Towards a Non-Functional Requirements Discovery Approach for Persuasive Systems

Nelly Condori-Fernandez
Universidade da Coruna, Spain
Vrije Universiteit Amsterdam
Amsterdam, The Netherlands
n.condori.fernandez@udc.es
n.condori-fernandez@vu.nl

Joao Araujo
Universidade Nova de Lisboa
Portugal
p191@fct.unl.pt

Alejandro Catala
Centro Singular de Investigacion en Tecnoloxias Intelixentes
Universidade de Santiago de Compostela, Spain
alejandro.catala@usc.es

Patricia Lago
Vrije Universiteit Amsterdam, The Netherlands
p.lago@vu.nl

ABSTRACT

A number of software systems that attempt to help people achieve behavior change have been proposed in various domains such as health and wellness. However, sometimes, such systems have failed to provide a satisfactory or sustainable User Experience (UX), as it is observed when users may be reluctant to respond to the activation of the systems' changing demands. Moreover, a negative User Experience (UX) can be exposed by Behavior Change Support Systems (BCSS) if designers do not have clear understanding of the requirements that factually help changing the user behavior that accomplishes a sustainability goal. We first explored the Persuasive System Design (PSD) model that should be considered in UX assessment of BCSSs. Then, we propose a requirements discovery process that can be considered to re-design a software interactive system based on negative UX.

CCS CONCEPTS

• **Software and its engineering → Design software**; • **Human-centered computing→ User studies**

KEYWORDS

non-functional requirements, Sustainability, persuasive systems, UX assessment

ACM Reference format:

N. Condori-Fernández, J. Araujo, A. Catala, P. Lago. 2020. In *Proceedings of ACM SAC '20, March 30-April 3, 2020, Brno, Czech Republic© 2020.* Copyright is held by the owner/author(s). ACM ISBN 978-1-4503-6866-7/20/03. https://doi.org/10.1145/3341105.3374059

1 INTRODUCTION

A Behavior Change Support Systems (BCSS) is "an information system designed to form, alter or reinforce attitudes, behaviors or an act of complying without using deception, coercion or inducements" [1]. Behavior and behavior change are based on behavioral theories. Examples of popular theories or models are the Transtheoretical Model of behavior change [2] or the Goal-setting Theory [3]. However, despite advances from the field of psychology to understand user behavior according to the changing demands [4], users may be reluctant to use BCSSs. According to Kelders et al. [5], a negative user experience can be induced, if designers do not select the right persuasive strategies. Nevertheless, to find the right way of persuasion is important to have first a clear understanding of requirements that factually help in changing the user behavior to achieve sustainability goals. This can be accomplished by discovering, from their actual use, relevant requirements that should be considered to (re)design and maintain software interactive systems, and consequently contribute to a longer term usage.

We have found several approaches for discovering requirements such as scenario-based approaches [6][7][8], and user feedback-driven approaches [9],[10]. One of the disadvantages of the scenario-based approaches is that the effectiveness of discovering requirements depends on how the scenarios are represented; whereas user-feedback driven approaches, depending on the available online data (in app store reviews, blogs, forums, etc.), can be biased [11] and consequently give fake feedback to software designers.

In contrast to these approaches, our proposal in this paper lies on the importance of understanding how user experiences with a software product and gaining insight into user needs. In this direction, this paper is concerned with: (i) the theoretical and methodological aspects that should be considered in UX assessment of BCSSs; (ii) the discovery of user needs emerging from a UX assessment; (iii) and the importance of translating such user needs into NFR that must be addressed for improving UX.

We use the Persuasive System Design (PSD) model [12] as the theoretical framework for designing our empirical UX assessment and defining the requirements discovery process. With our approach we aim to increase awareness in designers/developers about NFRs that were discovered as a consequence of detecting negative UX.

The rest of this paper is organized as follows. Section 2 introduces the PSD model as a design instrument for BCSSs. The NFR discovery approach is presented in Section 3. Section 4 draws the conclusions and discusses future work.

2 BACKGROUND: THE PSD MODEL

In this paper, we consider the PSD model [12], as the theoretical framework for our research. The PSD model is a recent conceptualization for designing, developing and evaluating persuasive systems. It consists of the premises behind any persuasive system, the persuasion context and the persuasive software system features. All BCSSs are based on the following premises [12],[13]:

- P1: *Useful*. The system really serves the needs of the user.
- P2: *User-friendly*. The system should be easy to use or dealt with.
- P3: *Unobtrusiveness*. The system should avoid being disturbing while the user is performing tasks.
- P4: *Open*. Designers should make the ideas and the goals of persuasion transparent.
- P5: *Cognitive Consistency*. People like their views about the world to be organized and consistent. Inconsistency disturbs people, and they easily want to reorganize their thinking and restore consistency, perhaps even feel obliged to do so.
- P6: *Incremental*. This means that persuasion goes stepwise, and all steps contribute to the goals to be realized.
- P7: *Information Technology (IT) is never neutral*. IT always influences attitudes and behavior.
- P8: *Direct and indirect routes*. Persuasion strategies can be divided into direct and indirect routes, and persuasion will depend on the ability and motivation of people to process information.

The analysis of the persuasion context consists of looking into (1) the intent, (2) the event and (3) the strategy. The event comprises the use situation, user's characteristics, technological platform and environment. The strategy includes the message itself and the route to be used to achieve a goal.

The PSD model describes persuasive software system features grouped in four categories: i) primary activity support, ii) dialogue support, iii) perceived system credibility and iv) social support[12]. The primary activity support category focuses on supporting the activities that lead to achievement of the BCSS goals. Dialogue support refers to techniques/mechanisms to motivate users to use BCSS. The credibility category relates to how to design a system so that it is more credible and thereby more persuasive. The social influence category describes how to design the system so that it motivates users by leveraging different aspects of social influence.

We argue that NFRs can emerge from a UX assessment of BCSSs. The following section presents our NFR discovery proposal.

3 NON-FUNCTIONAL REQUIREMENTS DISCOVERY BASED ON NEGATIVE UX

As shown in Figure 1, the requirements discovery lies on the importance of understanding the negative UX to gain more insight into useful user feedback, expressed through observable attitudes and behavior. Next we describe firstly the UX assessment based on attitudinal data, then the requirements discovering process, and finally the potential threats to validity.

3.1 UX assessment

It is supported by a wide range of research methods available, ranging from *attitudinal evaluations* (e.g. UX questionnaire, think-aloud) to *behavioral evaluations* (e.g. eye-tracking). In this phase, in contrast to Sonnleitner et al [15], we focus on negative User Experience (NUX) that is caused by the lack of fulfillment of needs during the interaction with a software product (e.g. BCSS). The effect of this cause impacts on the user attitudes ("what people say") and user behaviors ("what people do").

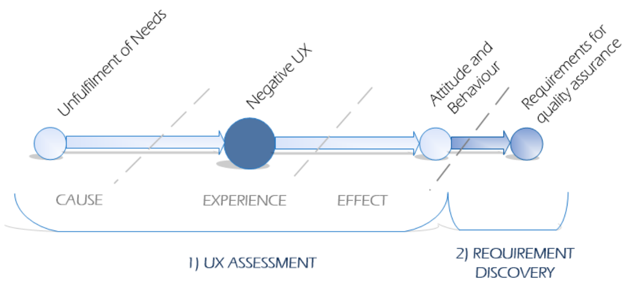

Figure 1. Requirements discovery based on Negative UX [14]

3.2 NFR discovery

In this sub-section we present a discovery process that uses the PSD model as a means to identify non-functional requirements (NFR) that should be addressed by a persuasive system. As shown in Figure 2, our approach consists of three steps: identifying unfulfilled user needs, mapping to the PSD premises, and categorizing user feedback to the PSD categories.

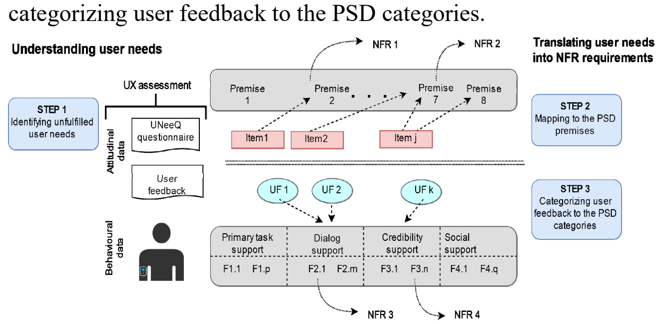

Figure 2. NFR discovery approach

Step 1: Identifying unfulfilled user needs. By analyzing the results of the User Needs Questionnaire (UNeeQ) questionnaire [15] conducted in different moments of the user study, the extent of fulfillment of our participants' needs is determined. We consider only those items of the questionnaire that get a low score.

Step 2: Mapping to PSD model premises affected by the negative user experience: By doing a first mapping between the PSD model premises and corresponding items identified in the first step, a first set of requirements are directly discovered by translating the corresponding premises (e.g., P1-P5) in NFRs. Some examples of these NFR requirements are: usefulness (P1), cognitive consistency (P5), unobtrusiveness (P3), fun and enjoyment(P2), and trustworthiness (P4).

Step 3: Categorizing user feedback: Content analysis is carried out to categorize the user feedback (UF) according to the four categories described in the PSD model (i.e., dialogue support, primary activity support, perceived credibility, social influence), which correspond to the persuasive software systems features identified by Oinas-Kukkonen [12].

The **first category** corresponds to the **dialogue support** feature. The requirements that can be derived from this category, such as awareness, engagement and cognitive consistency, would enable BCSS to provide relevant, motivating and adequate feedback to its users.

The **second category** consists of requirements that contribute to **primary activity support** features, which would enable the achievement of the BCSS goal. For example, tailorability, adaptability and learnability are important requirements that must be considered. The **third category, credibility,** trustworthiness and transparency of the system are examples of NFR that can be discovered based on the UX results. Finally, for the **social support category,** it consists of requirements that can be derived from the following features: social learning, social comparison, normative influence, social facilitation, cooperation, competition, recognition. This third step enables us to discover NFR that are directly related to these four type of features of the PSD model.

Therefore, the requirements are discovered, by i) understanding user needs that were not fulfilled (gathered by the UNeeQ questionnaire) and what people would like to have (user comments, suggestions); and ii) translating these user needs into NFR requirements.

3.3 Threats to validity

This section discusses the potential issues that may threaten the construct, internal and external validity.

Construct validity. *Mono-method bias*. Using a single type of measures or observations involves a risk. In our approach, different measures are collected (e.g., UNeeQ questionnaire, overall positive/negative UX).

Inadequate pre-operational explication of constructs. This threat occurs when the constructs are not adequately defined. As we focus on the UX assessment of BCSS, we mitigated this threat by considering the theoretical framework, named PSD model. It allows us to operationalize the UneeQ questionnaire, which was also validated by the original authors [15].

Internal validity. *Mortality*. As the user studies with BCSS are long-term and require to be conducted in a real setting with a null control, the likelihood of having dropouts is high. Our recommendation to reduce this threat is to inform to the potential participants beforehand about the length and estimated effort of the study. Moreover, reminders should be considered to meet the deadlines for completing some tasks (e.g., fill questionnaires).

External validity. *Interaction of setting and treatment*. This threat should be mitigated by conducting the user study in their natural environment.

4 CONCLUSIONS

In this paper we have firstly introduced the PSD model as a means for instrumenting the UX assessment and defining the requirements discovery process. Secondly, we list some potential threats that might threaten the validity of the results using our approach. It highlights the importance of UX assessment as a way to discover user needs and translate them into requirements. Therefore, we also contribute to the body of knowledge about the incorporation of UX into Requirements Engineering, which is not a widely practice yet as pointed out in [16].

As future work, we will evaluate the applicability of our approach in specific existing BCSSs from different domains.

ACKNOWLEDGEMENT

This work has received partial support by the projects FEDER-UE CSI ED431G/01, ED43C 2018/29, accreditation 2016-2019 ED431G/08, KUSISQA supported by N° 014-2019-FONDECYT-BM-INC.INV, NOVA LINCS UID/CEC/04516/2019, MICINN IJC2018-037522-I, RTI2018-099646-B-I00.

REFERENCES

[1] H. Oinas-Kukkonen. 2010. Behavior change support systems: A research model and agenda. In Persuasive Technology. Springer, 4–14.

[2] A E., McConnaughy, J. O Prochaska, and W. F Velicer. Stages of change in psychotherapy: Measurement and sample profiles. Psychotherapy: Theory, Research & Practice, 20(3):368, 1983.

[3] E. A Locke, D.-Ok Chah, S. Harrison, and N.Lustgarten. Separating the effects of goal specificity from goal level. Organizational Behavior and Human Decision Processes, 43(2):270 – 287, 1989.

[4] B. Fogg, "A behavior model for persuasive design," Proc. 4th Int. Conf. Persuasive Technology. - Persuasive. '09, p. 1, 2009.

[5] S. Kelders, O. Kulyk, L. Van Gemert-pijnen, and H. Oinas kukkonen, "Selecting effective persuasive strategies in Behavior Change Support Systems," Proc. Third Int. Work. Behav. Chang. Support Syst. co-located with 10th Int. Conf. Persuasive Technology. (PERSUASIVE 2015), no. Bcss, pp. 1–6, 2015.

[6] A. Sutcliffe, P. Rayson, C. N. Bull, & P. Sawyer (2014). Discovering affect-laden requirements to achieve system acceptance. In 2014 IEEE 22nd International Requirements Engineering Conference, RE 2014 - Proceedings (pp. 173–182).

[7] K. Zachos, N. Maiden, & A. Tosar. (2005). Rich-media scenarios for discovering requirements. IEEE Software, 22(5), 89–97. https://doi.org/10.1109/MS.2005.134

[8] A. Gregoriades, J. Hadjicosti, C. Florides, M. Pampaka, & H. Michail (2015). A driving simulator for discovering requirements in complex systems. In Simulation Series (Vol. 47, pp. 17–26).

[9] I. Morales-Ramirez, F. M. Kifetew, & A. Perini, (2017). Analysis of online discussions in support of requirements discovery. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 10253 LNCS). https://doi.org/10.1007/978-3-319-59536-8_11

[10] Z. Kurtanović and W. Maalej, "Automatically Classifying Functional and Non- functional Requirements Using Supervised Machine Learning," 2017 IEEE 25th International Requirements Engineering Conference (RE), Lisbon, 2017, pp. 490-495.

[11] R. Baeza-Yates. 2018. Bias on the web. Commun. ACM 61, 6 (May 2018), 54-61. DOI: https://doi.org/10.1145/3209581

[12] H. Oinas-Kukkonen and M. Harjumaa, "Persuasive systems design: Key issues, process model, and system features," Commun. Assoc. Inf. Syst., vol. 24, no. 1, 2009.

[13] H. Oinas-Kukkonen, 2013.Afoundation for the study of behavior change support systems. Personal and Ubiquitous Computing, 17 (6), 1223–1235.

[14] N. Condori-Fernandez, A. Catala, and P. Lago. 2018. Discovering requirements of behaviour change software systems from negative user experience. In Proceedings of the 40th International Conference on Software Engineering: Companion (ICSE '18). ACM, New York, NY, USA, 222-223.

[15] A. Sonneleitner, M. Pawlowski, T. K¨asser, M. Peissner. Experimentally Manipulating Positive User Experience Based on the Fulfilment of User Needs. 14th International Conference on Human-Computer Interaction (INTERACT), Sep 2013, Cape Town, South Africa. Springer, LNCS-8120 (Part IV), pp.555-562, 2013.

[16] S. Ebert, R. Humayoun, N. Seyff, A. Perini, and S. D. J. Barbosa, "Bridging the Gap Between Requirements Engineering and Human-Computer Interaction", Springer International Publishing, 2016, pp. 3–7.