

A Roadmap for Ethics–Aware Software Engineering

Fatma Başak Aydemir and Fabiano Dalpiaz

Presented by Elliott Skomski

A Question of Ethics

Society increasingly concerns itself with ethical issues existing outside of the tech industry

Given the pervasiveness of technology, shouldn't we be asking those questions as well?

If we are asking questions, are we asking the right ones?

Decisions regarding production and use of software should be based on ethical concerns

Argument: these concerns give rise to familiar suitability criteria, like functional requirements

These decisions should involve *everyone*

What About Standards?

Standards like ACM SE Code of Ethics provide adequate guidance, right?

High-level guidance for professional ethics

Initiatives exist to address specific concerns in tech as well, notably AI

Okay cool, but what about the rest of software?

We lack a reliable, general framework for communicating and dealing with ethical concerns

Current solutions don't address concerns of other stakeholders

What Questions Should We Ask?

What are the working conditions of software developers?

e.g. outsourcing, game industry

What are the environmental costs of developing and maintaining specific software?

Does the system encourage unhealthy behavior that may lead to addiction?

e.g. dark patterns

Does the software violate individual preferences for user privacy?

e.g. literally any social network

What Questions Should We Ask?

Are gender and diversity issues considered?

e.g. Harassment at Uber, ex-Google anti-diversity manifesto

Do systems avoid discrimination based on gender, race, age, income, etc.?

Systems inherit human biases

Are a software firm's business practices unethical or exploitative?

To what extent is software evolution driven by a democratic need analysis of users?

Vision of Ethics–Aware Software Engineering

Ethics concerns gave rise to codes of conduct, but these lack the breadth necessary to include anyone other than software professionals

Customers, organizations, and users play a crucial role in the software development process

To this end, how do we develop methods to understand and address the ethical concerns of people who aren't in software?

The Principle of Harmony

To the authors, developing such a framework starts with establishing *harmony*:

- Creating awareness of ethical issues

- Giving stakeholders ways to communicate their values

- Developing ways to verify compliance with values

Notion of harmony varies among stakeholders

An Ethics-Aware SE Method

A continuous, cyclical method of ethics awareness with four prerequisites for ethics-aware software engineering

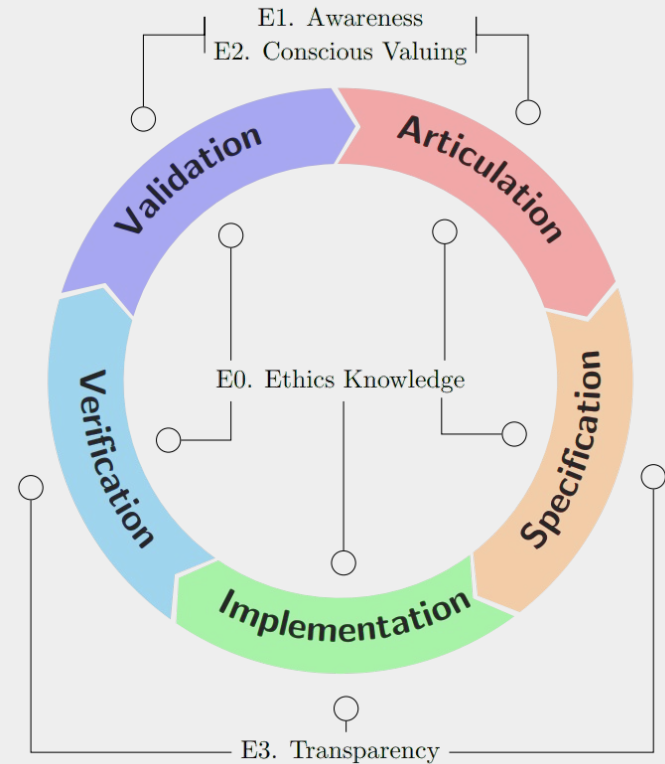
Articulation – elicit, model, and analyze ethics reqs

Specification – map ethics reqs to functional, quality reqs

Implementation – write software conforming to ethics specs, disregarding time-to-market constraints

Verification – continuously check software for ethics conformance

Validation – check if software aligns with ethics reqs



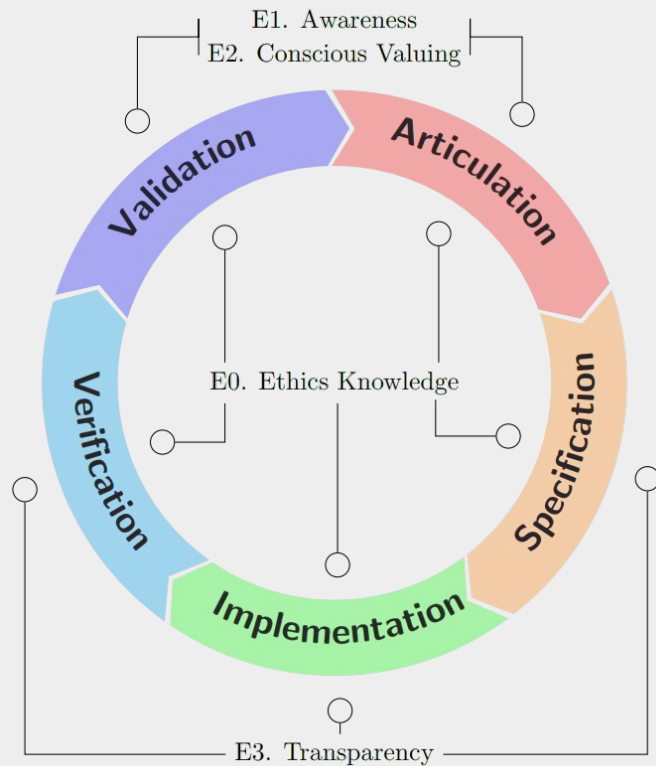
What Are the Prerequisites?

Ethics Knowledge – knowing of ethics issues, deciding their importance, determining what is “right”

Awareness – knowing the issues as well as their consequences deeply

Conscious Valuing – develop values for ethical issues

Transparency – ensure development processes and software can be validated against ethics reqs



Research Questions

Given this proposed framework, how do we move toward developing and validating it?

RQ 1: *What are the relevant ethics issues for software engineering?*

- Build a catalog of ethics issues in software engineering

- Develop crowd-driven ethics standards and codes of conduct

- Support requirements elicitation for both analysts and end users

RQ 2: *What are adequate modeling primitives to capture ethics requirements?*

- Develop a language for expressing ethics requirements

Research Questions

RQ3: *Which visual notations can help capture ethics requirements?*

Allow for communication of different perspectives via a common visual notation

RQ4: *How to analyze the interplay of ethics requirements and other requirements?*

Identify conflicts among ethics requirements and between functional/quality requirement

RQ5: *Which techniques can help trace ethics requirements?*

Discover ways to map abstract ethics requirements to concrete functional/quality requirements

Research Questions

RQ6: *How can we verify software artifacts and processes against ethics specifications?*

Devise acceptance tests and unit tests, and find ways to generalize tests across products

RQ7: *How can we validate software against ethics requirements?*

Ensure strong comprehension of requirements, develop ethics acceptance tests for customers

Conclusion

An ethics framework for software engineering is proposed

Meant to address the lack of a general framework for ethics-aware software engineering

Emphasizes ethical harmony in software and development through deliberate ethics considerations

Framework will be incrementally built and validated via exploration of research questions

Future work will begin with addressing RQ1 and RQ2:

- Cataloging ethical issues and conduct

- Developing a language for communicating ethics reqs

Questions?