

Towards a Systematic Study of Values in SE: *Tools for Industry and Education*

Emily Winter, Stephen Forshaw, Lucy Hunt, and Maria Angela Ferrario
School of Computing and Communications (SCC) Lancaster University Lancaster, UK
{e.winter; s.w.forshaw; l.hunt1; m.a.ferrario}@lancaster.ac.uk

Abstract—This paper reports on the development of ongoing research into the systematic study of human values in software engineering (SE). Firstly, we outline the rationale for this work, highlighting the significance of human values in SE and the need for both a more structured approach to their study and tools to support their articulation in the SE domain. We then explain our theoretical framework, which draws on social psychology research on values. Secondly, we introduce tools designed in line with this framework and the context of their use so far. Our tools adopt a variety of research methods, including structured mixed-method techniques, such as the ‘Values Q-Sort’ (V-QS), and more open design-based methods, such as the ‘Values Probes’. Given the multi-faceted nature of the topic, we argue that a combination of qualitative, quantitative and design thinking techniques is a necessary step for the study of human values in SE; and that these methods should be both a) robust and replicable, and b) creative and provocative. Finally, we highlight the potential applications of our tools in SE industry and education.

Keywords—human values, design thinking, quantitative methods, qualitative methods, SE education, SE industry

I. INTRODUCTION

This role of human values in SE decision-making processes is widely acknowledged [10], but so too is the challenge of researching values in a systematic way [21]. This paper introduces a research framework developed to respond to this challenge and tools designed in accordance with this framework. These tools have been designed to be both robust and replicable, and creative and provocative. We argue that such a combination of quantitative, qualitative and design thinking approaches [9][15] is vital in the study of values. We conclude with our vision for how these tools can be utilized in SE industry and education, and present new, emerging work on uses and applications in these contexts.

II. RATIONAL AND RESEARCH FRAMEWORK

In light of the growing influence of software on society, SE professionals occupy positions of increased moral responsibility [28] and thus need greater support from the SE and wider research community. Our work responds to such need by creating space for the articulation and deliberation of human values within SE industry and education. Our rationale is threefold. Firstly, we view values research as a crucial response to the growing impact of software on society [3]. Secondly, we identify a pressing need to develop appropriate and theoretically informed methods to study values [10]. Thirdly, we develop tools that can assist values investigations in a way that is related to the chosen theoretical framework and relevant to the SE community [21]. The theoretical framework we adopt stems from social psychology, specifically the research of Schwartz and Maio [20][26][27].

Social psychology represents a key field in which values have been conceptualized, empirically tested and measured;

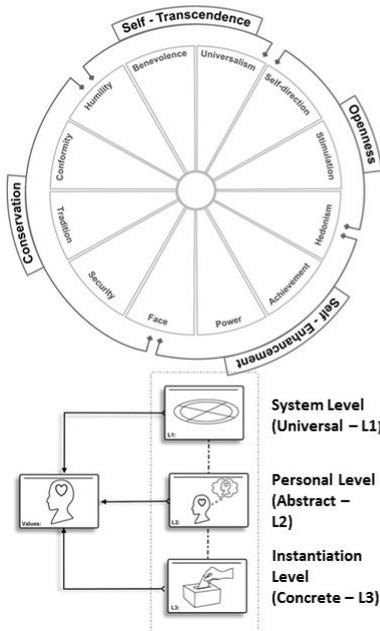
our research takes this body of work as its starting point. Schwartz offers a relational model of values, visually represented in circular form and operating through oppositional axes (Figure 1), such as the one juxtaposing self-enhancement values (e.g. achievement, power) vs. self-transcendence values (e.g. equality, concern for others). Empirical investigation into Schwartz’s model has shown that particular patterns of inter-relationships across values are common even in different cultural contexts [27].

Maio has expanded upon Schwartz’s work suggesting that values operate, and should be studied, at three levels: the system level - represented by Schwartz’s model of values relationships (L1); the abstract level - relating to personal interpretations and attitudes towards each value (L2); and the instantiation level - the actual behaviors driven by different values (L3) [20]. Within the SE context, we would expect certain patterns of values relationships (L1). For example, according to Schwartz’s model, software companies who most value commercial success may care less about their products’ environmental impact. At L2, different software engineers may have different interpretations of ‘public good’: for some, it may mean to develop secure software, for others to avoid working for ‘unethical’ businesses (e.g. the gambling industry). At L3, a search engine company concerned with privacy may intentionally ensure they never log user query results. Understanding these patterns, interpretations and behaviors can bring to light values tensions and alignment within SE teams.

In accordance with this framework, we develop two broad categories of tools: the first favours traditional quantitative methods (e.g. the Values Survey) and less conventional yet still structured mixed-method techniques (e.g. the Values Q-Sort [29], or V-QS). Both offer statistically robust, replicable ways of studying human values in SE. The second category includes tools that are designed to provoke more open and creative thinking among software practitioners. We refer to these as ‘Values Probes’ inspired by Gaver’s cultural probes [15]. Combining traditional surveys, Q-statistics [29], qualitative methods, and design thinking techniques [9] is not common, particularly in SE. We argue that this represents a key strength of our approach and is a necessary response to the complexity of values research [21].

The outcome is a set of tools that support the study of values across the three levels identified by the research framework. The Values Survey collects quantitative data on values relations (L1); the V-QS allows for the extraction of statistically generated SE practitioners’ types [29] (L1), but also the collection of semi-structured qualitative data on personal values perceptions (L2, L3); the Values Probes facilitate the elicitation of open-ended values narratives (L2, L3). Finally, we have just started exploring data-intensive approaches for automated values extraction from online corporate documentation and user generated data. Fig.1 outlines the tools developed, their aims, stage of development, how they map onto the three study levels, and

THEORETICAL FRAMEWORKS GUIDING TOOL DESIGN: Schwartz's Values Model, [30] (top) and the 3-Level Framework [24] (bottom)



TOOLS DEVELOPED: Aims, use so far, stage of development, values level

Tool	Aim and background	Use so far and status	Values level
	VALUES SURVEY: To capture computing researchers' perceptions of the values (personal and organizational) driving digital production. Schwartz's values model [27] as used in [6]	N=151 (computing researchers)	System level (L1)
	VQ-S: To investigate SE practitioners' own values perceptions. To provide insight at each of the three values levels as in [20]. Framed around ACM Code of Ethics [1] and Schwartz's values model [26]	N=12 (SE professionals) Adapted for HE use and used with 16 students	System level, personal level and instantiation level (L1, L2, L3)
	AI VQ-S: To explore values perceptions on AI. To explore use of V-QS in team settings. Framed around Asilomar AI Principles [14] and Schwartz's values model [26]	N=60 (HE students)	System level, personal level and instantiation level (L1, L2, L3)
	SOFTWARE WALLET: To capture perceptions of software "traits" through personification. Influenced by work on cultural probes [15] and speculative design [9]	N=5 (trial with computing researchers) N=30 (adapted for team context)	Personal level and instantiation level (L2, L3)
	SOFTWARE STARMAP: To capture perceptions of software "traits" through personification. Influenced by work on cultural probes [15] and Big 5 personality research [8]	Prototype stage	Personal level and instantiation level (L2, L3)
	AI PHYSICAL PROTOTYPES: AI values elicitation through future scenario reflection. Influenced by [11] and [15]	N=80 (HE students)	Personal level and instantiation level (L2, L3)
	DATA INTENSIVE TECHNIQUES: automatic extraction of corporate values orientations by mining on-line documentation and social media data	Under development	System level and instantiation level (L1, L2)

Fig. 1. Theoretical framework and tools

their use so far. Collectively, our tools have been used with more than two hundred participants ($N > 200$) from education, research and industry. Whilst we have published emerging results from the Values Q-Sort [30], this is the first paper to bring together all the tools we have developed and to reflect on the significance, further potential and future directions of our research for SE industry and education.

III. RELATED WORK

Our research is situated within a large body of work on human factors in SE that includes studies of innovative and risky software engineering behavior [7], and software engineering success [23]. However, unlike much of this work, our own concern with human factors has different implications to organizational efficiency and success. Rather, our work contributes to a growing research field defined by concern with human values in computing [13], particularly in SE [10][28]. Although our work recognises the importance and the role of morals and ethics in the SE profession [1], our research is distinguished from this work in that it focuses upon values conceptualized more broadly. For example, the very influential field of Values Sensitive Design (VSD) [13] has been critiqued for "cultivating a dogmatic response with respect to which values are worthy of consideration and disengaging from a commitment to understanding the nuanced manifestation of a plurality of values" [18]. Accordingly, we adopt a theoretical framework that considers a broader range of values than just those of "ethical import" [13].

IV. DEVELOPMENT OF TOOLS

Our key focus has been to develop and use appropriate tools for studying the role of values in SE. Here we briefly discuss these tools and highlight emerging findings.

A. Methodological Frameworks

The development of our tools has been influenced by two key methodological standpoints. Firstly, we argue that there is a need for robust, replicable methods to study values

[10][20]. Secondly, neither quantitative nor qualitative methods nor design-based thinking alone are enough to study a subject as complex as values. Statistics, mixed methods, conventional qualitative techniques, and creative, participatory tools all have a role to play. In summary, we suggest that values research in SE should be both a) robust and replicable and b) creative and provocative. Responding to (a), we developed a Values Survey that utilizes Schwartz's framework and used it with computing research professionals ($N=151$) in order to gain understanding of their values perceptions. The survey found general agreement that values shape technology, but a lack of clarity around which values do so, and how.

The V-QS was designed and used with software practitioners ($N=12$) to address these limitations and to gain greater understanding of how values work. The V-QS uses Q-methodology, an established mixed method that produces both quantitative and qualitative data and is a robust and replicable means for measuring subjectivity [29]. During a Q-Sort, participants rank a series of statements on a grid according to their level of agreement with them. Multiple Q-Sorts can be statistically analyzed, while the accompanying semi-structured interviews generate qualitative data. The V-QS was developed by mapping the 25 ACM Code of Ethics principles [1] to Schwartz's 19 values [26]. The ethics code provided a body of language relatable for SE practitioners. The Code was dual-coded, with 80% agreement between the two researchers; areas of discord were discussed, and value duplicates removed. In response to (b), we have developed a series of Values Probes inspired by 'cultural probes' [15], speculative design [9], and the use of physical artefacts in SE [4][11][22]. The Values Probes aim to elicit more open conversation than the V-QS, operating as provocations that try to "shift current perceptions of technology functionally, aesthetically, culturally, and even politically" [15].

B. Empirical Usage and Indicative Findings

Table 1 indicates current usage of these tools, ranging from extensive to more explorative. Whilst the Values

TABLE 1- PAST AND POTENTIAL USES OF TOOLS

Tool	Used So Far	+ Strengths - Weaknesses	Use (Potential/Used so far)	
			SE industry	SE education
Values Q-Sort	Industry SE; freelance developers; technical support; research SE. (N=12) Software Studio Undergraduate 2 nd year students (N=19)	+ Mixed method- provides both qualitative and quantitative data + Replicable + Works with small samples + Structured exercise that is appealing to software engineers - Difficult to keep conversation rooted in a specific SE project - Doesn't provide much insight at behavioral level (L3) - Conversation led by nature of statements	Project start, mid, end; both individual and team VQ-S. Articulation / deliberation of team members' values differences. 'Types' extraction may help with team composition – e.g. the 'Fun-loving, Extrinsically-motivated Developer' may work well with the 'Autonomous, Nonconforming Risk-taker' if piloting highly innovative project ideas.	Project start, mid, end; both individual and team VQ-S. Role play, articulation of team members' values differences. Reflection <i>on action</i> and <i>in action</i> [25]. For example, the V-QS has been used to articulate group project values, which were then included in the project brief. There has been ongoing class discussion of trade-offs between values and design choices.
Software Persona Wallet	Research Pilot (N=5) Business workshop (N=15)	+ Non-prescriptive and open to values as they emerge [18] - Current prototype is time-consuming to produce at scale	Various stages, i.e. at the start of systems requirements identification. Used in workshop groupwork to identify desirable personal characteristics of assistive technology professionals	As part of an interactive values seminar or workshop. Take home personal reflection piece. Piloted with a small team of researchers and PhD students to articulate personal viewpoints on software being developed
Software Personality Star map	Prototype stage; Research Pilot (N=3)	+ Quick to make and use + Open and provocative - Work needed to evaluate prototype further	Project 'Pre-mortems' for risk identification: what type of software could this become? It could also be used for comparing how software may be perceived by developers and by end-users in early user evaluations.	As part of an interactive values seminar or workshop. Take home personal reflection piece. Similar to industry, the star map could be used for comparing students' perceptions of their system with end-users' perceptions.

Probes are in the piloting stage, the results of the Values Survey and the V-QS have both been analyzed, and emerging results from the VQ-S are reported in [30]. The Values Survey established a strong basis for the importance of this research with the majority of respondents (92%) thinking that values shape technology outputs with 53.6% indicating “very much so” and 38.4% “to some extent”. The V-QS results from our first 12 participants were analyzed statistically, yielding three “types” or “factors”, i.e. “patterns of similarity in the Q-Sort configurations produced” [29]. These are indicative of the mental relationships between values (L1) held by SE practitioners and represent three types of software engineer that emerged from our study, namely: the Intrinsically-motivated, Socially-concerned Engineer; the Autonomous, Nonconforming Risk-taker; and the Fun-loving, Extrinsically-motivated Developer [30]. Qualitative data from the V-QS provides insight into the interpretations of different values (L2) and, to a more limited extent, their instantiation (L3). Detailed analysis of both the V-QS and survey results is beyond the scope of this paper.

V. APPLYING THE TOOLS IN INDUSTRY AND EDUCATION

From the development and use of these tools, we offer the following vision for their application in SE industry and education. Table 1 indicates the potential use of different tools in both sectors at different stages of SE projects. Whilst these are distinct sectors, they also interact with and influence each other, with SE students being exposed, through placements for example, to different industry cultures, and the future of the SE industry being shaped by the pedagogical experiences of today's students.

A. SE Industry

Miller and Larson have identified values articulation in industry as a challenge, arguing that software practitioners

find it difficult to express “ideas about human values with language that is not as precise or articulate as the language routinely used to express technical ideas” [21]. The V-QS, as a systematic exercise grounded in language generated by the computing profession, offers an ideal entry point for encouraging values articulation in industry. The further challenge of encouraging industry to embrace more design-based approaches has also been acknowledged [24]. Nonetheless, there are examples of complex decision-making processes in industry being aided by design-thinking, such as a game designed to help explore and inform decision-making processes in the domain of cyber security [11]. Future work includes using different tools at different stages of the software development lifecycle (see Table 1).

The aim of such collaboration is to support software practitioners with tools and methods to map and reflect upon their personal and team values and how they manifest, or not, in the systems developed. In practice, for each Q-Sort statement, a number of ‘how-to’ practical examples could be elicited and documented. In algorithms deciding which job adverts to show, for example, a good practice associated with the V-QS statement “It is important to me that I do not discriminate against others when developing software” [1][30] would be to ensure there is no gender bias influencing which adverts are shown, resulting in a potential trade-off with cost effectiveness [17]. This corresponds with Hussain et al.'s suggestion [16] to maintain “an up-to-date library of the valufied design patterns” and incorporate lessons learned from the real-world adoption of such patterns.

B. SE Education

The need to provide SE students with a practice-based reflective education has been widely identified [2][19]. Cervantes et al., in their work on a game developed to teach

SE students architecture design, identify that software architecture courses “frequently rely on relatively simple examples, with relatively little context, as these are easier to describe and convey” [4]. This critique also applies to values considerations [21]. Within this context, the Software Studio [19] is one of the most promising learning environments for adopting and adapting values tools in that it actively creates bridges between practice and a pedagogical reflective space by emphasizing “practical hands-on work and experimentation” [19]. For example, we used the V-QS with our Software Studio students (N=19) at the start of their project to help them articulate their project values, include them in their project briefs, and check their design decisions against them. One of the teams, who are developing a train journey planner, put the ‘not discriminating’ V-QS statement as one of their key priorities. They are now adopting a ‘most-vulnerable-first’ approach to system design and development, prioritizing search needs for people with disabilities, people with young children, and the elderly.

VI. CONCLUSION

Our research calls for an advance in systematically studying the role of human values in SE, and has focused on developing appropriate tools for this. Here we argue for a mix of quantitative, qualitative and design-thinking techniques for studying values and offer some suggestions as to how these tools might be further developed and used in SE industry and education. There is potential for digitising the tools to scale up access. However, this comes with challenges, particularly the potential misuse of tools, such as the V-QS, that rely on the elicitation of rich personal narratives and sampling restrictions [29]. We caution that the tools should be used to encourage discussion and awareness of values. In particular, though tempting, the V-QS results should not be over-generalized [29].

REFERENCES

- [1] ACM, ACM Code of Ethics and Professional Conduct, 2018. Retrieved 11 September 2018 from <https://www.acm.org/code-of-ethics>
- [2] M. Balaban and A. Sturm, “Software engineering lab: an essential component of a software engineering curriculum,” Proceedings of the 40th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET 2018), pp. 21-30. <https://doi.org/10.1145/3183377.3183395>
- [3] S. Betz and A. Fritsch, “A comparison of value sensitive design and sustainability design,” H. C. Mayr and M. Pinzger (eds.). Informatik. Bonn, Gesellschaft für Informatik, pp. 267-274, 2016.
- [4] B. Boehm and L. G. Huang, “Value-based software engineering,” *Computer*, vol. 36, no. 3, pp. 33–41, 2003.
- [5] H. Cervantes, S. Hazyjev, O. Hrytsay, and R. Kazman, “Smart decisions: an architectural design game,” In Proceedings of the 38th International Conference on Software Engineering (ICSE 2016), pp. 327-335. <http://doi.acm.org/10.1145/2889160.2889184>
- [6] Common Cause Foundation, *Perceptions Matter: The Common Cause UK Values Survey*, London: Common Cause Foundation, 2016.
- [7] K. C. Desouza and Y. Awazu, “Managing radical software engineers: between order and chaos,” Proceedings of the 2005 workshop on Human and social factors of software engineering (HSSE 2005), pp. 1-5. <https://doi.org/10.1145/1082983.1083005>
- [8] M. B. Donnellan, F. L. Oswald, B. M. Baird, and R. E. Lucas, “The mini-IPIP scales: tiny-yet-effective measures of the Big Five factors of personality,” *Psychological assessment*, vol. 18, no. 2, p.192, 2006.
- [9] A. Dunne, and F. Raby, *Speculative everything: design, fiction, and social dreaming*. MIT press, 2013.
- [10] M. A. Ferrario, W. Simm, S. Forshaw, A. Gradinar, M. T. Smith, and I. Smith, “Values-first SE: research principles in practice,” Proceedings of the 38th International Conference on Software Engineering (ICSE 2016), pp. 553-562. <http://dx.doi.org/10.1145/2889160.2889219>
- [11] S. Forshaw, L. Cruickshank, and A. Dix., 2012. Collaborative communication tools for designing: physical-cyber environments. In Proceedings of the 26th BCS Conference on Human Computer Interaction (HCI 2012), pp. 1-9
- [12] S. Frey, A. Rashid, P. Anthonysamy, M. Pinto-Albuquerque, and S. A. Naqvi, “The good, the bad and the ugly: a study of security decisions in a cyber-physical systems game,” Proceedings of the 40th International Conference on Software Engineering (ICSE 2018), pp. 496-506. <https://doi.org/10.1145/3180155.3182549>
- [13] B. Friedman, “Value-sensitive design,” *interactions*, vol. 3, no. 6, pp. 16–23, 1996.
- [14] Future of Life Institute, *Asilomar AI Principles*, 2017. Retrieved 24 September, 2018 from <https://futureoflife.org/ai-principles/>
- [15] B. Gaver, T. Dunne, and E. Pacenti, “Design: cultural probes,” *interactions*, vol. 6, no. 1 (January 1999), pp. 21-29.
- [16] W. Hussain, D. Mougouei, and Whittle, J., 2018. "Integrating social values into software design patterns." Proceedings of the ACM/IEEE International Workshop on Software Fairness (Fairware 2018), pp. 8-14. <https://doi.org/10.1145/3194770.3194777>
- [17] A. Lambrecht and Catherine E. Tucker, 2018. “Algorithmic bias? An empirical study into apparent gender-based discrimination in the display of STEM career ads”, *SRRN*, pp. 1-40. <http://dx.doi.org/10.2139/ssrn.2852260>
- [18] C. A. Le Dantec, E. S. Poole, and S. P. Wyche, “Values as lived experience: evolving value sensitive design in support of value discovery,” Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2009), pp. 1141-1150. <http://doi.acm.org/10.1145/1518701.1518875>
- [19] J. Lee, G. Kotonya, J. Whittle, and C. Bull, “Software design studio: a practical example,” Proceedings of the 37th International Conference on Software Engineering (ICSE 2015), pp. 389-397.
- [20] G. R. Maio, “Mental representations of social values,” *Advances in Experimental Social Psychology*, vol. 42, Academic Press, 2005, pp. 1-43.
- [21] K. W. Miller and D. K. Larson, “Agile software development: human values and culture,” *IEEE Technology and Society Magazine*, vol. 24, no. 4, pp. 36-42, 2005.
- [22] P. Newman, M. A. Ferrario, W. Simm, S. Forshaw, A. Friday, and J. Whittle, “The role of design thinking and physical prototyping in social software engineering,” Proceedings of the 37th International Conference on Software Engineering (ICSE 2015), Vol. 2. pp 487-496.
- [23] P. Ralph and P. Kelly, “The dimensions of software engineering success,” Proceedings of the 36th International Conference on Software Engineering (ICSE 2014), pp. 24-35. <http://doi.org/10.1145/2568225.2568261>
- [24] A. Rotondo and N. G. Freier, “The problem of defining values for design: a lack of common ground between industry and academia?” *Extended Abstracts on Human Factors in Computing Systems (CHI EA 2010)*, pp. 4183-4188. <doi>10.1145/1753846.1754123>
- [25] D. A. Schön, *Educating the reflective practitioner*, San Francisco: Jossey-Bass, 1987
- [26] S. H. Schwartz et al., “Refining the theory of basic individual values,” *Journal of personality and social psychology*, vol. 103, no.4, pp. 663-688, 2012.
- [27] S. H. Schwartz, “Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries,” in *Advances in experimental social psychology*, vol. 25, Academic Press, 1992, pp. 1-65.
- [28] J. Van den Hoven, G. J. Lokhorst, and I. Van de Poel, “Engineering and the problem of moral overload” *Science and engineering ethics*, vol. 18, no. 1, pp. 143–155, 2012.
- [29] S. Watts and P. Stenner, *Doing Q methodological research: Theory, method & interpretation*, Sage, 2012.
- [30] E. Winter, S. Forshaw, and M. A. Ferrario, “Measuring human values in software engineering”. Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2018), pp. 1-4, <https://doi.org/10.1145/323>