Poster: Ethics-Aware Software Engineering

Fatma Başak Aydemir
Utrecht University
The Netherlands
f.b.aydemir@uu.nl

Fabiano Dalpiaz
Utrecht University
The Netherlands
f.dalpiaz@uu.nl

ABSTRACT

Human interaction and behavior are at the core of most software engineering (SE) activities. Furthermore, software is created to fulfill human stakeholders' needs and wishes, and the resulting software is ultimately utilized (directly or indirectly) by human users. Today's software is highly intertwined with our lives, and it possesses an increasing ability to act and influence us. Besides the obvious example of self-driving cars and their potential harmfulness, more mundane software such as social networks may introduce bias, break privacy preferences, lead to digital addiction, etc. Additionally, the SE process itself is highly affected by ethical issues, such as diversity and business ethics. This paper introduces ethics-aware SE, a version of SE in which the ethical values of the stakeholders (including developers and users) are captured, analyzed, and reflected in software specifications and in the SE processes.

KEYWORDS

ethics, ethical software engineering, ethical design, ethics compliance, requirements engineering

ACM Reference format:
Fatma Başak Aydemir and Fabiano Dalpiaz. 2018. Poster: Ethics-Aware Software Engineering. In *Proceedings of 40th International Conference on Software Engineering Companion, Gothenburg, Sweden, May 27-June 3, 2018 (ICSE '18 Companion),* 2 pages.
https://doi.org/10.1145/3183440.3195075

1 INTRODUCTION

For the selection and purchase of non-software goods and services, people are increasingly taking into serious account *ethical aspects* [1, 2] besides the more traditional criterion of cost-effectiveness. People are rightfully concerned about how goods are produced (e.g., do subcontractors use child labor? Is fair payment guaranteed?), what the cost for the environment is (e.g., what is the carbon footprint? Are the raw materials obtained without poisoning or destroying natural ecosystems?), and what substances goods contain that may have negative side-effects on people's health (e.g., are there any pesticides in the food? Do any of the ingredients lead to some kind of addiction?).

We advocate that similar questions should be asked for the software products we use, and ethics should play a role in the decisions of end-users, customers, industry professionals, and companies regarding the production and use of software. There are numerous ethical concerns such as diversity, inclusiveness, sustainability, that can be applied to software development processes and products.

ICSE '18 Companion, May 27-June 3, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5663-3/18/05…$15.00
https://doi.org/10.1145/3183440.3195075

2 BACKGROUND

Cohen and Cornwell [3] discuss teaching information systems ethics. Pierce and Henry [4] argue that proper ethical behavior of software professionals is influenced by the individual's own code, the code that exists in the work place, and the formal code of conduct and discusses the degree of influence of these three codes. Hall [5] discusses good professional conducts for software engineers. Gotterbarn [6] provides a list of ethical issues concerning software engineering. Allhoff [7] approaches to applied ethics from a science and engineering perspective. Software Engineering Code of Ethics and Professional Practice is the output of the joint task force by ACM and IEEE-CS [8].

3 OUR VISION OF ETHICS-AWARE SE

3.1 The Principle of Harmony

Our goal is to lay the foundations for ethics-aware software engineering to *establish harmony* with respect to ethical values and behavior in SE. This includes (i) *creating awareness* of ethical issues regarding software artifacts and software development processes, (ii) providing stakeholders with tools to let them *articulate* their ethical values, and (iii) building methods to *monitor, verify, and validate* software engineering processes and software artifacts to ensure compliance with those values.

3.2 The Ethics-Aware SE Method

Figure 1 is a high-level illustration of our method for ethics-aware SE. It involves all stakeholders regarding software development, from customer and software users to developers, organizations, and others. The method is agnostic of specific development paradigms; its activities can be embedded in any method including agile, waterfall, iterative, and V-model. Five distinct phases of the method are shown in a cycle which indicates that being ethics-aware is not a one-shot activity, but rather requires continuous effort throughout software design and development, maintenance, evolution, and use. Figure 1 also lists four distinct enablers of being ethics-aware in software engineering practices (E0–E3).

Articulation involves eliciting, modeling, and analyzing ethics values, what we call *ethics requirements* for software artifacts and their development processes. One source of such requirements is, naturally, the traditional stakeholders of software artifacts, such as customers, users, legislative organizations. However, the voices of software engineering professionals and companies also need to be heard to ensure the harmony among all stakeholders, as well as the products they build.

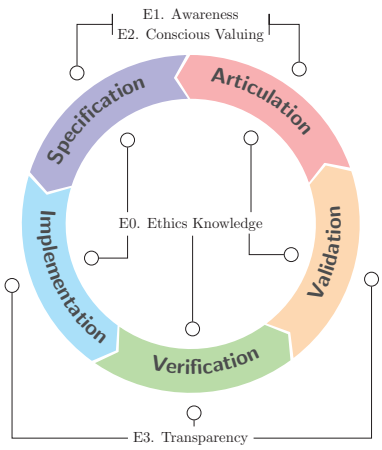

Figure 1: Method for Ethics-Aware Software Engineering

Specification fills the gap between the ethics requirements and the corresponding functional and quality requirements on software artifacts and processes. This step requires transferable knowledge from the discipline of ethics to software engineering. For example, if the customer states that "the system shall be gender neutral", the specialist must be able to identify the related use cases, and the compliant system behavior.

Implementation not only refers to writing code and documents for software products, but also building software development processes based on the ethics specification. A key challenge is to prevent the way-too-common situation in which quality requirements are disregarded because of time-to-market constraints.

Verification of ethics requirements involves both the software artifact and the software development processes, and require organizations to put in place appropriate means to continuously check that the software is being built according to the ethics specification.

Validation checks whether the software artifact and the development processes satisfy the ethics requirements imposed by the stakeholders at the beginning of the process. The outcomes of validation may result in a certificate that testifies how well a software system aligns with its intended ethics requirements, and explains deviations, if any.

We have identified four enablers for the adoption of ethics-aware SE by a range of stakeholders including users with no technical background to software development organizations:

E0: Ethics knowledge is required for each step involved in the method. Users, professionals, and organizations should know about ethical issues related to software engineering to decide whether they care about these issues, and what is their desired "right" behavior for each issue. Especially for the articulation and specification steps, professional knowledge on both ethics and software engineering is required to identify use and misuse cases and translate ethical concerns into concrete system behavior.

E1: Awareness. All stakeholders must be aware of the ethical issues and their consequences within the realm of SE. Issues are discussed and treated only when there is a wide awareness of them. For example, there has been a great awareness on privacy issues in software, as a result companies have provided several tools and customization options for regulating user privacy, and legislative institutions have regulated the specific issues.

E2: Conscious valuing. The stakeholders must develop a conscious value for the issue being discussed. If a certain issue, such as accountability of the software engineers for algorithmic bias does not have a value for the public, they will not articulate their requirements for the issue for they do not care about it. However, this situation might change depending on the course of events, for example allegations on social media for biased timeline presentations might create first awareness (E1) and then conscious value (E2) for an ethical issue in software engineering.

E3: Transparency. In order to ensure that the behavior of the artifact as well as software development processes follow the specification, transparency is key. Software development processes and the behavior of the artifact must be transparent so that they can be validated against the ethics requirements.

4 CONCLUSIONS

Our work is inspired by other phenomena that are occurring in our society, such as responsible movements in food and clothing industries due to public aware ness, conscious valuing, and demand of ethical behavior. In this light, our aim is to provide an umbrella ethics framework that educates end users as well professionals on ethics, and that provides alternative solutions based on their individual and organizational values as well as their requirements.

Our future includes expanding the list of ethics issues and creating a classification to use as the basic input for further elicitation and prioritization of ethics requirements. Another input we are working on is the creation of short scenarios to provide as examples to the software engineering crowd to collect their preferred behavior. Also, we are working on refining the modeling primitives for capturing ethics statements and requirements. Our aim is to reason on ethics requirements to detect and resolve conflicts, analyse satisfaction and later on check compliance.

REFERENCES

[1] K. Dowd and K. J. Burke, "The influence of ethical values and food choice motivations on intentions to purchase sustainably sourced foods," *Appetite*, vol. 69, pp. 137–144, 2013.
[2] C. Jayawardhena, K. Morrell, and C. Stride, "Ethical consumption behaviours in supermarket shoppers: determinants and marketing implications," *Journal of Marketing Management*, vol. 32, no. 7-8, pp. 777–805, 2016.
[3] E. Cohen and L. Cornwell, "A question of ethics: Developing information system ethics," *Journal of Business Ethics*, vol. 8, no. 6, pp. 431–437, 1989.
[4] M. A. Pierce and J. W. Henry, "Computer ethics: The role of personal, informal, and formal codes," *Journal of Business Ethics*, vol. 15, no. 4, pp. 425–437, Apr 1996.
[5] D. Hall, "The ethical software engineer," *IEEE Software*, pp. 9–10, 2009.
[6] D. Gotterbarn, "Ethical considerations in software engineering," in *Proceedings of the 13th International Conference on Software Engineering*, 1991.
[7] F. Allhoff, "What are applied ethics?" *Science and Engineering Ethics*, vol. 17, no. 1, pp. 1–19, 2010.
[8] A.-C. J. T. Force", "Software engineering code of ethics and professional practice (version 5.2)," Online. [Online]. Available: https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct