Trust as a Sustainability Requirement

Baraa Zieni
University of Leicester
Leicester, UK
bz60@leicester.ac.uk

Ruzanna Chitchyan
University of Leicester
Leicester, UK
rc256@leicester.ac.uk

Reiko Heckel
University of Leicester
Leicester, UK
rh122@leicester.ac.uk

Abstract—**Trust is the key concern that underpins social sustainability. In this paper we provide a brief overview of trust from a number of perspectives (from security to customer relationship management), and present our take on trust as an interaction-based phenomenon.**

Index Terms—**trust, sustainability, trust requirements.**

I. INTRODUCTION

Klaus Pohl states that "Requirements engineering is the process of eliciting individual stakeholder requirements and needs and developing them into detailed, agreed requirements documented and specified in such a way that they can serve as the basis for all other system development activities" [1]. Inability to identify the relevant requirements, or to keep up with changing requirements is the key reason for software project failures [2]. Furthermore, since it is in requirements that the core focus, functions, and constraints of the software system-to-be are defined, Requirements Engineering has also a key role to play in developing software that would foster sustainability [3].

Sustainability is often defined as the state whereby the humankind can "meet the needs of the present without compromising the ability of future generations to satisfy their own needs" [4]. The requirements engineering research has interpreted this broader definition as an objective for sustainability inducing software systems to continuously support the combined set of sustainability dimensions (i.e., environmental, economic, personal, social, and technical) [5], [6].

In this work we focus on the social dimension of sustainability, specifically on its trust requirements. As noted by R. Goodland [7], "Social sustainability [...] create[s] the basic framework for society. It lowers the cost of working together and facilitates cooperation: **trust lowers transaction costs**."

But what exactly is trust? And how does it lower transaction costs? And what requirements do we need to "elicit, document, and agree" so that the resultant software system promotes social sustainability through enhanced trust? These are the kinds of questions that we hope to address in our research. This paper provides the very brief summary of related work on trust (in Section 2), and an overview of our initial thoughts and research direction (in Section 3).

II. RELATED WORK

The topic of trust has been studied in many sciences and from many perspectives. We will discuss some of those that we consider relevant to our work below (though we note that other relevant areas, such as group and social psychology are missing here).

A. Trust in Security

Many researchers have examined the notion of trust in software systems as closely aligned with security or privacy. Some have focused on security of the system as an artefact, while others focus on the human aspects of security. For instance Elahi [8] suggests that the more trust there is between the end users of the software system and other stakeholders, the less security they need (e.g., if there is no perceived risk or loss - i.e., mistrust - there is no perceived need for protections). The perception of security of a software system is a mere prerequisite for the initiation and maintenance of trust towards it. Security does not guaranty trust towards a software system, but it is important to make it trustworthy.

B. Trust in (Customer) Relationships

Trust in (customer) relationship management can be divided into two categories: the initial trust and the ongoing trust.

Initial trust involves the willingness to trust the other party without having a prior experience or knowledge of its background [9]. Ongoing trust is dynamic and relies on actual experiences and interactions between two parties [9]. In either case, trust is one of the key foundations that facilitates relationships. Brown [10] suggests that trust in relationships can be built over time via very small actions. Thus, when considering changes of trust over time, it is important to distinguish between the initial perspective and ongoing trust to better highlight the evolving and changing nature of the notion of trust.

Trust seems to play a crucial role in reducing users' uncertainty. Damian-Reyes and colleagues suggest that it is one of the factors that can affect user confidence in a software system [11]. Previous studies reported trust as behavioural intention which can affect vulnerability and uncertainty [12]. Similarly, Ruohomaa et. al [13] discussed trust as a key tool which helps the end users cope with the uncertainty in making a decision.

C. Trust Factors

Li and colleagues [14] report that the initial trust is affected by the perceived social pressure (to perform or not in accordance with some "subjective norms"), the cognitive reputation, calculative, and organisational situational normality based factors. They also observe that individual's personality

or the technology did not substantially affect the trusting beliefs.

The factors that affect trust were studied specifically for the case of new software systems. The authors have grouped these factors into several "trust categories", which include:

- Personality-based trust;
- Cognition-based trust (or reputation);
- Institutional-based trust (structural assurance);
- Information technology, which includes factors such as security, privacy, and general online experiences;
- Social factors, such as national culture;
- Diffusion of innovation: as users initially receive some information about an innovation and its advantages and disadvantages, this forms their initial attitude toward the innovation and influences subsequent adoption decisions.

The specific mix and selection of the listed factors for each individual depends on their characteristics (e.g., no prior experience in IT systems for business), and context characteristics (e.g., national culture of the Jordanian context).

III. RESEARCH APPROACH AND OUTLOOK

As outlined in the previous section, a number of interpretations and factors of trust have been investigated. The present work is focused on operationalising the notion of trust into software requirements in order to inform trusted software systems design. For this we first need to establish the scope of the notion of trust for this work. Drawing on the previous research we observe that:

- Trust is a **relationship**. Although some key characteristics (such as security, an individual's willingness and aptitude to trust, social conventions, etc.) are essential for the initiation of the relationship, it also **requires interaction** between the involved entities.
- Trust is **dynamic**. As any relationship that involves humans, a trust relationship is **subject to continuous change**. The change is driven by feedback from the interaction whose results are evaluated by the participants and, where considered relevant, contribute to building up or eroding the relationship.
- Trust is **cumulative**. While a single result from a specific interaction may not have substantial effect on the trust relationship, repeated similar results are likely to have a defined cumulative effect (e.g., if an employee is late for one of the meetings, (s)he is likely to be excused; but if (s)he is repeatedly late, (s)he is likely to gain an "always late" reputation).

Thus, we suggest that *trust is a relationship which will change over time through the evaluation of relevant interactions by the participants in the relationship.*

Given the above interpretation of trust, our work aims to (i) elicit (specific and measurable) requirements that enable trustworthy software systems engineering; and (ii) define a trust evaluation model for measuring the current trust level within the given software system and its dynamics.

For this, we must first account for the initial trust relationship between the software system (or system-to-be) and its stakeholders. This will serve as the starting point of the trust relationship. The dynamic model of the relationship evolution and evaluation will then build upon the initial trust.

In light of this, we will work on building a trust model that:

1) Starts from eliciting the end-user requirements which support the socio-technical interactions between users and the system. The first set of such requirements will address the most frequently repeated requirements/services that the users ask for, and that (as per related published work) is considered essential on establishing the trust relationship between the stakeholders and the system.
2) Study how the socio-technical interactions are affected by various environments (e.g., product markets);
3) Study how such interactions are affected by specific stakeholder requirements and constraints (e.g., posed by developers, or the business owners).

In conclusion we would like to underline that some may suggest that trust requirements (possibly implicitly) are already covered in traditional RE, as trust towards a system is gained if the system i) functions as expected, ii) is efficient, iii) is reliable, iv) is usable, and v) is safe and secure. Yet, the key distinction of this work from any previously published RE work on topics that relate to trust is that we underline the need for continuous evolution in a trust model and requirements. All previous work has formulated static requirements, which, though may relate to trust, do not reflect its' dynamic nature.

REFERENCES

[1] K. Pohl, *Requirements Engineering - Fundamentals, Principles, and Techniques*. Springer, 2010.

[2] C. Ebert, "Episode 114: On requirements engineering," 2008. [Online]. Available: http://www.se-radio.net/2008/10/episode-114-christof-ebert-on-requirements-engineering/

[3] C. Becker, S. Betz, R. Chitchyan, L. Duboc, S. M. Easterbrook, B. Penzenstadler, N. Seyff, and C. C. Venters, "Requirements: The Key to Sustainability," *IEEE Software*, vol. 33, no. 1, pp. 56–65, Jan-Feb 2016.

[4] G. H. Brundtland and UN World Commission on Environment and Development, *Our common future*. Oxford University Press, 1987.

[5] B. Penzenstadler, V. Bauer, C. Calero, and X. Franch, "Sustainability in software engineering: A systematic literature review," in *16th International Conference on Evaluation & Assessment in Software Engineering, EASE 2012, Ciudad Real, Spain, May 14-15, 2012. Proceedings*, 2012, pp. 32–41.

[6] C. C. Venters, N. Seyff, C. Becker, S. Betz, R. Chitchyan, L. Duboc, D. McIntyre, and B. Penzenstadler, "Characterising sustainability requirements: A new species, red herring, or just an odd fish?" in *Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Society Track*, ser. ICSE-SEIS '17, 2017, pp. 3–12.

[7] R. Goodland, "Sustainability: Human, social, economic and environmental," 2002.

[8] G. Elahi and E. Yu, "Trust trade-off analysis for security requirements engineering," in *Requirements Engineering Conference, 2009. RE'09. 17th IEEE International*. IEEE, 2009, pp. 243–248.

[9] J.-N. Lee, M. Q. Huynh, and R. Hirschheim, "An integrative model of trust on it outsourcing: Examining a bilateral perspective," *Information Systems Frontiers*, vol. 10, no. 2, pp. 145–163, 2008.

[10] B. Brown, *Daring Greatly: How the Courage to Be Vulnerable Transforms the Way We Live, Love, Parent, and Lead*. NY: Gotham Books, 2012.

[11] P. Damián-Reyes, J. Favela, and J. Contreras-Castillo, "Uncertainty management in context-aware applications: Increasing usability and user trust," *Wireless Personal Communications*, vol. 56, no. 1, pp. 37–53, 2011.

[12] C. Moorman, G. Zaltman, and R. Deshpande, "Relationships between providers and users of market research: The dynamics of trust within and between organizations," *Journal of Marketing Research*, vol. 29, no. 3, pp. 314–328, 1992.

[13] S. Ruohomaa and L. Kutvonen, *Trust Management Survey*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 77–92.

[14] X. Li, T. J. Hess, and J. S. Valacich, "Why do we trust new technology? a study of initial trust formation with organizational information systems," *The Journal of Strategic Information Systems*, vol. 17, no. 1, pp. 39 – 71, 2008.