Guiding Socio-technical Reflection of Ethical Principles in TEL Software Development: The SREP Framework

Sebastian Dennerlein[1], Christof Wolf-Brenner[2], Robert Gutounig[3], Stefan Schweiger[4] and Viktoria Pammer-Schindler[1,2]

[1] Graz University of Technology, Graz Austria sdennerlein@tugraz.at
[2] Know-Center GmbH, Graz, Austria [cbrenner,vpammer]@know-center.at
[3] FH JOANNEUM, Graz, Austria robert.gutounig@fh-joanneum.at
[4] Bongfish GmbH, Graz, Austria schw.stefan@gmail.com

Abstract. In society and politics, there is a rising interest in considering ethical principles in technological innovation, especially in the intersection of education and technology. We propose a first iteration of a theory-derived framework to analyze ethical issues in technology-enhanced learning (TEL) software development. The framework understands ethical issues as an expression of the overall socio-technical system that are rooted in the interactions of human actors with technology, so-called socio-technical interactions (STIs). For guiding ethical reflection, the framework helps to explicate this human involvement, and to elicit discussions of ethical principles on these STIs. Prompts in the form of reflection questions can be inferred to reflect on the technology functionality from relevant human perspectives, and in relation to a list of fundamental ethical principles. We illustrate the framework and discuss its implications for TEL.

Keywords: Sociotechnical Systems, Ethical Principles, Reflection, Co-Design.

1 Introduction

Technological innovation has become the subject of regular social and political debates, even more so in the digital age with phenomena such as the Cambridge Analytica data scandal [1]. Special caution is required when technology intervenes in fundamental spheres of human life such as education. For example, developers of a TEL solution need to ask themselves whether a representative and bias-free data set was used to train the learning algorithm, so that undesirable effects such as the adverse treatment of students (e.g. gender and racial issues) are not intensified.

To avoid unethical consequences, careful evaluation regarding ethical principles is required. The creation of guidelines setting out ethical principles has become established as an approach to ensure ethical standards in software development in the field of AI. Among ethical principles derived in the guidelines, five have emerged as widely accepted: transparency, justice & fairness, non-maleficence, responsibility & accountability and privacy [2]. However, the mere proposal of ethical principles lacks concrete guidance in their application and rules for judgement [3], and software developers often

have to work out emerging ethical issues and solution approaches themselves. They need to learn from their experiences and (collaboratively) reflect to deal with ethical issues first-hand. This ethical reflection must be explicated and guided by frameworks.

Recent approaches started to explore the potential of reflection and design thinking to address ethical research questions, but none of them contains a structured explication of ethical issues and they miss the potential of aggregating meaningful reflection questions with high relevance for practice. For example, [4] suggest addressing ethical questions already during software design. In the TEL context, attempts are made to integrate ethical reflection in engineering education projects through peer ethic advisors [5].

This leads to our overall research question: *How to guide identifying and elaborating ethical problems in development and use of co-design based software development projects in TEL?* To answer the research question, we synthesized a conceptual framework based on research in ethics, socio-technical systems, co-design and reflection [6]. A literature review was conducted that specifies the research gap and makes conceptual relations explicit to inform the creation of the ethical framework in a second step.

2 Background

By **socio-technical system**, *we understand all reciprocal relationships between humans and technology* when it comes to the creation and use of technology [7]. Humans and technology are inextricably linked and we interact with technology in different roles in our private and working lives, both in the development and use: e.g. members of a development team usually take care of *data acquisition by defining an input mechanism, storage of the input in a database, design of functions like training algorithms and the output in form of a visualization, whereas users apply the service* to make decisions and facilitate work, and are affected by classifications and forecasts. If we want to know which ethical questions are relevant, it is necessary to determine in what way humans form socio-technical structures during development and use.

We understand these two essential phases of socio-technical interaction to alternate in iterations of software development. The traditional waterfall model of specifying requirements first, and implementing them afterwards has been overruled by **co-design based approaches** aiming at meaningful solutions with sustainable impact on practice (e.g. [8–10]). These approaches design software in context by iterating prototypes with relevant stakeholders (e.g. developers, designers & domain representatives): i.e. prototypes are first built and then tested in the field as part of one design iteration. *While the design team defines the research questions and re-designs the prototype in the development phase, stakeholders are confronted with the prototype during usage phase and observations are analyzed by the design team.* Here, we conceptualize that sociotechnical interactions must be reflected on alongside codesign iterations.

Reflection is the conscious *self-regulated re-evaluation of an experience* to guide future behavior in private and professionally, in formal and informal contexts [10, 11]. Reflection practice usually includes the *interruption of an ongoing activity by reflection triggers that can be guided by prompts such as reflection questions* and that result in insights applicable in one's operative activity [12]. Consequently, focusing on the

intersection of ethics and socio-technical systems in the design process allows software developers to reflect multitude of productive ethical questions.

3 SREP: Socio-technical Reflection of Ethical Principles

We have synthesized the above bodies of knowledge - understanding the object of ethical exploration to be socio-technical systems and reflection as means of learning while facing complex issues in the co-design process - into a first version of the framework for socio-technical reflection on ethical principles (SREP). The SREP framework suggests exploring ethical issues alongside the co-design process of socio-technical TEL innovations for the identification of reflection triggers. Prompting these triggers with reflection questions is expected to empower the design team to solve ethical issues based on their shared and collaboratively reflected experiences.

The main contribution of SREP is the **socio-technical design cycle for the identification of ethical reflection triggers** (see Fig. 1), which is inherently about *"building the socio-technical system" by software development and "testing the socio-technical system" in the context of usage*. In these two-alternating socio-technical design phases, **humans interact with the prototype(s) in so-called sociotechnical interactions (STIs)**. To consider ethical principles in software development, first, relevant STIs need to be identified as they help to *pinpoint reflection about ethical principles in the interactions of humans with prototypes in the development or usage phase*.

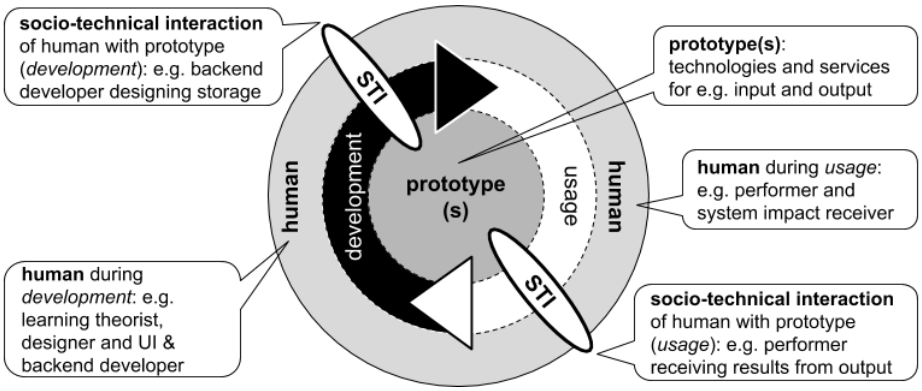

Fig. 1. Socio-technical Reflection Cycle - Identification of Ethical Issues in STIs

While the *prototype remains the same in the development and usage, it exposes not all technological elements or services for interaction with the human*. Elements and services for **algorithmic function** and **database** are exposed in the development phase, but not visible in the usage phase and only take effect there; elements and services for **input** and **output** are part of both phases. Human actors, however, reflect different persons and roles in both phases. In the development phase, humans fill roles such as **designer, learning theorist, or developer**, whereas in the usage phase they fill roles such as **performer** (e.g. lecturer) **or system impact receiver**, (e.g. student).

Due to this interdependence of humans and prototypes, a *discussion of ethical principles is only meaningful in the context of STIs*. They represent the anchor points, at which ethical issues arise and at which competing solution approaches can be ethically reflected. **To identify STIs for ethical reflection, we map social and technical components of the socio-technical system to each other** in the context of one complete design iteration*:* i.e., relate the human roles to the technological elements/ services in the development as well as usage phase (**human role x technological element**). Whereas STIs in development form the TEL prototype by designing the input mechanism (role: designer x element: input) and training the algorithm (role: data scientist x element: function), for example, STIs in usage reflect the appropriation of the TEL input and output by enhancing teaching (role: performer x element: output) and provide learning opportunities (role: system impact receiver x element: output).

Analyzing STIs in the design process from an ethical perspective represents a reflection trigger. For example, let us consider a TEL system for anticipating the student performance to prevent dropouts in a course with several exams. Unbeknownst to them, a machine learning model is trained based on test results and personal data of students who have already completed the course. With this model, the performance of new students can be predicted based on their first exam. If the system anticipates that a student will fail, they will automatically be enrolled in additional tutoring. Design teams should appropriately select and clean training data for machine learning models to avoid unfair results based on biases resulting from biased historic data (reflection trigger: data scientist x function → fairness). Furthermore, any attempt to hide machine predictions and their consequences must be avoided to ensure transparency for and agency of students (reflection trigger: impact receiver x output → transparency).

SREP proposes to **insert shared ethical reflection sessions between each socio-technical design cycle**, or even phase. This allows collaborative reflection on reflection triggers in the design team and beyond. Anticipating potential ethical issues with prompts allows to act proactively. These prompts can guide reflection with productive questions that have been inferred from past ethical issues and validated for targeted ethical reflection of certain STIs and principles. Following up the preceding example, reflection questions could include: e.g. “How to identify a bias-free data set for training the core functionality of the system?” & “How to clearly present the impact of modelling on dropout prevention to allow self-regulation and to avoid additional tutoring?”

4 Conclusion and Discussion

We propose the SREP framework for Socio-technical Reflection of Ethical Principles, which includes experiential self-regulated learning about ethical issues into the TEL co-design process. The goal is to facilitate identifying, understanding and exploring ethical issues in the co-design based software development projects in TEL. The framework interprets innovations as socio-technical systems, explicates the human involvement during development and usage, and elicits targeted discussions of ethical principles. The framework empowers practitioners and researchers to systematically integrate ethical reflection into TEL software design via a continuous reflection on

STIs. In this sense, SREP contributes to the UN sustainable development goals of equitable quality education. The framework can guide developers in designing a more inclusive, fair and higher quality learning experience by reflection on fairness and transparency in sociotechnical learning and teaching systems, for example.

In the future, we will develop SREP by applying it to an innovation initiative in higher education to deepen our understanding and design a concrete reflection practice. In ongoing software development activities, we see potential to identify and directly address ethical issues when they are discovered. The exploration of these spontaneous reflections is subject to future work, like the advantages or disadvantages of directly reflecting on ethical issues by the individual as compared to the team that allows discussing competing solution approaches. Finally, the applicability of the SREP framework to other domains should be explored. We assume that it is also suitable to inform software development in other domains such as medicine (e.g. imaging techniques).

5 References

1. Cadwalladr, C., Graham-Harrison, E.: Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach, (2018).
2. Jobin, A., Ienca, M., Vayena, E.: The global landscape of AI ethics guidelines. Nat. Mach. Intell. 1, 389–399 (2019). https://doi.org/10.1038/s42256-019-0088-2.
3. Saslow, K.: Why EU will find it difficult to legislate on AI, https://euobserver.com/opinion/147346, last accessed 2020/03/13.
4. Mulvenna, M., Boger, J., Bond, R.: Ethical by Design. In: Proceedings of the European Conference on Cognitive Ergonomics 2017 - ECCE 2017. pp. 51–54. ACM Press, New York, New York, USA (2017). https://doi.org/10.1145/3121283.3121300.
5. Eun Ah Lee, Nicholas R. Gans, Magdalena G. Grohman, Marco Tacca, M.J.B.: Guiding engineering student teams' ethics discussions with peer advising. Sci. Eng. Ethic. (2020). https://doi.org/10.1007/s11948-020-00212-6.
6. Imenda, S.: Is there a conceptual difference between theoretical and conceptual frameworks? J. Soc. Sci. 38, 185–189 (2014).
7. Ropohl, G.: Philosophy of Socio-Technical Systems. Techné Res. Philos. Technol. 4, 186–194 (1999).
8. Kvan, T.: Collaborative design: What is it? Autom. Constr. 9, 409–415 (2000). https://doi.org/10.1016/s0926-5805(99)00025-4.
9. Wang, F., Hannafin, M.J.: Design-based research and technology-enhanced learning environments. Educ. Technol. Res. Dev. 53, 5–23 (2005). https://doi.org/10.1007/bf02504682.
10. Dennerlein, S., Tomberg, V., Treasure-Jones, Tamsin Theiler, D., Lindstaedt, S., Ley, T.: Co-designing tools for workplace learning: A method for analysing and tracing the appropriation of affordances in design-based research. Inf. Learn. Sci. (2020).
11. Boud, D., Garrick, J.: Understanding Learning at Work. (1999).
12. Krogstie, B.R., Prilla, M., Pammer, V.: Understanding and supporting reflective learning processes in the workplace: The CSRL model. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8095 LNCS, 151–164 (2013). https://doi.org/10.1007/978-3-642-40814-4_13.