Practical Findings from Applying the PSD Model for Evaluating Software Design Specifications

Teppo Räisänen, Tuomas Lehto, and Harri Oinas-Kukkonen

University of Oulu, Department of Information Processing Science
Rakentajantie 3, 90570 Oulu, Finland
{Teppo.Raisanen,Tuomas.Lehto,Harri.Oinas-Kukkonen}@oulu.fi

Abstract. This paper presents practical findings from applying the PSD model to evaluating the support for persuasive features in software design specifications for a mobile Internet device. On the one hand, our experiences suggest that the PSD model fits relatively well for evaluating design specifications. On the other hand, the model would benefit from more specific heuristics for evaluating each technique to avoid unnecessary subjectivity. Better distinction between the design principles in the social support category would also make the model easier to use. Practitioners who have no theoretical background can apply the PSD model to increase the persuasiveness of the systems they design. The greatest benefit of the PSD model for researchers designing new systems may be achieved when it is applied together with a sound theory, such as the Elaboration Likelihood Model. Using the ELM together with the PSD model, one may increase the chances for attitude change.

Keywords: Persuasive Systems Design Model, specifications, research, practice, guide, analysis.

1 Introduction

As we have gained deeper understanding of the field of persuasive technology, the expectations for persuasive designs have also increased. This is especially evident in competitive markets such as Web 2.0, e-business and the mobile domain. When designing solutions for these kinds of environments, the designers must have a thorough understanding of the various persuasion, motivation and influence strategies to gain competitive advantage.

Oinas-Kukkonen and Harjumaa [1] have conceptualized a framework for designing and evaluating persuasive systems, known as the Persuasive Systems Design (PSD) model. In the PSD model, the categories for persuasive system design principles are: primary task support (supporting the user's primary task), dialogue support (supporting the interaction between the user and the system), system credibility (the more credible the system is, the more persuasive it is), and social support (the system motivates users by leveraging social influence). In addition to this, the model can be utilized to analyze the intent, event and strategy of the persuasion context.

T. Ploug, P. Hasle, H. Oinas-Kukkonen (Eds.): PERSUASIVE 2010, LNCS 6137, pp. 185–192, 2010.

As the PSD model is a meta-level model, it is designed to be used together with suitable theories, such as the Elaboration Likelihood Model [2], the Theory of Reasoned Action [3], the Theory of Planned Behavior [4], Social Cognitive Theory [5], or other behavior or attitude change theories. This paper will describe practical findings from applying the PSD model to evaluating software design specifications.

Section two will discuss persuasive technologies in the light of software requirements engineering. Section three will present the case, and section four the results. Section five will discuss the relationship between the PSD model and the Elaboration Likelihood model (ELM) and how to apply the PSD model in conjunction with the ELM. Finally, section six discusses the findings, and section seven concludes the paper.

2 Requirements Engineering

Requirements engineering is an early part of the software system development process. It is a "branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behavior, and to their evolution over time and across software families." [6]. Nuseibeh and Easterbrook [7] have identified the following core requirements engineering activities:

- Eliciting requirements.
- Modeling and analyzing requirements.
- Communicating requirements.
- Agreeing requirements, and
- Evolving requirements.

The first step of requirements engineering is the elicitation of requirements. It aims at identifying all the stakeholders –such as customers, developers and users – as well as the objectives and tasks of the system. Prototyping, brainstorming and focus groups are examples of elicitation techniques. In addition, various elicitation methods provide guidance on how to use the techniques in the elicitation process [7].

Modeling the requirements aims at producing abstract descriptions of the requirements so that they would be amenable to interpretation. For example, in the case of large information systems, data modeling is often used to understand how information is to be manipulated and managed [7]. In the case of behavior-change support systems [8], domain modeling and behavior modeling must be used to gain a deeper understanding of the intent, event and strategy of the targeted behavior change. Various analysis methods should also be used together with modeling techniques to validate and verify the models. Finally, modeling also transfers the requirements into a communicable form. This way they can be presented to the stakeholders.

The core activities of communicating and agreeing on the requirements with different stakeholders aims at establishing a consensus that the requirements and models elicited provide an accurate account of stakeholder requirements [7]. Finally, evolving requirements is about managing the changes to the requirement documents.

The research presented in this paper deals mostly with analyzing models. We utilized the Persuasive Systems Design (PSD) model [1] as a vehicle for the analysis. The next section will present the case.

3 Case Description

In this case, the PSD model was utilized for evaluating the requirements specifications of a mobile Internet device under planning. We received several requirements documents from Company X. The documents related to user interface and user interaction issues. They were rich with life-sized sketches of the proposed solutions. Figure 1 illustrates the pictures and the level of detail found in the set of documents. In addition to the touchpad gestures, the pictures in the specifications also had screenshots of user interface elements to illustrate how the gestures can be used to interact with the elements.

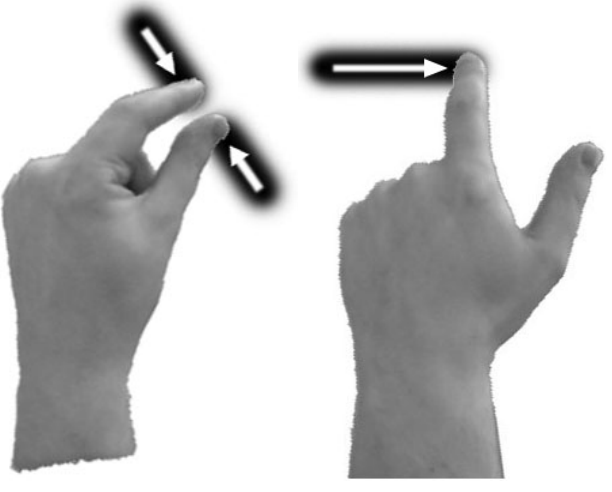

Fig. 1. Example picture from the software design specifications. The pictures demonstrate two touchpad gestures.

The evaluation protocol was as follows. Two researchers independently evaluated the design documents in order to recognize the persuasive features through the PSD model. After this, the findings were discussed and a joint understanding was formed through face-to-face discussion. All persuasive features were graded in terms of how well the planned implementation seemed to support the feature. Company X was then provided with the results and a short list of suggestions on how to improve the persuasiveness and usability of the system. Thus, even if we focused mostly on the analysis of the modeled requirements, we included some elements of requirement communication as well.

4 Results

Software design specifications define the roadmap for the system building process [9]. Evaluation of the specifications aims at ensuring that the requirements are feasible and that no crucial requirements are missing. In addition, the evaluation of

the specifications aims at picking up any problems before resources are committed to addressing the requirements [10].

The case demonstrated that the PSD model as such is feasible for evaluating design specifications. In this kind of work, it becomes imperative to clearly identify and define the primary task for the target of the evaluation. With mobile Internet devices, the primary task can vary greatly from browsing the Web to watching videos, social networking and all the way to using the device as a mobile phone. In this case, the device was of general purpose, i.e., it was intended to be used for any of the abovementioned purposes. This made the analysis of persuasiveness regarding the primary task issues somewhat more difficult than with a more precise goal for the device.

The main findings can be summarized as follows:

- The case demonstrated how the various PSD principles work together. For example, the principles of suggestion and reward are interlinked with each other. If the application suggests new ways of doing things for the user, some kind of rewarding feature would naturally supplement it.
- It seems to be that some principles do not work very well together. For example, the abundant use of reduction makes tunneling nearly useless.
- Our experiences confirm the claim put forward by the developers of the PSD model that there is no point in trying to utilize all persuasion techniques in a single case. Rather, a coherent set of the techniques should be recognized and selected. This should be done in such a manner that there would be as much synergy as possible between the different techniques included.

Additional persuasive effects may be achieved through the synergy between different design principles. Many of the principles in the PSD model seem to operate well in pairs or as part of larger sets. Paired principles, such as suggestion–reward, suggestion–personalization, self-monitoring–reminders, similarity–liking, competition–cooperation, and simulation–rehearsal can be effective together. For instance, users could rehearse the target behavior with simulations. Further examples would be a self-monitoring functionality that could be triggered by using reminders, or using suggestion to prompt the user to personalize an application or service.

Yet, harnessing principles with low synergy might result in less persuasive effects than anticipated. Thus, it is indeed a coherent set of persuasive design principles that may be able to provide advantages and the best persuasive results.

An aspect of the PSD model that posed a challenge for us was that some of the social support principles are highly interlinked with each other. The connections between social learning, social comparison, social facilitation, and normative influence are difficult to differentiate from each other. For example, it is difficult to see where social comparison starts and social learning ends. This overlapping makes the utilization of the model for evaluation purposes very challenging. Furthermore, in some cases, the persuasive principles may not be determined based solely on design specifications, as some of the principles (especially in the credibility support and social support categories) might only be observed in the actual system-to-user interaction situations.

The greatest challenge in applying the PSD model was that no explicit evaluation heuristics have been defined for it yet. The model would gain more strength from

explicitly defined scales and instructions for evaluating the implementation of each principle. These types of heuristics would substantially reduce the influence of the subjective views of an individual evaluator, and thus would diminish potential bias. Nevertheless, it should be noted that evaluation will always remains subjective. A further challenge is that since the PSD model incorporates a total of 28 design principles, the evaluation of specifications and/or applications may be quite laborious and time consuming.

Beyond the evaluation of persuasive principles, the PSD model can be utilized to recognize the intent (persuader, change type), event (use, user, and technology context) and strategy (message, route) of the persuasion context [11]. In this case, we were not able to extract this information from the software specifications beyond a rather generic level. (It should be noted that we were not part of the design team/organization.) In our view, defining the persuasion context early in the design process is essential and may eliminate costly rework.

5 The Relationship between Theories and the PSD Model

Practitioners with no strong theoretical background can apply the PSD model for their designs and/or evaluations and gain benefits from it, whereas researchers still need a theoretical framework for their own work to gain the full benefit from the PSD model. When applying the PSD model for research purposes for design, it is important to lean on an understanding provided by a persuasion-related theory to be able to select the desirable persuasive features through the PSD model.

For instance, the Elaboration Likelihood Model, or ELM for short, [2] is one of the most widely used theories to explain how attitude change happens. Next, we will describe how the PSD model could be used in conjunction with the ELM to increase the persuasiveness of a designed system. See Fig. 2 and note the arrows.

The figure shows the two parts of the ELM in two separate spirals. These are the central and peripheral routes (the PSD model calls these the direct and indirect routes). The direct route is used when information processing is based upon critical thinking. An example of this could be the decision to stop drinking alcohol. In contrast, the attitude shift through the indirect route is based on rules of thumb. These rules can be, e.g., personalized content [12]. In figure 1, the direct route is the outer spiral formed by persuasive communication, motivation, ability, cognitive processing and direct attitude change. The indirect route is the inner spiral formed by persuasive communication, indirect cues and indirect attitude shift. For clarification purposes, the indirect route is also drawn with dashed arrows. When the ELM is drawn in a spiral, it emphasizes the incremental nature of persuasion (cf. [1]).

Especially important in the ELM are the yes-arrows. The attitude change happens along them. As the direct attitude change is enduring, resistant and predictive of behavior, the outer part of the spiral is crucial. If we want to design the system to help achieve attitude change, the PSD principles should aim at producing this yes-effect (i.e., the principles should support movement along the yes-arrows in the figure). For example, dialogue support should aim at making sure that communication is persuasive (the yes-arrow from “persuasive communication” to “motivated to

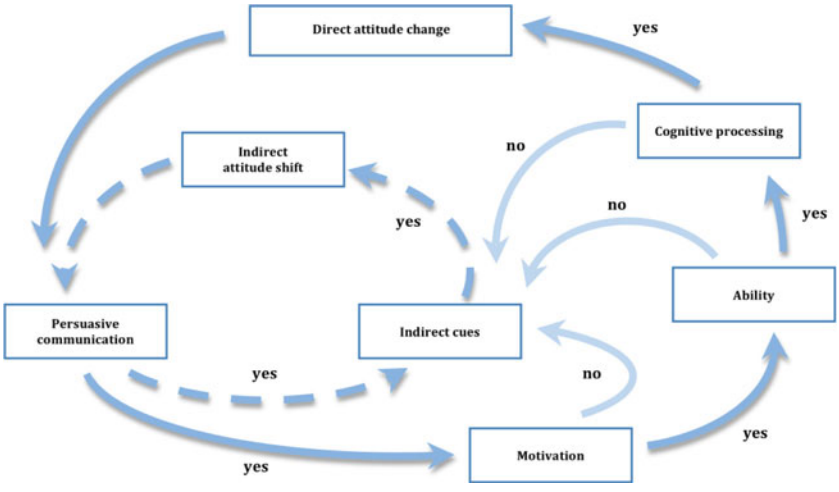

Fig. 2. Two key spirals in the Elaboration Likelihood Model

process"), social support could provide the means to motivate the users to process the message (yes-arrow from motivation to ability), primary task support should strengthen the ability to do this, and credibility support could best match the indirect cue (i.e., it seems to be better suited for an indirect attitude shift rather than change).

Support for the nature of cognitive processing and cognitive structure change seem to be very hard to achieve. In fact, this is the only yes-arrow from the ELM that the PSD model does not directly support. Yet, support for cognitive processing is not only difficult in a persuasion context. Similar problems do exist also in the context of knowledge work [13], where designed solutions can at the same time both support and hamper cognitive performance.

The ELM and the PSD model can be utilized together to achieve attitude change. Similarly, in order to design persuasive systems that aim at producing behavior change, we should apply the PSD model together with behavior change theories and models. For example, the primary task category could be used to make target behavior simpler, thus increasing the chances for behavior change [14]. This would probably lead into gaining further benefits from the PSD model.

6 Discussion

This paper provided a case description of applying the PSD framework to identify the (support for) persuasive features on a mobile Internet device. The main findings of this study were: i) the case demonstrated how the various PSD principles work together, ii) applying the PSD framework as a tool to evaluate the persuasive features of the system (based on software design specifications) is feasible. In addition, our results confirm that trying to utilize all the persuasion techniques in a single case is redundant. Rather, a coherent set (matching the persuasion context) of the techniques

should be considered and conveyed. This should be done in such a manner that there would be as much synergy as possible between the different techniques included.

With mobile Internet devices, the primary task can vary greatly from surfing the Web, social networking, playing games and using the device as a mobile phone. This made the analysis of persuasiveness concerning the primary task issues somewhat more difficult than with more defined goals for the device. On the other hand, analyzing dialogue support was rather straightforward, as the dialogue support category deals mostly with system-to-user interaction issues, which were described in detail in the design specifications.

Behavioral modeling is an important part of requirements engineering [7]. With behavioral modeling, we can also identify opportunities for behavior change. In this case, however, the persuasive features were not necessarily directly linked with behavior change *per se*. Most of the persuasive system features (or the support for them) found in the design specifications seemed to enhance the user experience, i.e., boosting the ease of use and usefulness of the device. For instance, reduction (reducing complex behavior into simple tasks) decreases the effort that is required of the users to perform their target behavior.

In this particular context, analyzing persuasive design is a demanding interpretative task. As there are no explicit evaluation heuristics defined for the PSD model yet, there lies a potential bias in the interpretation of the specifications. Nevertheless, in extracting and categorizing persuasive features, we rigorously observed if the specifications clearly included the described variables. Obviously, the specifications did not always follow the very same terminology as found in the PSD model, and thus the analysis was mostly based on interpretive categorization. In spite of its wide coverage, the PSD model still is not an exhaustive list of persuasive features. New persuasion techniques may be identified in the future. The model has been built in such a manner that it may evolve, but even as it stands now, it is an important tool for any persuasive system developer.

7 Conclusions

Based on this case, the PSD model seems to fit relatively well for evaluating software design specifications. Practitioners who have no theoretical background can apply the PSD model to increase the persuasiveness of the systems they design. The greatest benefit of the PSD model for researchers designing new systems may be achieved when it is applied together with a sound theory, such as the Elaboration Likelihood Model. Using the ELM together with the PSD model, one may increase the chances for attitude change. The only part in the ELM that the PSD model does not directly support is the use of the direct route for achieving a positive or negative attitude change.

The level of details of the software design specification plays an important role in the evaluation process. Evaluating highly detailed specification documents enables more reliable and relevant investigation of the persuasive features than evaluating a specification document that is on a very high level of abstraction.

As the evaluation of specifications and/or applications may be quite laborious and time consuming, the PSD model would benefit from explicit heuristics defined for

each of its design principles. Also, a clearer distinction between the design principles in the social support category would make them easier to use. Additional persuasive features may be identified from practice.

A potential future research area is supporting cognitive processing and cognitive structural change. Additionally, a very interesting and multidisciplinary line of research is how to design, implement and evaluate (persuasive) behavior change support systems for a multitude of domains such as healthcare and education.

Acknowledgements

We wish to thank the Graduate School on Software Systems and Engineering (SoSE), the National Technology Agency of Finland (Tekes) and the RichWeb project for funding this research.

References

1. Oinas-Kukkonen, H., Harjumaa, M.: Persuasive Systems Design: Key Issues, Process Model, and System Features. Communications of the Association for Information Systems 24, 28 (2009)
2. Petty, R.E., Cacioppo, J.T.: Communication and Persuasion: Central and Peripheral Routes to Attitude Change. Springer, Heidelberg (1986)
3. Fishbein, M., Ajzen, I.: Belief, attitude, intention, and behavior: An introduction to theory and research. Addison-Wesley, Reading (1975)
4. Ajzen, I.: The Theory of Planned Behavior. Organ. Behav. Hum. Decis. Process. 50, 179–211 (1991)
5. Bandura, A.: Social Cognitive Theory of Self-Regulation. Organ. Behav. Hum. Decis. Process. 50, 248–287 (1991)
6. Zave, P.: Classification of Research Efforts in Requirements Engineering. ACM Computing Surveys 29(4), 315–321 (1997)
7. Nuseibeh, B., Easterbrook, S.: Requirements engineering: a roadmap. In: Proceedings of the Conference on The Future of Software Engineering table of contents, Limerick, Ireland, pp. 35–46 (2000)
8. Oinas-Kukkonen, H.: Behavior change support systems: The next frontier for web science. In: The Second Web Science Conference, Raleigh, NC, USA, April 26-27 (2010)
9. Nunamaker Jr., J.F., Chen, M., Purdin, T.D.M.: Systems Development in Information Systems Research. J. Manage. Inf. Syst. 7, 89–106 (1990)
10. Bourque, P., Dupuis, R., Abran, A., et al.: The Guide to the Software Engineering Body of Knowledge. IEEE Software 16, 35–44 (1999)
11. Dey, A.K.: Understanding and using Context. Personal and Ubiquitous Computing 5, 4–7 (2001)
12. Tam, K.Y., Ho, S.Y.: Web Personalization as a Persuasion Strategy: An Elaboration Likelihood Model Perspective. Inform. Syst. Res. 16, 271–291 (2005)
13. Nakakoji, K., Yamamoto, Y., Akaishi, M., et al.: Interaction Design for Scholarly Writing: Hypertext Representations as a Means for Creative Knowledge Work. New Review of Hypermedia and Multimedia 11, 39–67 (2005)
14. Fogg, B.J.: A Behavior Model for Persuasive Design. In: Persuasive 2009, Claremont, California, USA, April 26-29 (2009)