design, the Q&A interface asks for an explicit feedback on cost, and this results in minimizing the incoherent behaviors, but also in focusing users attention on cost.

The Reputation System makes users focus on aggregated scores from previous users, and by doing so it also increases users overall attention in the interface. With the RS, the answer given in the Q&A interface becomes one of the possible driving factors together with the scores shown in the RS box. This is tested with the hypothesis **H1**:

H1: the RS reduces the relevance of cost aspects when compared to the Q&A interface.

The RS does not ask for an explicit feedback in order to let the user go further with the download, so it should be expected an increase in incoherent behaviors when compared with a Q&A interaction (hypothesis **H2**).

H2: the Q&A interface minimizes incoherent behaviors when compared to the other interfaces.

From these considerations, combining the Q&A with the RS should increase incoherent behaviors when compared to plain Q&A (hypothesis **H3**), and reduce them when compared with the common Authenticode interface (hypothesis **H4**).

H3: the RS with Q&A increases incoherent behaviors when compared to the Q&A.

H4: the RS with Q&A reduces incoherent behaviors when compared to the SWDB.

And we also test **H5**, **H6** and **H7**:

H5: the RS with Q&A reduces the relevance of cost aspects when compared to the Q&A interface.

H6: the RS with Q&A increases the relevance of trust when compared to the Q&A interface.

H7: the RS with Q&A increases the relevance of trust when compared to the SWDB interface.

3. GRAPHICAL DESIGNS

The Question-&-Answer interface (Q&A) [Dini et al., 2007] utilized in our study is a wizard with three steps. The first step replicates the information of the common SWDB, while the second step asks the cost of the software under download, and the third step allows to download the software if the given answer is correct (Fig. 3). The reason why we choose to focus on the cost information is the following. In a previous work [Dini et al., 2007] we found incoherent behaviors of users with respect to cost, namely users stated they wanted to download free software whereas in practice they actually downloaded expensive one (or vice versa). We also proved that the Q&A interface mitigates such a behavior. Therefore, in this work, we decided to start from such consolidated results. Conceptually, the Q&A interface may present questions on any other information that is considered relevant, such as name of the producers, name of the software, et cetera.

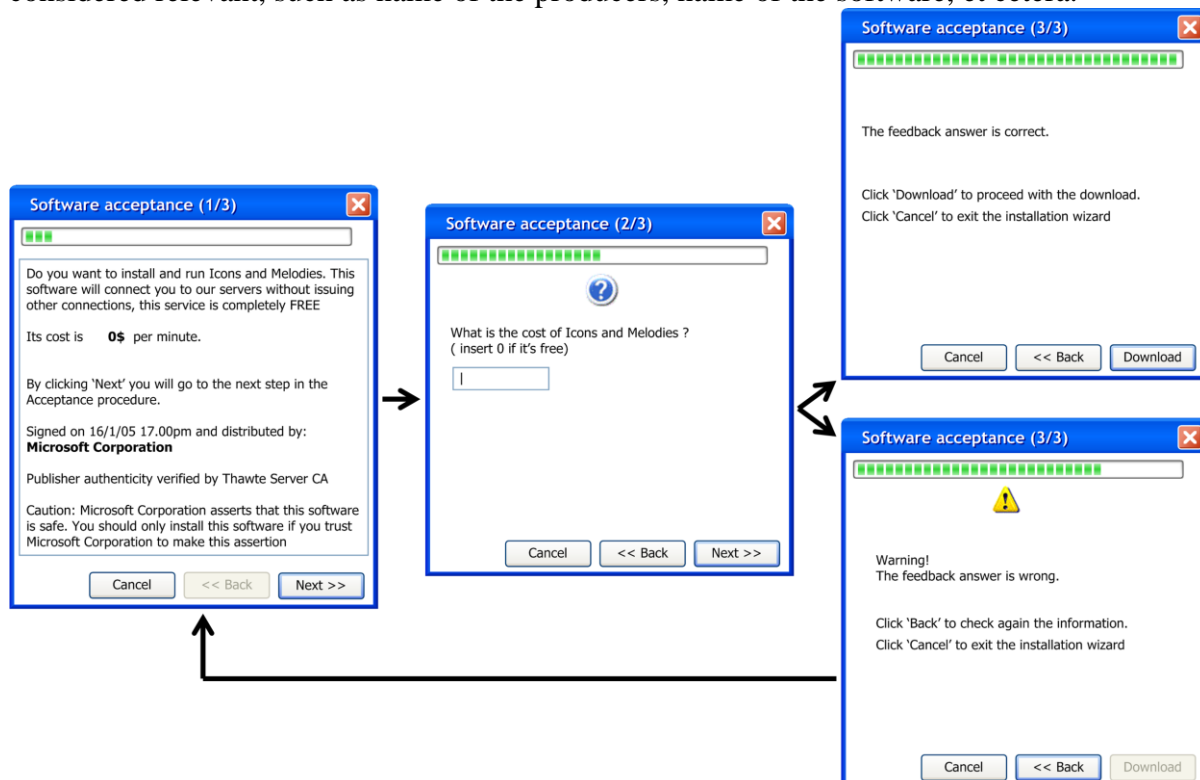


Figure 3. The Question & Answer download interface with the 3 steps wizard.

Figure 4 shows the other interfaces used in our experiment. They are, respectively, the Security Warning Dialog Box (left) and the Q&A interface (right), both augmented by a reputation systems which display the aggregation of users rankings regarding the software to be downloaded [Dini et al., 2013]. As shown in Fig. 4, the RS is placed at the bottom of the dialog box, showing the total number of feedback reviews, as well as the rate of positive, neutral, and negative reviews. According to the classification of RS ranking aggregation mechanisms proposed by [Josang et al. 2007], the interface, like that of the eBay Feedback Forum, is a simple summation system. In a simple summation RS, the system accumulates all given ratings to get the overall reputation. The RS is simple and readable at a glance, providing a concise view of the distribution of users' feedback [Dini et al., 2013].

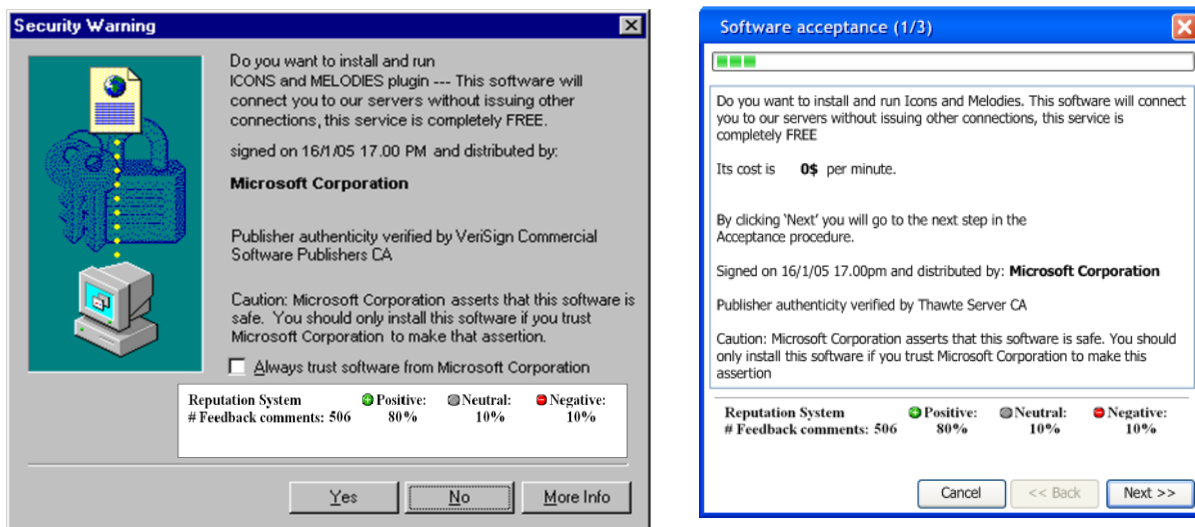


Figure 4. Reputation System in Security Warning dialog box (SWDB) on the left, and in the first step of the Question-&-Answer (Q&A) interface on the right.

4. METHOD

4.1. POPULATING THE REPUTATION SYSTEM

The Reputation System was inspired by the eBay Feedback Forum also described by [Resnick et al. 2000], with the purpose to influence user trust beliefs about software publishers while dealing with software download. The RS effect is assessed by looking at the software download acceptances rates and by considering the motivations given for each acceptance or refusal. The RS must be populated meaningfully, in order to simulate an every day scenario for all three classes of publishers tested in the experiment. Looking at the eBay Feedback Forum, often well-known and established merchants have a high and positive reputation ranking, unless they decide to have no ranking at all as their brand effect is strong enough. On the other hand, deceptive merchants who may provide an RS present few comments mostly negative. Finally, common name merchants tend to support an RS as they want to achieve visibility and, contextually, be trusted by end users. Thus, in our experiments we do not test patterns in the population of the RS, but instead we choose to assume that the RS is already populated in line with the above observations. In particular: well-known publishers have a *high and positive* RS, or *do not support it*; deceptive publishers have a *low and negative* RS, or *do not support it*; publishers with a common name have either *high and positive* or *low and negative* RS ranking. The reason is twofold: reducing the number of test cases for each user, and simulating real life scenarios by associating appropriate reputation ranks to the three classes of publishers, while optimizing the time to complete the experiment.

In detail, the rankings present in the experiments are shown in Table 1: a high ranking means that the number of reviews is greater than 400, with 80% positives, 10% neutrals, 10% negatives; a low ranking means that the number of feedback is smaller than 40, with 30% positives, 30% neutrals, 40% negatives; and finally the publisher can decide to not support the RS.

| RS Ranking | Number of reviews | Positive | Neutral | Negative |
|-------------------|-------------------|----------|---------|----------|
| High and positive | > 400 | 80% | 10% | 10% |
| Low and negative | < 40 | 30% | 30% | 40% |
| Not Supported | - | - | - | - |

Table 1. The three rankings associated to the software publishers in the usability test.

4.2. TEST PROCEDURE

The study included the following independent variables.

- Software publisher name. Software publishers were varied according to three classes: (i) well-known software publishers (WK); (ii) common name publishers (CN), and (iii) companies whose name was deceptive (DN).
- Software cost. It was varied between free and charge entailed (above 20 €).
- Reputation System ranking. The RS was varied with the following scheme:
 - o High and positive rankings for WK publishers and CN publishers;
 - o Low and negative rankings for CN publishers and DN publishers;
 - o Not supported: the publisher decides not to support a Reputation System, for WK publishers and DN publishers.
- Interface for software downloading. Participants interacted with four interfaces to perform differential analyses: the original SWDB, the SWDB enriched with the RS, the 3SW enriched with the RS and the plain 3SW.

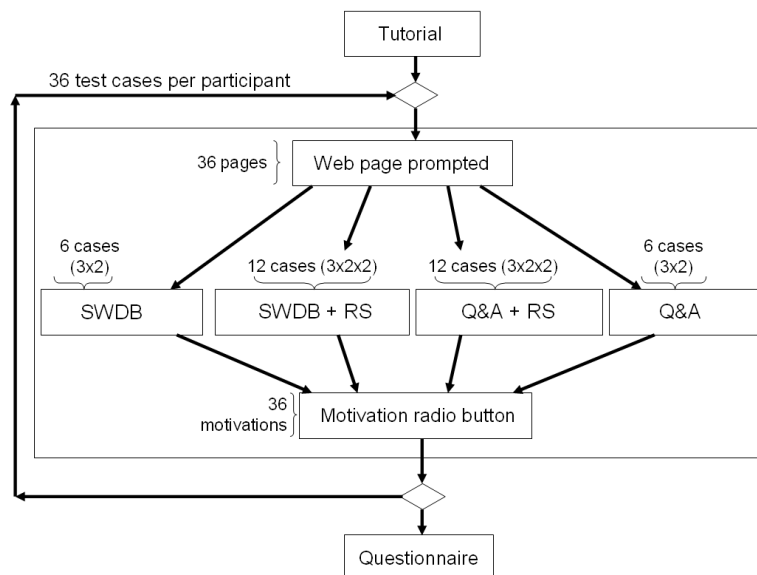


Figure 5. Test procedure.

Each participant was shown a complete combination of the independent variables. More in details, each participant completed 36 test cases (Fig. 5), obtained from combining the independent

variables with: (i) the SWDB interface, publisher name (3 values), and cost details (2 values); (ii) the SWDB + RS interface, publisher name, cost details, and RS ranking (2 values); (iii) the Q&A + RS interface, publisher name, cost details, and RS ranking; and finally, (iv) the Q&A interface, publisher name and cost details. In particular, we defined a pool of 36 pages divided in 4 groups as described in Figure 5. Such pages are then presented each participant in a random order.

| Decision | Motivation | Label | Incoherent, if in combination with |
|------------|--|-------|------------------------------------|
| Acceptance | I was very interested in the product | I | |
| | I was not very interested, but I trust the code publisher | T | a deceptive publisher name |
| | I was not very interested, but the cost was fine | O | a free code |
| | I was not very interested, but it was free | F | a code entailing a charge |
| | I didn't want to accept it | O | (explicit incoherent behav.) |
| | Other motivations | O | |
| Refusal | I was not interested in it at all | NI | |
| | I could be interested, but I didn't trust the code publisher | DT | |
| | I could be interested, but it was expensive | C | a free code |
| | I could be interested, but even if it was free, I refused | O | a code entailing a charge |
| | I didn't want to refuse it | O | (explicit incoherent behavior) |
| | Other motivations | O | |

Table 2. Motivation options to be chosen in each test case, and related incoherent motivations. Labels identify single motivations, those labeled with O were chosen in minor quantity by participants.

The test consists of a sequence of test cases, where each test case consists of three stages. In the first stage we display a web page that proposes a software download. In the second stage, the participant decides whether to accept or not the software product by using one of the displayed interfaces. In the third stage, the participant is required to motivate his/her download choice with a motivation radio-button. The given motivations, together with the acceptance or refusal of a piece of software, are dependent variables in our experiments. The motivation radio-button presents a set of options for participants to choose. The content of this window is described in Table 2 (column Motivation). If the participant accepts to download, then he/she is presented a list of motivations mainly related to interest (I), trust (T), free cost (F) and other minor motivations (O). In contrast, if the participant refuses to download, he/she is shown another list of motivations related to no interest (NI), distrust (DT), high cost (C) and other minor ones (O).

At the end of the 36 test cases, participants answered a multiple-choice questionnaire containing personal questions and questions about the used interface. The full questionnaire is reported in Appendix B. , As each participant had to complete 36 test cases, we minimized learning effects by not giving participants feedback on their actions during the experiment.

In a previous work [Dini et al., 2006], it was observed that user intentions and proper behaviors for each single test case could not be derived by just looking at the final questionnaire answers, as each test case needed a precise motivation. For this reason, we let participants choose the most suitable motivation out of a set (see Table 2), after either accepting or refusing a downloading. By doing this, participants can motivate their decisions with precision. In addition, we used motivations to clearly identify participants' coherent and incoherent behaviors. In fact, in the cited previous works, it was observed that users interacted with interfaces incoherently at times: a participant behaves incoherently if his/her motivation does not match with the features of a piece of software downloaded or refused. In Table 2, column "Incoherent, if in combination with" shows when

motivations not matching actual software features identify incoherent behaviors. An example of an incoherent behavior is if a participant refuses a download picking the motivation “I could be interested, but it was expensive”, while the software is free.

At the beginning of a test session each participant was asked to “either accept or refuse the proposed software as if you were using your own computer”, since we want to simulate the scenario of a typical Web user who browses the Internet and downloads software according to its own interests. Participants were not rewarded for their time, but they were told: “you are participating to the design of a new user interface for e-government web applications”. A gender balanced group of 40 participants was recruited, with ages ranging from 18 to 40 (mean age 23). 34 participants were high school students or undergraduate students, while 6 participants were employed in public administration. Participants attended the tests in person in our labs or in their office (rather than remotely via web). The recruited group had good Internet experience, as shown in Table 3. All the experiment were performed on a windows based platform.

| Question | Answers | | | |
|---|---------------|------------------|--------------------|----------------|
| How often do you use the Internet? | Rarely (-) | Once a month (-) | Once a week (5) | Every day (35) |
| Why do you use the Internet? | Work (5) | Study (27) | Entertainment (8) | Other (-) |
| Do you usually download software? | Never (-) | At times (9) | Frequently (31) | |
| As an Internet user, have you ever seen the “Security Warning” dialog box? | Yes (40) | No (-) | Don’t remember (-) | |
| How do you usually behave when confronted with a “Security Warning” dialog box? | Click Yes (8) | Click No (4) | It depends (28) | |
| Do you usually pay attention to what is written in the “Security Warning” dialog box? | Yes (15) | No (11) | At times (14) | |

Table 3. Sample characteristics.

All participants actions were logged on a database for subsequent analyses. Results presented in the paper are given reporting raw percentages, or Chi-Square (χ^2) statistics. Chi-square was adopted to compare groups with different cardinalities, and we also considered the Yates correction for one degree of freedom. It provides a p-value that indicates the probability that the observed difference is due to chance, where $p < 0.05$ is commonly accepted to indicate a statistically significant difference in the sample [Glantz, 2005].

4.3. VALIDITY

We minimized learning effects by not giving the users any feedback on each single test cases (e.g., users performing free acceptances or acceptances entailing a charge without highlighting, during the test, incoherent motivations). We observed that the actions taken by the users with tested interfaces were very conservative: they tried to minimize possible risks (even if they were aware that the experiment could not harm their terminal). Thus, we believe that the test environment had little impact on our results. In addition, we did not observe a lack of motivation in the participants, also because after initial tests, we minimized the duration of a single test at less than 30 minutes on average.

There were three ranking categories shown in the Reputation System in our test: high and positive rankings, low and negative rankings, and rankings not supported. Although we observed that rankings tend to be classified mainly in line with these three categories, other studies should investigate how users behave while dealing with other reputation patterns.

5. RESULTS

In this Section we verify the hypotheses under test specified in Section 2.4 and give further significant findings. Tables with full results in details are reported in Appendix A.

5.1 TESTED HYPOTHESES

Presenting aggregated social-feedback with an RS influences user decisions even in presence of a Q&A interface asking for cost details: users' focus shifts from cost to rankings given from previous users (**H1**, **H6**). Such a shift of focus can cause an effect that we identify with a soft information overload, since incoherent behaviors are minimized only with the plain Q&A interface (**H2**). The presence of more information provided by the RS in the Q&A interface makes users to focus on aspects not related to cost but also causes an increase in incoherent behaviors (**H3**).

H1: the RS reduces the relevance of cost aspects when compared to the Q&A interface. - **Confirmed** ($X^2(1,40):p<0.05, power>0.8$).

H2: the Q&A minimizes incoherent behaviors when compared to the other interfaces. - **Confirmed** ($X^2(1,40):p<0.05, power>0.8$).

H3: the RS with Q&A increases incoherent behaviors when compared to the Q&A. - **Confirmed** ($X^2(1,40):p<0.05, power>0.8$).

The RS not only shifts users' focus from cost, but it also reduces incoherent behaviors observed in the SWDB (**H4**). "Flow" is a concept that characterizes the human-computer experience as playful and exploratory [Trevino & Webster, 1992]. We believe that the RS presence increases users "flow", making them more active in exploring the information shown in the dialog box, and by doing so they make fewer *incoherent behaviors*.

The RS presence in the Q&A manages to increase the relevance of trust, reducing the focus on cost, when compared with the plain Q&A interface (**H6**). This means that showing a social feedback from previous peer users is considered a valid source of information and influences users.

H4: the RS with Q&A reduces incoherent behaviors when compared to the SWDB. - **Confirmed** ($X^2(1,40):p<0.05, power>0.8$).

H5: the RS with Q&A reduces the relevance of cost aspects when compared to the Q&A interface. - **Not confirmed** ($X^2(1,40):p>0.05, power>0.6$).

H6: the RS with Q&A increases the relevance of trust when compared to the Q&A interface - **Confirmed** ($X^2(1,40):p<0.05, power>0.8$).

H7: the RS with Q&A increases the relevance of trust when compared to the SWDB interface - **Not confirmed** ($X^2(1,40):p>0.05, power>0.8$).

The RS in the Q&A did not show statistically significant differences (**H5** not confirmed) when compared with the plain Q&A even if raw percentages indicate a difference, but looking at the statistical power, the non-significance can be due to the limited number of participants. No statistical significance on trust motivations was observed when the Q&A + RS is compared with the SWDB interface (**H7** not confirmed). This is due to the fact that with the SWDB users gather mainly the publisher name, and base their decisions on the trust that they have in it. Thus, trust motivations with SWDB are high, but they are high because users do not find the relevant information and as a consequence base their decisions on the publisher name.

The macro effect of the RS on motivations (tables 9, 10) is reduced as results aggregate positive, negative or null rankings in the classes of publishers, but different ranking influence downloads acceptances and refusals as shown on Table 6 in Appendix A.

The Q&A has few graphical effects, and asks for explicit feedback on cost, to ensure that the user gathers it. The results show that when users are pushed to think about cost details (as it happens with the Q&A interface) cost becomes the driving factor for software acceptance/refusal. In fact, with the plain Q&A interface users prefer to download free software, and download software entailing a charge only if they are interested in it (Table 4).

| | Interface | | | |
|------------|-----------|-----------|----------|-------|
| | SWDB | SWDB + RS | Q&A + RS | Q&A |
| Motivation | I | I | I | I |
| WK | 58 % | 63 % | 74 % | 100 % |
| CN | 44 % | 55 % | 59 % | 100 % |
| DN | 45 % | 64 % | 61 % | 100 % |

Table 4. Acceptance motivations for software entailing a charge.

5.2 GENERAL TRENDS

We investigated the rates of incoherent behaviors for all the tested interfaces. Minimizing the rate of the observed *incoherent behaviors* can be considered as an indicator of interface quality. Observing the decreasing tendency line of incoherent behaviors in Fig. 6, we observe that the plain Q&A performs better than the other proposed interfaces, ensuring the safest interaction paradigm. An SWDB-based design results more prone to incoherent behaviors, causing higher acceptances of costly pieces of software (often not desired by the users). Adding graphical elements, like the RS, causes a sort of information overload [Farhoomand & Drury, 2002], as incoherent behaviors with the plain Q&A are lower than with the Q&A + RS.

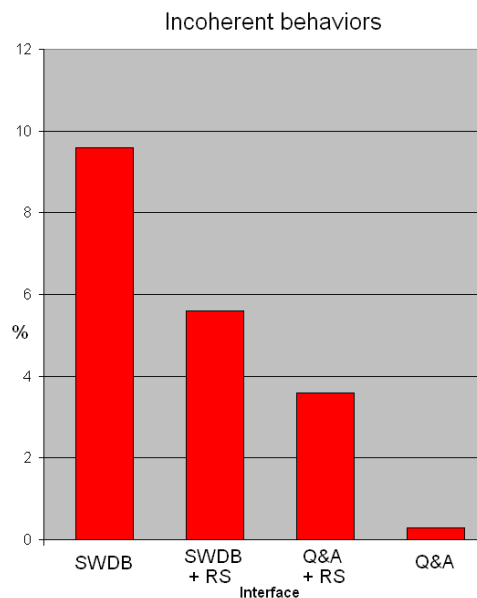


Figure 6. Incoherent behaviors across all tested interfaces, on average.

The analyses of the three novel interfaces (SWDB + RS, Q&A+RS, and Q&A) compared with the SWDB show a global reduction of accepted costly software, in particular concerning software by WK publishers. Fig. 7 shows the acceptance trends of costly pieces of software (data are taken from Tables 5 and 6 in Appendix A). The effect of introducing the Q&A and the RS is clear compared to the basic SWDB: the Q&A reduces significantly the acceptances of costly code, while the RS

reduces the relevance of cost providing higher costly acceptances than the basic Q&A. The Q&A minimizes acceptances for all publishers if software is costly.

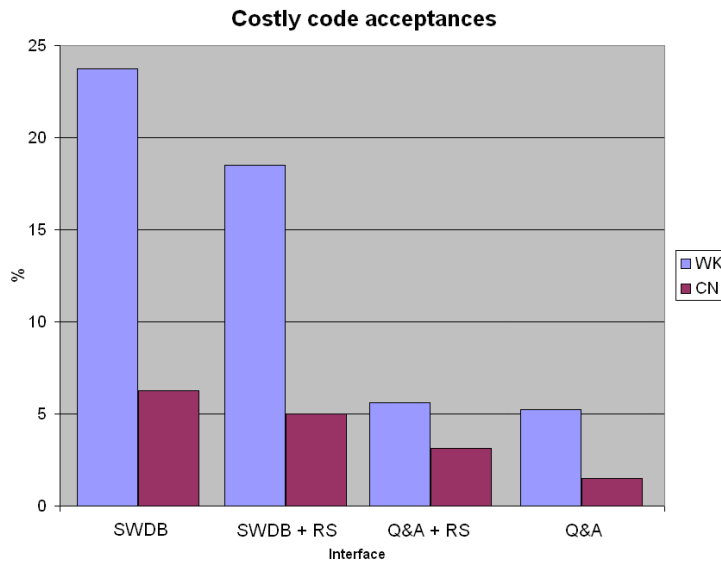


Figure 7. Download acceptances of costly software well-known publishers (WK) and publishers with a common name (CN). The Reputation System has mitigating effects between the SWDB and the Q&A.

The motivations given by the users for either accepting or refusing software download are shown in Fig. 8 and 9, respectively (data are taken from Tables 7 and 8 in Appendix A). Both figures show the effect of the Q&A on the SWDB, as well as the mitigating role played by the RS between the SWDB and the Q&A. Aligning acceptance motivations from the four interfaces, the Q&A pushes free motivations reducing interest (I) and trust (T) motivations. Dually, as to refusal motivations, the Q&A emphasizes the influence of motivations related to high Cost (Fig. 9), still reducing the other refusal motivations, namely non-interest (NI) and distrust (DT), from the average rows. Looking back at the histograms about the *incoherent behaviors* (Fig. 7), the Q&A related acceptances and motivations match more closely with actual users intentions compared to the other tested interfaces, as the Q&A nullifies incoherent behaviors.

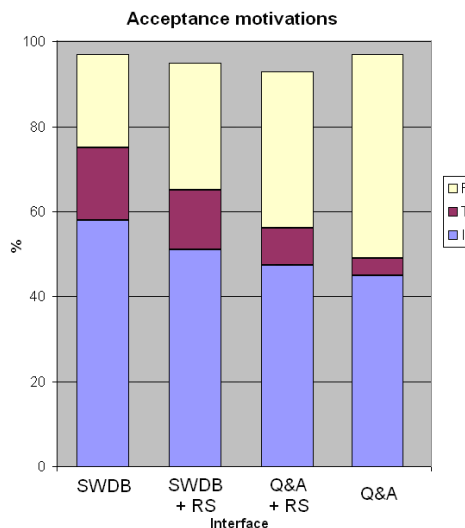


Figure 8. Average motivations of acceptances for Interest (I), Trust (T), Free cost (F) among the interfaces.

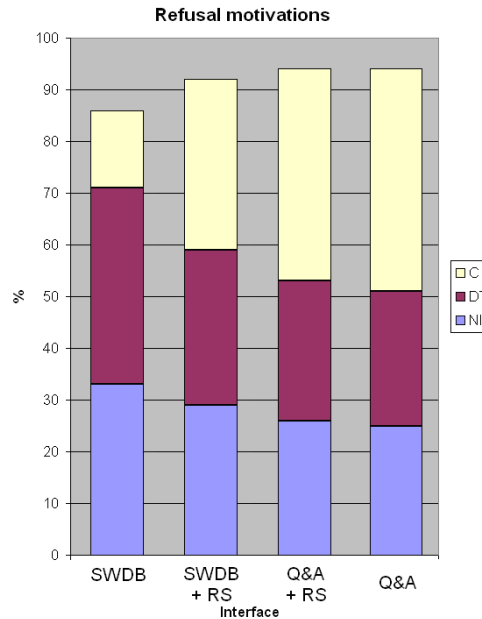


Figure 9. Average motivations of refusals for No Interest (NI), DisTrust (DT), Cost (C) among the interfaces.

6. CONCLUDING DISCUSSION

The usability studies as a whole have shown stable trends in user behaviors while interacting with the different graphical interfaces. The Question-&-Answer interface focuses very significantly the user attention on cost aspects, driving users to largely accept free software. Adding a Reputation System in the Q&A interface causes a reduced relevance of cost, while in the classic SWDB interface it reduces a little incoherent user behaviors and influences a little their trust beliefs.

The overall results, as shown in the General Trends section, suggest guidelines to be considered while choosing the best interface depending on the brand effect of the software publisher and/or the need to increase the software market penetration. From our design, where the Q&A asked for cost information, it derives that a Q&A-like interface is to choose if software is free, while, if software entails a charge, such an interface should be proposed if the publisher wants to ensure that users fully understand cost aspects of the product offered. In detail, our Q&A interface prevents from having users who in the future give negative feedback about a publisher because the interface is very robust in not causing *incoherent behaviors*. However, the Q&A can increase the attention level on any content of the interface, so if the publisher wants to increase the attention on aspects other than cost, like brand name, this can be an option, though not investigated in our tests.

The classic SWDB interface results now somehow out-of-date because of the spreading of social aspects in a plethora of web based activities. Providing a feedback aggregation mechanism (such as a reputation system) upon software download is a feature that is not unexpected by the users, as also shown in our questionnaire answers. For this reason the publisher should not provide an RS only if negative feedback is likely to be received. This might be the case of software out as alpha- or beta-version, that still has not the desired quality. When the publisher has not a strong brand effect, but the product is valid, providing an RS has a positive effect on software downloads. When the RS is embedded in the Q&A we designed, it increases trust-based motivations, reducing those related to cost aspects. However, enriching the Q&A with the RS shows *incoherent behaviors* similar to the common SWDB interface with RS, and an acceptance rate similar to the plain Q&A for well-known publishers, so its adoption has little advantages. Conversely, the Q&A with RS should be preferred compared to the plain Q&A by common-name publishers as it provides higher acceptances for software entailing a charge, keeping incoherent behaviors low.

REFERENCES

- Allen M., Currie L.M., Bakken S., Patel V.L. and Cimino J.J., 2006. Heuristic evaluation of paper-based Web pages: A simplified inspection usability methodology; *Journal of Biomedical Informatics*, v.39(4):412-423; Elsevier.
- BMS, 2012. *Browser Market Share*, Retrieved online at: <http://marketshare.hitslink.com/browser-market-share.aspx?qprid=1&qpcustomb=0>
- Boella G., Remondino M., 2010. Collaboration and human factor as drivers for reputation systems effectiveness; *Lecture Notes in Business Information Processing*, v.45(1):3-16, Springer.
- Bolton G.E., Katok E., Ockenfels A., 2004. How effective are electronic reputation systems? An experimental investigation; *Management Science*, V.50(11):1587-1602; Informa.
- Brustoloni J.C., and Villamarin-Salomon R., 2007. Improving security decisions with polymorphic and audited dialogs. In *Symposium on Usable Privacy and Security SOUPS 2007*, 76-85, ACM.
- Chen M., Liu X., 2011. Predicting popularity of online distributed applications: iTunes app store case analysis, *ACM 2011 iConference*, pp.:661-663.
- Cheskin research and studio archetype/Sapient, 1999. Ecommerce trust study, retrieved online at www.cheskin.com.
- Chia P.H., Heiner A., Asokan N., 2010. The wisdom of cliques: use of personalized social rating for trustworthy application installation; *NRC-TR-2010-001*, Nokia.
- CommerceNet, 2000. Barriers to electronic commerce 2000 study. Retrieved online at www.commerce.net.
- Corritore C.L, Kracher B. and Wiedenbeck S., 2003. On-line trust: concepts, evolving themes, a model. *Int. J. Human-Computer Studies*, 58, pp.:737-758, Elsevier.
- Cranor, Lorrie Faith. Security and usability: Designing secure systems that people can use. O'reilly, 2007.
- Dellarocas C., Resnick P., 2003. Online reputation mechanisms a roadmap to future research; *Summary report of the first Interdisciplinary Symposium on Online Reputation Mechanisms*, pp.:16, Cambridge, USA.
- Dini G., Foglia P., Prete C.A., Zanda M., 2006. An Analysis of User Interface Factors influencing the Acceptance of Code Download. *28th IEEE Information Technology Interfaces Conference*. pp. 233-238.
- Dini G., Foglia P., Prete C.A., Zanda M., 2007. Effects of Increasing Users' Attention on Cost in Software Download Interfaces on the Internet. *29th IEEE International Conference on Information Technology Interfaces*, pp.:227-232.
- Dini G., Foglia P., Prete C.A., Zanda M., 2013. A social-feedback enriched interface for software download, *Journal of Organizational and End User Computing*, in press.
- Egger F.N., 2000. "Trust me, I'm an online vendor": towards a model of trust for e-commerce system design; *CHI'00 extended abstracts on Human Factors in computing systems*, pp.:101-102, ACM.
- Flavian C., Guinaliu M. and Gurrea R., 2006. The role played by perceived usability, satisfaction and consumer trust on website loyalty; *Information & Management*, v.43(1):1-14, Elsevier.
- Fogg B.J., Mashall J., Laraki O., Osipovic, A., Varma, C., Fang, N., Paul, J., Rangnekar, A., Shon, J., Swani, P. and Treinen, M., 2001. What makes Web sites credible? A report on a large scale quantitative study. *Proceedings of ACM CHI'01*, pp.:61-68.
- Fullam K.K., Klos T.B., Muller G., Sabater J., Schlosser A., Topol Z., Barber K.S., Rosenschein J.S., Vercouter L., Voss M., 2005. A specification of the agent reputation and trust (ART) testbed: experimentation and competition for trust in agent societies, *Int. Conf. on Autonomous agents and multi agent systems (AAMAS05)*, pp.:512-518, ACM Press.
- Garcin F., Faltings B., Jurca R., Joswig N., 2009. Rating aggregation in collaborative filtering systems; *Third ACM Conf. on Recommender Systems*, pp.:349-352, ACM Press.

- Gefen D. and Straub, D., 2004. Consumer trust in B2C e-commerce and the importance of social presence: Experiments in e-products and e-services; *Omega*, 32, 407-424.
- Glantz S. A., 2005. *Primer of Biostatistics*. McGraw-Hill Medical Publishing, New York.
- Hoffman L.J., Blum J. and Lawson-Jenkins K., 2006. Trust beyond security: an expanded trust model; *Communications of the ACM*, v.49(7):94-101.
- Jansen W.A., 2000. Countermeasures for mobile agent security. *Computer Communications*, v.23(17), pp.:1667-1676, Elsevier.
- Josang A., Ismail R., Boyd C., 2007. A survey of trust and reputation systems for online service provision; *Decision Support Systems – Emerging issues in collaborative commerce*, v.43(2):618-644, Elsevier.
- Laberge J. and Caird J.K., 2000. Trusting the online banking interface: development of a conceptual model relevant to e-commerce transactions. *Designing interactive systems for 1-to-1 e-commerce workshop at the ACM SIGCHI conference '00*, The Hague, Netherlands.
- Lanford P., 2006. E-commerce: a trust perspective, in *Proc. of the 2006 International conference on Semantic Web and Web services, the 2006 World Congress in Computer Science, Computer Engineering, and applied Computing*, 2006.
- Likarish, P., Dunbar, D., Hourcade, J.P., Jung, E., 2009. Bayeshield: conversational anti-phishing user interface; *Symposium on Usable Privacy and Security (SOUPS 2009)*, ACM.
- Lin X., Rong G. and Thatcher J.B., 2009. Swift Trust in Web Vendors: the role of appearance and functionality; *Journal of Organizational and End User Computing*, v.21(1):88-108; IGI.
- Maxion R.A. and Reeder R.W., 2005. Improving user-interface dependability through mitigation of human error; *Int. J. of Human-Computer Studies*, v.63(1-2):25-50; Elsevier.
- Mayer R., Davis J. and Schoorman F., 1995. An integrative model of organizational trust; *Academy of Management Review*, 20(3):709-734.
- Meyerson D., Weick K. and Kramer R., 1996. Swift trust and temporary groups; *Trust in organizations: frontiers of theory and research*, (eds.: Kramer & Tyler), pp.:166-195; Sage Publications.
- Nilsson, M., Adams, A. and Herd, S., 2005. Building security and trust in online banking, *ACM CHI 2005*, pp.:1701-1704, Portland, USA.
- Olson J.S. and Olson G.M., 2000. i2i trust in ecommerce. *Communications of the ACM*, 43(12): 41-44.
- Resnick P., Kuwabara K., Zeckhauser R. and Friedman E., 2000. Reputation systems, *Communications of the ACM*, v.43(12), pp.:45-48, USA.
- Schlosser A., Voss M., Bruckner L., 2006. On the simulation of global reputation systems, *Journal of artificial societies and social simulation*, v.9(1):1-23, Surrey University, UK.
- Shin D.H., 2010. The effects of trust, security and privacy in social networking: a security-based approach to understand the pattern of adoption; *Interacting with Computers*, v.22(5):428-438, Elsevier.
- Shukla S., Nah F.F.H., 2005. Web browsing and spyware intrusion, *Communications of the ACM*, v.48(8):85-90, ACM 2005.
- Talwar A., Jurca R., Faltings B., 2007. Understanding user behavior in online feedback reporting; *Proc. Of the 8th ACM Conference on Electronic Commerce*, pp.:134-142, ACM Press.
- Thawte, 2007. *Certify your software integrity with Thawte code signing certificates*, Technical Report, Thawte, USA.
- Trevino L.K. and Webster J., 1992. Flow in computer-mediated communication: Electronic mail and voice mail evaluation and impacts. *Comm. Resear.* 19, 539–573.
- Walsh K. and Siner E.G., 2006. Experience with a distributed object reputation system for peer-to-peer filesharing; *USENIX Symposium on Networked System Design and Implementation*.
- M. Wood, 'WANT MY AUTOGRAPH?': THE USE AND ABUSE OF DIGITAL SIGNATURES BY MALWARE, the 20th Virus Bulletin International Conference - 29 September - 1 October 2010, Vancouver, BC, Canada.

Appendix A – Tables with full results

| Interface | SWDB | | | | | | Q&A | | | | | |
|-----------|-----------------|--------|-------|--------------|--------|-------|-----------------|--------|-------|--------------|--------|-------|
| | Acceptances (%) | | | Refusals (%) | | | Acceptances (%) | | | Refusals (%) | | |
| | Free | Costly | Sum | Free | Costly | Sum | Free | Costly | Sum | Free | Costly | Sum |
| WK | 40 | 23.75 | 63.75 | 10 | 26.25 | 36.25 | 40.25 | 5.25 | 45.5 | 9.75 | 44.75 | 54.5 |
| CN | 30 | 6.25 | 36.25 | 20 | 43.75 | 63.75 | 38.75 | 1.5 | 40.25 | 12.25 | 47.5 | 59.75 |
| DN | 22.5 | 8.75 | 31.25 | 27.5 | 41.25 | 78.75 | 24 | 5 | 29 | 23.5 | 47.5 | 71 |
| Average | 31 | 13 | | 19 | 37 | | 34.5 | 4 | | 15 | 46.5 | |

Table 5. Acceptance and refusal rates with the plain Security Warning dialog box (SWDB) and the Question & Answer interface.

| Interface | SWDB + RS | | | | | | Q&A + RS | | | | | |
|------------------------|-------------------|-------------------|-------|-------------------|--------------------|-------|----------------------|-------------------|-------|----------------------|---------------------|-------|
| | Acceptances (%) | | | Refusals (%) | | | Acceptances (%) | | | Refusals (%) | | |
| | Free | Costly | Sum | Free | Costly | Sum | Free | Costly | Sum | Free | Costly | Sum |
| WK | 41.88 | 18.38 | 59.26 | 8.12 | 32.62 | 40.74 | 43.75 | 5.63 | 49.38 | 6.25 | 44.37 | 50.62 |
| <i>RS: high / n.p.</i> | <i>49,75 / 34</i> | <i>20 / 16,75</i> | | <i>0,25 / 16</i> | <i>30 / 34,25</i> | | <i>46 / 41,5</i> | <i>10 / 1,25</i> | | <i>5 / 7,5</i> | <i>40 / 48,75</i> | |
| CN | 26.88 | 5 | 31.88 | 23.12 | 45 | 68.12 | 26.25 | 3.13 | 29.38 | 23.75 | 46.87 | 70.62 |
| <i>RS: high / low</i> | <i>29,75 / 24</i> | <i>8 / 2</i> | | <i>20,25 / 26</i> | <i>41,5 / 48,5</i> | | <i>36,25 / 16,25</i> | <i>5 / 1,25</i> | | <i>13,75 / 33,75</i> | <i>45 / 48,75</i> | |
| DN | 15.75 | 5 | 20.75 | 34.25 | 45 | 79.25 | 25 | 4.38 | 29.38 | 25 | 45.62 | 70.62 |
| <i>RS: n.p. / low</i> | <i>18,5 / 13</i> | <i>5 / 5</i> | | <i>31,5 / 37</i> | <i>45 / 45</i> | | <i>33,75 / 16,25</i> | <i>6,25 / 2,5</i> | | <i>16,25 / 33,75</i> | <i>43,75 / 47,5</i> | |
| Average | 28.2 | 9.8 | | 21.8 | 40.2 | | 32 | 4.5 | | 18.5 | 45.5 | |

Table 6. Acceptance and refusal rates for the interface with the Reputation System. Results are given for each code publisher name type, detailing for the reputation system scores.

| | | Interface | | | | | | | |
|----------|--------------|-----------|-------|-------|-------|-----|-----|-----|-----|
| Decision | | SWDB | | | | Q&A | | | |
| Accept | Motivation → | I | T | F | Os | I | T | F | Os |
| | WK | 49% | 23.5% | 25.5% | 2% | 36% | 11% | 53% | 0% |
| | CN | 72.4% | 6.9% | 20.7% | 0% | 50% | 0% | 50% | 0% |
| | DN | 52% | 20% | 20% | 8% | 50% | 0% | 41% | 9% |
| | Average | 58% | 17% | 22% | 3% | 45% | 4% | 48% | 3% |
| Refuse | Motivation → | NI | DT | C | Os | NI | DT | C | Os |
| | WK | 58.6% | 13.8% | 13.8% | 13.8% | 31% | 6% | 59% | 4% |
| | CN | 23.5% | 39.2% | 21.6% | 15.7% | 32% | 13% | 40% | 15% |
| | DN | 16.4% | 61.8% | 10.9% | 10.9% | 12% | 58% | 30% | 0% |
| | Average | 33% | 38% | 15% | 13% | 25% | 26% | 43% | 6% |

Table 7. Motivations for acceptance/refusal with the SWDB interface and the Question & Answer, respectively.

| | | Interface | | | | | | | |
|----------|--------------|-----------|-----|-----|-----|----------|-----|-----|-----|
| Decision | | SWDB + RS | | | | Q&A + RS | | | |
| Accept | Motivation → | I | T | F | Os | I | T | F | Os |
| | WK | 61% | 17% | 20% | 2% | 46% | 15% | 34% | 5% |
| | CN | 49% | 16% | 29% | 6% | 45% | 10% | 40% | 5% |
| | DN | 53% | 9% | 32% | 6% | 51% | 4% | 34% | 11% |
| | Average | 54% | 14% | 27% | 4% | 47% | 10% | 36% | 7% |
| Refuse | Motivation → | NI | DT | C | Os | NI | DT | C | Os |
| | WK | 32% | 11% | 45% | 12% | 32% | 6% | 58% | 4% |
| | CN | 34% | 30% | 29% | 7% | 32% | 25% | 33% | 10% |
| | DN | 20% | 58% | 14% | 8% | 13% | 51% | 32% | 4% |
| | Average | 29% | 33% | 29% | 9% | 26% | 27% | 41% | 6% |

Table 8. Motivations for acceptance/refusal with the SWDB+RS interface and the Q&A + RS, respectively.

Appendix B – Final Questionnaire

- 1) How often do you use the Internet? (Rarely, Once a month, once a week, Every day)
- 2) Why do you use the Internet? (Work, Study, Entertainment, Other)
- 3) Do you usually download software? (Never, At times, Frequently)
- 4) As a Internet user, have you ever seen the “Security Warning” dialog box? (Yes, No, Don’t remember)
- 5) How do you usually behave when confronted with a “Security Warning” dialog box? (Click Yes, Click No, It depends)
- 6) Do you usually pay attention to what is written in the “Security Warning” dialog box? (Yes, No, At times)
- 7) While deciding whether to accept or refuse a download, the publisher name matters to me. (Agree completely, Agree somewhat, Yes/no, Disagree somewhat, Disagree completely)
- 8) Do you trust more well-known publishers? (A lot, Not much, Yes/no, Little, Very little)
- 9) Did you notice publisher names similar to famous ones? (Yes, No)
- 10) How did you behave when dealing with publishers with a name similar to those of well-known ones? (Always accepted, Always refused, It depends)
- 11) Do you know what a premium rate connection is? (Yes, No)
- 12) Is it a risk for you that software download entails a charge? (A lot, Not much, Yes/no, Little, Very little)
- 13) Is it a risk for you that software usage implies a premium rate connection? (A lot, Not much, Yes/no, Little, Very little)
- 14) How much are you willing to spend to use a piece of software? (0 - I don’t want to spend, 3€, 10€, 30€ for full download, more than 30€ for full download)
- 15) Would you feel safer if software which entails a charge were guaranteed by well-known entities? (A lot, Not much, Yes/no, Little, Very little)
- 16) Before this test, had you already dealt with a Reputation System and feedback systems? (Yes, No)
- 17) How much were you influenced by the extra information provided by the reputation system? (A lot, Not much, Yes/no, Little, Very little)
- 18) How much do you trust a publisher with no reputation system? (A lot, Not much, Yes/no, Little, Very little)
- 19) Do you consider it useful to know the precise number of users who gave feedback comments? (A lot, Not much, Yes/no, Little, Very little)
- 20) Do you feel safer having a reputation system in the dialog box? (A lot, Not much, Yes/no, Little, Very little)

Authors biographies

Gianluca Dini received his “Laurea” degree in Electronics Engineering from the University of Pisa, Pisa, Italy, in 1990, and his PhD in Computer Engineering from Scuola Superiore “S. Anna”, Pisa, Italy, in 1995. He is Associate Professor at the Information Engineering Department at the University of Pisa. His research interests are in the field of distributed computing systems, with particular reference to security. Currently he is working on security in networked embedded systems and intrusion detection in multi-agent systems. He has published more than seventy papers in international conferences, books and journals and has participated to many projects funded by the Commission of the European Community, the Italian Government and private companies.

Pierfrancesco Foglia is researcher at the Information Engineering Department at the University of Pisa since 2004. His studies address embedded systems with a focus on architectures and interfaces. He has published results of his works on conferences and journals mainly in the fields of cache memories and systems usability. He participated to several research projects funded by public and private entities.

Cosimo Antonio Prete is full professor at the Information Engineering Department at the University of Pisa since 2004. His research studies started with computer architectures, and currently he has been working also web usability and systems for workflow automation. He has published in international conferences, books

and journals, and has participated and coordinated research projects at a regional, national and European level.

Michele Zanda is post doctoral fellow at the Information Engineering Department at the University of Pisa. He received his Master of Science in Computer Engineering cum laude from the University of Pisa in 2004, and completed his Ph.D. studies in Computer Science and Engineering at the IMT Advanced Studies in Lucca in 2008. He investigated the usability of human computer interactions on both industrial systems and the Web. He assessed the effectiveness of enriching websites with intelligent avatars, and novel interaction paradigms and measurements to improve current user interfaces. Results of his studies were published on International conferences and journals.

First of all, we would like to thank the editors for the time and the effort devoted to the paper, and for their suggestions that considerably helped us to improve the quality of the paper.

The letter is organized in two parts. In the first part we answer to the questions raised by the Editor. In the second part, for each reviewer, we address in detail his/her comments and criticisms.

Answers to the Editor

The paper reports a study of how different download user interfaces affect the users behavior. The study might find some interesting results, but as the reviewers pointed out, there are many unclear issues in the writings and then I'm sorry but we could not consider the acceptance in the current form.

The major problems are summarized as below;

Need the improvements in the writings. The organization of the paper must be improved. And please carefully avoid the overlap and duplications of the contents.

We have improved the paper. In particular, we have largely rewritten the Introduction, with concepts streamlined. Furthermore, we have also worked Section 2 in order to reduce it and avoid duplication and overlapping, as suggested by Reviewer 2. Finally, we have thoroughly proofread the paper.

The experimental design and methodology must be described clearly. Especially the description about the user's tasks, what the participants were asked to do in what kind of situation/scenario/task context must be described clearly.

We have added a full description of the user's tasks, together with a description of the context of the research project. A more detailed answer can be found in the answer to Question #5 of Reviewer #1.

Please clarify the methodology, the target and the context of the study and describe them explicitly in the organized manner.

As required by Reviewer #1 in Question #3, we have better explained the rationale of our different interfaces design. Furthermore, according to Question #5 of Reviewer #1, we have also improved the description of user tasks, and according to Question #6 of Reviewer #1, we have improved the use of acronyms and the description of hypothesis. Finally, according to Question #3 of Reviewer #2, we have clearly described the test procedure, so that now methodology, target and context are clearly explained in the paper. More details can be found in the answers to the aforementioned questions.

The terms and concepts must be clearly defined. For example, need to clarify what "The" common download interface is. And the major concepts must be described by the same terms through the paper.

We clarified what the common download interface is, and streamlined in the paper the descriptions of the concepts introduced. In particular, we have reworked the abstract and other sections in the paper, where the common download interface was referenced inappropriately. The details of the major changes are reported as answer to Question #2 of Reviewer #1.

Literature reviews must be improved and enriched by adding the related works by other researchers rather than self-references.

We have improved the literature review. Furthermore, as recommended by both the reviewers, we have validated our claims by citing other authors, and adding references relevant to the issues raised. Among the others, we added and cited the following ones:

- Allen M., Currie L.M., Bakken S., Patel V.L. and Cimino J.J., 2006. Heuristic evaluation of paper-based Web pages: A simplified inspection usability methodology; *Journal of Biomedical Informatics*, v.39(4):412-423; Elsevier.
- Maxion R.A. and Reeder R.W., 2005. Improving user-interface dependability through mitigation of human error; *Int. J. of Human-Computer Studies*, v.63(1-2):25-50; Elsevier.
- Cranor, Lorrie Faith. Security and usability: Designing secure systems that people can use. O'reilly, 2007.
- M. Wood, "Want my autograph?: The use and abuse of digital signatures by malware," the 20th Virus Bulletin International Conference - 29 September - 1 October 2010, Vancouver, BC, Canada.

For greater details, please refer to the answers to reviewers' questions.

Detailed Answers to Reviewers

Reviewer #1

This paper describes a study of how different download user interfaces affect 'incoherent behavior' when choosing to download software, where incoherent behavior concerns people providing reasons for their decisions that are based upon incorrect facts (like it wasn't free when it was). The study found that Ratings about the software detract from the clarity that Q&A about the product provide, but improve over standard download prompts with security information. Recommendations are made according to different conditions, such as if the software is free etc.

My recommendation is for it to be resubmitted later on with more detail, primarily because there is clear methodological information that makes it difficult to know whether to accept.

Question #1: In my opinion, the main flaw of the paper is that the writing makes many presumptions, and often, therefore, fails to provide sufficient information and context. Some other writing characteristics make it difficult to follow too.

We have removed all presumptions in the paper and added information and context. More in the following. Furthermore, we have thoroughly proofread the paper.

Question 2. The first sense of missing context and presumption about the readers, is in referring to 'the common download interface' in the abstract. "The" common download interface is immediately unclear. In fact the majority of the abstract/introduction, when read alone, could refer to the design of websites that provide information about a software product you might want to download, which is what one may immediately think of when considering a download interface. The paper could, instead, encourage a larger number of readers by being much clearer, in a context, that this is an investigation about

download/installation prompts after having chosen to download software from the web. It can be clear that this is about making an installation decision after having chosen to download something, based upon security issues. The paper also could also be clearer that they are 'the common download interface' based on a certain operating system, and instead step back a bit about focusing on conceptually what 'they do', since the three comparisons are about conceptual approaches to this install check.

According to the reviewer comment, we have clarified soon in the Abstract and the Introduction what a download interface is. We have also defined what we intend by “the common interface,” namely the download interface delivered with the Windows operating system within the Microsoft Authenticode framework. We have chosen this framework because it is a widely used one as it is embedded into Internet Explorer. We have also clarified that the paper is not about websites interfaces, but it addresses the design of dialog boxes to either accept or refuse software download.

Question #3. One thing that remains very unclear in the paper is why the focus of the Q&A design is on cost of the product. This makes a fundamental assumption, along the lines of that people want to download free software only. Are there a range of Q&As that can be conceptually applied, or is it that cost information should be much more apparent in general in order to make decisions. because the standard SWDB could be augmented with cost information more clearly. So again, this is probably do with lack of clarity and presumed prior knowledge that the full description and comparison of conceptual approaches, in a clear use scenario, is missing. Because you are potentially comparing security warning vs cost information vs rating information, rather than standard, vs questioning someone, vs giving social feedback. because the Q&A condition is confounded by both the focus on money and the act of answering a question. Section 3 is essentially under described.:

We agree that the Q&A interface has been presented with little context. We clarified the rationale of the Q&A interface in Section 3. We clarified that the aim of asking a question is increasing the overall attention level of the user, in particular to try to reduce incoherent behaviours. This is a general behaviour of users in front of such an interface, as also stated in [2]. Besides [1] highlights the poor design of the Microsoft Authenticode interface. Our Q&A design try to solve both the issues, by forcing the user to put attention on interface content. Section 3 was streamlined, by removing useless subparagraphs, and by adding descriptive contents. The major changes applied to the section are the following:

OLD VERSION [§ SECTION 3]

The Question-&-Answer interface (Q&A) [Dini et al., 2007] asks for and checks for the user input about cost (fig. 4).

New Version [§ SECTION 3]

The Question-&-Answer interface (Q&A) [Dini et al., 2007] utilized in our study is a wizard with three steps. The first step replicates the information of the common SWDB, while the second step asks the cost of the software under download, and the third steps allows to download the software if the given answer is correct (Fig. 3). The reason why we choose to focus on the cost information is the following. In a previous work [Dini et al., 2007] we found incoherent behaviors of users with respect to cost, namely users stated they wanted to download free software whereas in practice they actually downloaded expensive one (or vice versa). We also proved that the Q&A interface mitigates such a behavior. Therefore, in this work, we decided to start from such consolidated results. Conceptually, the Q&A interface may present questions on any other information that is considered relevant, such as name of the producers, name of the software, et cetera.

Figure 4 shows the other interfaces used in our experiment. They are, respectively, the Security Warning Dialog Box (left) and the Q&A interface (right), both augmented by a reputation systems which display the aggregation of users rankings regarding the software to be downloaded [Dini et al., 2013]. As shown in Fig. 4, the RS is placed at the bottom of the dialog box, showing the total number of feedback reviews, as well as the rate of positive, neutral, and negative reviews. According to the classification of RS ranking aggregation mechanisms proposed by [Josang et al. 2007], the interface, like that of the eBay Feedback Forum, is a simple summation system. In a simple summation RS, the system accumulates all given ratings to get the overall reputation. The RS is simple and readable at a glance, providing a concise view of the distribution of users' feedback [Dini et al., 2013].

We also added the following sentences in the conclusion:

[§ CONCLUDING DISCUSSION]

.From our design, where the Q&A asked for cost information, it derives that a Q&A-like interface is to choose if software is free

[...]

However, the Q&A can increase the attention level on any content of the interface, so if the publisher wants to increase the attention on aspects other than cost, like brand name, this can be an option, though not investigated in our tests.

Question #4: A relatively good literature review is presented, which would be better with the document as a whole making less presumption about context. One thing that appears strange is in the initial motivation in the introduction, a paper about security attacks is presented as the reference for "User interfaces for software download on the Internet are the front end of software publishers for remote users. The design of these interfaces is critical for users who may make unconscious or misguided decisions if interfaces are hard to understand and use" which is not a sentence about security concerns. its a sentence about how confusing user interfaces are. So many other papers, like heuristic evaluation, could be cited here. there is just a mismatch between statement and why the reference is chosen.

According to the reviewer's comment, we removed old references and added new, more relevant ones, in particular in the Introduction that has been partly rewritten.

Question 5. My main concern with being able to recommend if the paper is accepted or not is missing information about the study. It is not at all clear what the user task is. It appears to be here is a bunch of prompts, should you install it or not. the article does not describe what the participants were asked to do. Were they given a scenario: you need to download software for a project, the software is designed to do X,Y,Z. use these prompts to make your decision. There is just no information about the task context. Because the options for choice are: Not Interested etc. How can someone make that decision without some sort of scenario. And software download decisions can also be highly dependent (ie the importance of price) by the importance of the task the software is needed for. So what is the task context here. Its not clear what is happening that led to your data, and without this we cannot judge as to whether this data is valuable/valid. Section 4 is missing critical information.

According to the reviewer's comment, we added the task description, and we also added information on the context of the experiment, and on its limitations. In particular, *We added in section 4.2 the following descriptions:*

OLD VERSION:

[§ TEST PROCEDURE]

In the first stage we display a web page that proposes a software download. In the second stage, the user decides whether to accept or not the piece of software, by using one of the interfaces. In the third stage, the user is required to motivate their choice with a motivation radio-button. A user, following their decision of code acceptance or code refusal, must choose, with a radio-button, the major motivation that drove their decision. The given motivations, together with the acceptance or refusal of a piece of software, are dependent variables in our experiments. The motivation radio-button presents a set of options for participants to choose. The content of this window is described in table 3 (column Motivation). If the user downloads a piece of software, they are shown a list of motivations mainly related to interest (I), trust (T), free cost (F) and other minor motivations (O). If the user refuses the download, they are shown another list of motivations such as no interest (NI), distrust (DT), high cost (C) and other minor ones (O).

A gender balanced group of 40 participants was recruited, with ages ranging from 18 to 40. 34 participants were high school students or undergraduate students, while 6 participants were employed in public administration. Participants attended the tests in person in our labs or in their office (rather than remotely via web). The recruited group had good Internet experience, as shown in table 2:

[§ VALIDITY]

We minimized learning effects by not giving the users any feedback on each single test cases (e.g.: free acceptances/acceptances entailing a charge and incoherent motivations). We observed that the actions taken by the users with tested interfaces were very conservative: they tried to minimize possible risks (even if they were aware that the experiment could not harm their terminal). Thus, we believe that the test environment had little impact on our results.

NEW VERSION

[§ TEST PROCEDURE]

The test consists of a sequence of test cases, where each test case consists of three stages. In the first stage we display a web page that proposes a software download. In the second stage, the participant decides whether to accept or not the software product by using one of the displayed interfaces. In the third stage, the participant is required to motivate his/her download choice with a motivation radio-button. The given motivations, together with the acceptance or refusal of a piece of software, are dependent variables in our experiments. The motivation radio-button presents a set of options for participants to choose. The content of this window is described in Table 2 (column Motivation). If the participant accepts to download, then he/she is presented a list of motivations mainly related to interest (I), trust (T), free cost (F) and other minor motivations (O). In contrast, if the participant refuses to download, he/she is shown another list of motivations related to no interest (NI), distrust (DT), high cost (C) and other minor ones (O).

At the end of the 36 test cases, participants answered a multiple-choice questionnaire containing personal questions and questions about the used interface. The full questionnaire is reported in Appendix B. , As each participant had to complete 36 test cases, we minimized learning effects by not giving participants feedback on their actions during the experiment.

In a previous work [Dini et al., 2006], it was observed that user intentions and proper behaviors for each single test case could not be derived by just looking at the final questionnaire answers, as each test case needed a precise motivation. For this reason, we let participants choose the most suitable

motivation out of a set (see Table 2), after either accepting or refusing a downloading. By doing this, participants can motivate their decisions with precision. In addition, we used motivations to clearly identify participants' coherent and incoherent behaviors. In fact, in the cited previous works, it was observed that users interacted with interfaces incoherently at times: a participant behaves incoherently if his/her motivation does not match with the features of a piece of software downloaded or refused. In Table 2, column "Incoherent, if in combination with" shows when motivations not matching actual software features identify incoherent behaviors. An example of an incoherent behavior is if a participant refuses a download picking the motivation "I could be interested, but it was expensive", while the software is free.

At the beginning of a test session each participant was asked to "either accept or refuse the proposed software as if you were using your own computer", since we want to simulate the scenario of a typical Web user who browses the Internet and downloads software according to its own interests. Participants were not rewarded for their time, but they were told: "you are participating to the design of a new user interface for e-government web applications". A gender balanced group of 40 participants was recruited, with ages ranging from 18 to 40 (mean age 23). 34 participants were high school students or undergraduate students, while 6 participants were employed in public administration. Participants attended the tests in person in our labs or in their office (rather than remotely via web). The recruited group had good Internet experience, as shown in Table 3. All the experiment were performed on a windows based platform.

(FOOTNOTE) ²The work was co-founded by the "Cassa di Risparmio di Pisa" in the framework of the Easy.Gov project on Public Administration websites and Web usability.

[§ VALIDITY]

We minimized learning effects by not giving the users any feedback on each single test cases (e.g., users performing free acceptances or acceptances entailing a charge without highlighting, during the test, incoherent motivations). We observed that the actions taken by the users with tested interfaces were very conservative: they tried to minimize possible risks (even if they were aware that the experiment could not harm their terminal). Thus, we believe that the test environment had little impact on our results. In addition, we did not observe a lack of motivation in the participants, also because after initial tests, we minimized the duration of a single test at less than 30 minutes on average.

There were three ranking categories shown in the Reputation System in our test: high and positive rankings, low and negative rankings, and rankings not supported. Although we observed that rankings tend to be classified mainly in line with these three categories, other studies should investigate how users behave while dealing with other reputation patterns.

Question #6. The results, in how they refer to the hypotheses, are ok. The whole presentation is a bit hard to follow because a) there are a large number of acronyms in use in the paper, and b) the hypotheses are presented using different terms to these acronyms. It would be much clearer if the hypotheses used RS instead of social-feedback (which I believe is the same thing). Also it would be a bit less confusing to use numbering rather than letters to enumerate the hypotheses (H1-7 rather than HA-HG).

We agree with the reviewer that presentation has to be improved. So, according to the reviewer's indication, acronyms have been reduced, in particular, those deriving from concepts of other

researchers cited in the background section. We now use RS in the hypotheses rather than social-feedback: they are the same thing but we used social-feedback because we thought that it could be easier to understand, but now with fewer acronyms the paper is easier to read. Hypotheses have been numbered and rephrased (see Section 2.4), as suggested by the reviewer.

In particular, *the removed acronyms are*: CA (for certification authority), W&C (for warn and continue), CSG (for context sensitive guidance), MoTEC (Model of Trust for Electronic Commerce).

Hypotheses were rephrased:

And we also test **H5**, **H6** and **H7**:

H5: the RS with Q&A reduces the relevance of cost aspects when compared to the Q&A interface.

H6: the RS with Q&A increases the relevance of trust when compared to the Q&A interface.

H7: the RS with Q&A increases the relevance of trust when compared to the SWDB interface.

Overall, as stated at the beginning, I would recommend that this paper is resubmitted with significant improvements, so that a decision for acceptance can be judged on clearer methodological information, better contextual information, and clearer writing.

References for reviewer1

1. Cranor, Lorrie Faith. Security and usability: Designing secure systems that people can use. O'Reilly, 2007
2. M. Wood, 'WANT MY AUTOGRAPH?': THE USE AND ABUSE OF DIGITAL SIGNATURES BY MALWARE, the 20th Virus Bulletin International Conference - 29 September - 1 October 2010, Vancouver, BC, Canada.

Reviewer 2

This paper presents the results of a user test on the design of browser interfaces for software download for investigating how to reduce errors in users' decisions. The authors used the common software download interface as a benchmark and compared it with a Question-and-Answer interface, also embedding both interfaces with a Reputation System. The results of the study show that the pure Q&A interface is the strongest in minimizing incoherent behaviors, and that the differences in reputation rankings significantly influence users.

Question #1: The study seems to be well-founded and rigorous but the presentation is a bit heavy and suffers from too many repetitions: the first 4/5 pages could be greatly reduced.

According to the reviewer comments, we have reworked the first pages. In particular we have reworked the Introduction and Section 2.4, trying to avoid repetitions and reducing the length of the paper.

Question #2. There are too many self-references used to validate the study, it would be better to also reference related works of other authors.

We have increased the number of referenced works especially in the Introduction, to better position and validate our work. Among the others, we cite [1], which highlights that the Microsoft Authenticode interface was not successful in providing the user all the information they needed in order to make an informed decision. Furthermore, we added [2], which highlight that users seldom act appropriately when presented with Certification Authority warnings and often immediately accept a digital certificate with little or no scrutiny, without further investigating the reasons.

Question #3. Some points need to be clarified: Section 4.2 Test Procedure: 'The test consists of a sequence of test cases ...'. Is the sequence always the same? Have you split the sample in 4 groups in order to assign a different sequence order of the interfaces to each group, in order to balance the results.

We have clarified this point (see Section 4.2). In short, we define a pool of 36 pages divided in 4 groups as described in Figure 6. Such pages are then presented in a random order to every user.

Question #4. Table 2 reports the characteristics of the sample (40 users) but there is a 41st value. (?)

We have fixed it.

Question #5. Was the user test only made for a Windows platform?

Yes, we worked only on a Windows platform. We have better specified it in Section 4.2

Question #6. In Section 4.2 the average of the users' ages could be added.

We have fixed it.

Question #7: Table 5 could be moved from Appendix A to the paper - Section 5.

Since we believe that all tables have the same importance, we have preferred to keep all together and moved all of them into the Appendix in order to make the reading of the paper easier.

Question 8: Revise the paper accurately, there are some minor English revisions to make, e.g.: we compare such two designs --> we compare two such designs; in order to make users confident that they are being not cheated --> they are not being cheated; habituation --> habit; with respect to --> compared to; etc.

We have fixed all of them.

References for reviewer2

1. Cranor, Lorrie Faith. *Security and usability: Designing secure systems that people can use*. O'reilly, 2007.
2. M. Wood, '**WANT MY AUTOGRAPH?: THE USE AND ABUSE OF DIGITAL SIGNATURES BY MALWARE**', the 20th Virus Bulletin International Conference - **29 September - 1 October 2010**, Vancouver, BC, Canada.

Dear Editor,

This is the revised version of the paper Social and Q&A interfaces for App download, by G. Dini, P. Foglia, C.A. Prete, M. Zanda.

We thank both the editor and the reviewers for the time devoted to the reading of the paper, and we try to fix all the question raised by them.

Hoping to have good news from you,

Regards,

P. Foglia and M. Zanda

Highlights

Social and Q&A interfaces for App download

Submitted to: **Journal of Information Processing and Management.**

Gianluca Dini, Pierfrancesco Foglia, Cosimo Antonio Prete, Michele Zanda
Dipartimento di Ingegneria dell'Informazione: Informatica, Elettronica, Telecomunicazioni;
Università di Pisa, Via Diotallevi 2, 56100 PISA, Italy

These are the highlights of our revision:

We have improved the paper. In particular, we have largely rewritten the Introduction, with concepts streamlined. Furthermore, we have also worked Section 2 in order to reduce it and avoid duplication and overlapping, as suggested by Reviewer 2. Finally, we have thoroughly proofread the paper.

We have added a full description of the user's tasks, together with a description of the context of the research project. A more detailed answer can be found in the answer to Question #5 of Reviewer #1.

As required by Reviewer #1 in Question #3, we have better explained the rationale of our different interfaces design. Furthermore, according to Question #5 of Reviewer #1, we have also improved the description of user tasks, and according to Question #6 of Reviewer #1, we have improved the use of acronyms and the description of hypothesis. Finally, according to Question #3 of Reviewer #2, we have clearly described the test procedure, so that now methodology, target and context are clearly explained in the paper. More details can be found in the answers to the aforementioned questions.

As required by the editor, we clarified what the common download interface is, and streamlined in the paper the descriptions of the concepts introduced. In particular, we have reworked the abstract and other sections in the paper, where the common download interface was referenced inappropriately. The details of the major changes are reported as answer to Question #2 of Reviewer #1.

We have improved the literature review. Furthermore, as recommended by both the reviewers, we have validated our claims by citing other authors, and adding references relevant to the issues raised. Among the others, we added and cited the following ones:

- Allen M., Currie L.M., Bakken S., Patel V.L. and Cimino J.J., 2006. Heuristic evaluation of paper-based Web pages: A simplified inspection usability methodology; *Journal of Biomedical Informatics*, v.39(4):412-423; Elsevier.
- Maxion R.A. and Reeder R.W., 2005. Improving user-interface dependability through mitigation of human error; *Int. J. of Human-Computer Studies*, v.63(1-2):25-50; Elsevier.
- Cranor, Lorrie Faith. Security and usability: Designing secure systems that people can use. O'reilly, 2007.

- M. Wood, “Want my autograph?: The use and abuse of digital signatures by malware,” the 20th Virus Bulletin International Conference - 29 September - 1 October 2010, Vancouver, BC, Canada.

For greater details, please refer to the answers to reviewers’ questions.

Dr. Michele Zanda, contact author, email: michele.zanda@gmail.com
Prof. Gianluca Dini,
Prof. Pierfrancesco Foglia,
Prof. Cosimo Antonio Prete.