CS 4530
Fundamentals of Software Engineering

Module 17A: Engineering Ethical Software

Adeel Bhutta, Mitch Wand

Khoury College of Computer Sciences

Learning Goals

- By the end of this lesson, you should be able to…
  - Illustrate how software can cause inadvertent harm or amplify inequities
  - Explain the role of human values in designing software systems
  - Explain some techniques that software engineers can use in producing software systems that are more congruent with human values.

Ethically and morally implicated technology is **everywhere**!

- Algorithms that gate access to loans, insurance, employment, government services…
- Algorithms that perpetuate or exacerbate existing discrimination
- Bad medical software can kill people (Therac-25)
- UIs that discriminate against differently-abled people
- Third-party data collection for hyper-targeted advertising
- LLM's that harvest copyright or personal data
- And on… and on… and on…

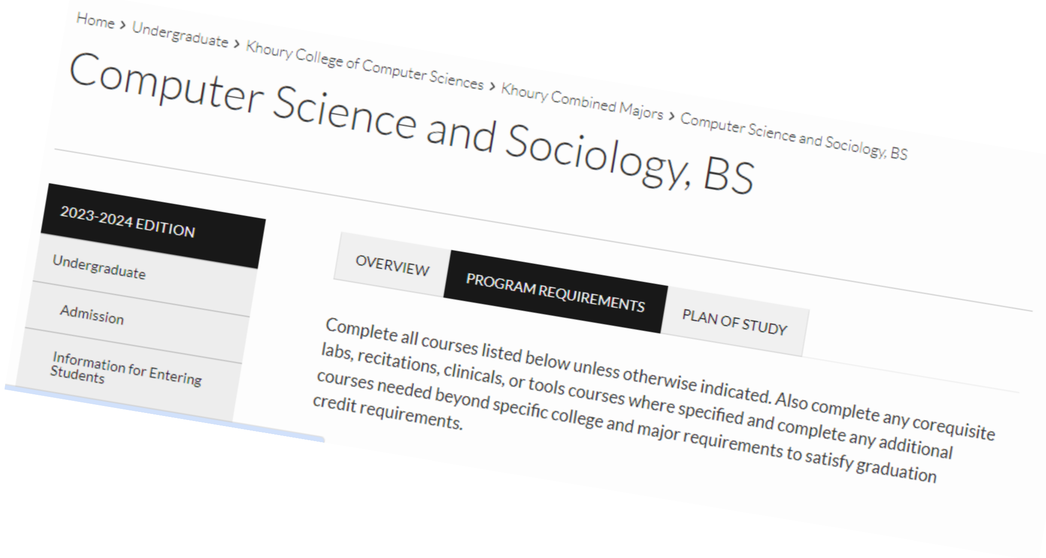
And this is only the tip of the iceberg

- Other Challenges:
  - interfaces and systems designed to be addictive;
  - corporate ownership of personal data;
  - weak cyber security and personally identifiable information (PII) protection;
  - and many more …

SOCL 4528. Computers and Society. (4 Hours)

Focuses on the social and political context of technological change and development. Through readings, course assignments, and class discussions, offers students an opportunity to learn to analyze the ways that the internet, artificial intelligence, and other technological advances have required a reworking of every human institution—both to facilitate the development of these technologies and in response to their adoption.

Attribute(s): NUpath Difference/Diversity, NUpath Societies/Institutions



Home › Undergraduate › Khoury College of Computer Sciences › Khoury Combined Majors › Computer Science and Sociology, BS

Computer Science and Sociology, BS

2023-2024 EDITION

Undergraduate

Admission

Information for Entering Students

OVERVIEW | PROGRAM REQUIREMENTS | PLAN OF STUDY

Complete all courses listed below unless otherwise indicated. Also complete any corequisite labs, recitations, clinicals, or tools courses where specified and complete any additional courses needed beyond specific college and major requirements to satisfy graduation credit requirements.

Equity and Software

As new as software engineering is, we're newer still at understanding its impact on underrepresented people and diverse societies.

We must recognize imbalance of power between those who make development decisions that impact the world.

and those who simply must accept and live with those decisions that sometimes disadvantage already marginalized communities globally.

Quote: "Software Engineering at Google: Lessons Learned from Programming Over Time," Wright et al, 2020 (O'Reilly)

Recognize inequities in your software

One mark of an exceptional engineer is the ability to understand how products can advantage and disadvantage different groups of human beings

Engineers are expected to have technical aptitude, but they should also have the discernment to know when to build something and when not to

Demma Rodriguez
Head of Equity Engineering, Google 2018-2020
Meta 2020-2022
AirBnB 2022-present

Quote: “Software Engineering at Google: Lessons Learned from Programming Over Time,” Wright et al, 2020 (O’Reilly)

What values might our software promote or diminish?

- Human rights - Inalienable, fundamental rights to which all people are entitled
- Accessibility - Making all people successful users of the technology
- Justice - Procedural justice (process is fair) + distributive justice (outcomes are fair)
- Privacy - An individual's agency in determining what information about them is shared
- Human welfare - Physical, material and psychological well-being

Value Sensitive Design @ Khoury

To analyze this question, we have to understand the human and social contexts in which our software will run

Some categories of contexts:

- Social Context
- Business Context
- Legal and Regulatory Context

What is the social context?

- what categories of people will benefit from our software?
- what categories of people will be harmed by the use of our software?

What is the business context?

- Who is going to pay for this software?
  - users?
  - advertisers?
  - sponsors or sponsoring agencies?
- What are their incentives?
  - how do their incentives affect (or distort) our priorities in designing or developing this software?
    - if our sponsors are selling advertising, then we may be pressed to prioritize "engagement"
    - if our sponsors want to help disadvantaged people to connect with services, we may be pressed to prioritize accessibility.

Who is selling what to who?

- "If you're not paying for the product, then you are the product"

--

generally credited to Richard Serra and Carlota Fay Schoolman (1974, about TV advertising)

What is the legal and regulatory context?

- Americans with Disabilities Act (ADA)
- Litigation-averse sponsors may insist on elaborate Terms & Conditions
- Software for use in the EU may need to comply with the GPDR.
- What about financial software?
- What about personal data?
- What about leaving cookies, etc., on our machines?

Is the activity of the software transparent?

- What data does it collect about the individual user?
- Does it store things on our computers?
- Does it touch our files?
  - Does it violate the "CIA" of computer security?

Special considerations for AI

- Incentives for AI system may be particularly mysterious
- What objective function is your LLM optimizing?
- AI "agents" that operate in the real world present particularly complicated risks

A short course in AI safety

aisafety.dance

A short plug for Nicky Case

ncase.me

Consider human values throughout the project

- Projects evolve, often in unpredictable ways
- New issues may arise as the project is elaborated
- Users will ALWAYS use the software in unexpected ways
- Users will ALWAYS find ways to misuse the software

Identifying Unintended Consequences

- Technology *will* be adopted in unanticipated ways. Being intellectually rigorous means considering and mitigating risks in designs ahead of time.
- What if:
  - Our recommendation system promotes misinformation or hate speech?
  - Our database is breached and publicly released?
  - Our facial recognition AI is used to identify and harass peaceful protestors?
  - Our child safety app is used to stalk women?
  - Our chatbot is sexist or racist?

Example 1: Content Moderation

The issue: *free expression* in tension with *welfare* and *respect*

- Some speech may be hurtful and/or violent
- Removing this speech may be characterized as censorship

Bad take: unyielding commitment to free speech, no moderation

- Trolls and extremists overrun the service, it becomes toxic, all other users leave
- Violent speech actually impedes free speech in general

Bad take: strict whitelists of acceptable speech

- Precludes heated debate, discussion of “sensitive topics”
- Disproportionately impacts already marginalized groups

Good take: recognizing that moderation will never be perfect, there will be mistakes and grey areas

- Doing nothing is not a viable option
- Clear guidelines that are earnestly enforced create a culture of accountability

Update (January 2025): Or maybe the owner's politics will dictate your site's moderation policies (or lack thereof)

Example 2: Image Generation

AI text-to-image generators have a well-documented bias problem. AI models are trained on images from the internet, so bias in, bias out. A recent experiment from Bloomberg on the image generator Stable Diffusion found that AI portraits of architects, doctors and CEOs skewed white and male, while images of cashiers and housekeepers skewed towards women of color.

Adobe's solution to the bias issue was to use data that estimates the skin tone distribution of a Firefly user's country, and apply it randomly to any human Firefly creates. In other words, if someone in the U.S. used Firefly to make an image of a doctor or a gardener, the chances that person would be a woman or have non-white skin would be roughly proportional to the percentage of women and people of color in the U.S.

In Firefly world, about 14% of doctors should be Black — the same percentage as the Black population in the U.S. But in the messy, unequal real world, only 6% of doctors are Black.
So, should AI images depict the world as it is? Or as it should be?
"That becomes almost like a philosophical question," said Rumman Chowdhury, a Responsible AI Fellow at Harvard's Berkman Klein Center for Internet and Society.

https://www.marketplace.org/2023/10/10/solutions-to-ai-image-bias-raise-their-own-ethical-questions/

There are SE-level mitigations for some of these risks

- Form a diverse team
  - People from diverse backgrounds bring different experiences and different perspectives
- Consider human values throughout the project
- Rely on standards when possible
  - ADA
  - ARIA
  - etc.
- Monitor actual usage & misusage, user feedback
  - who? what? when? how?

Systemic mitigations

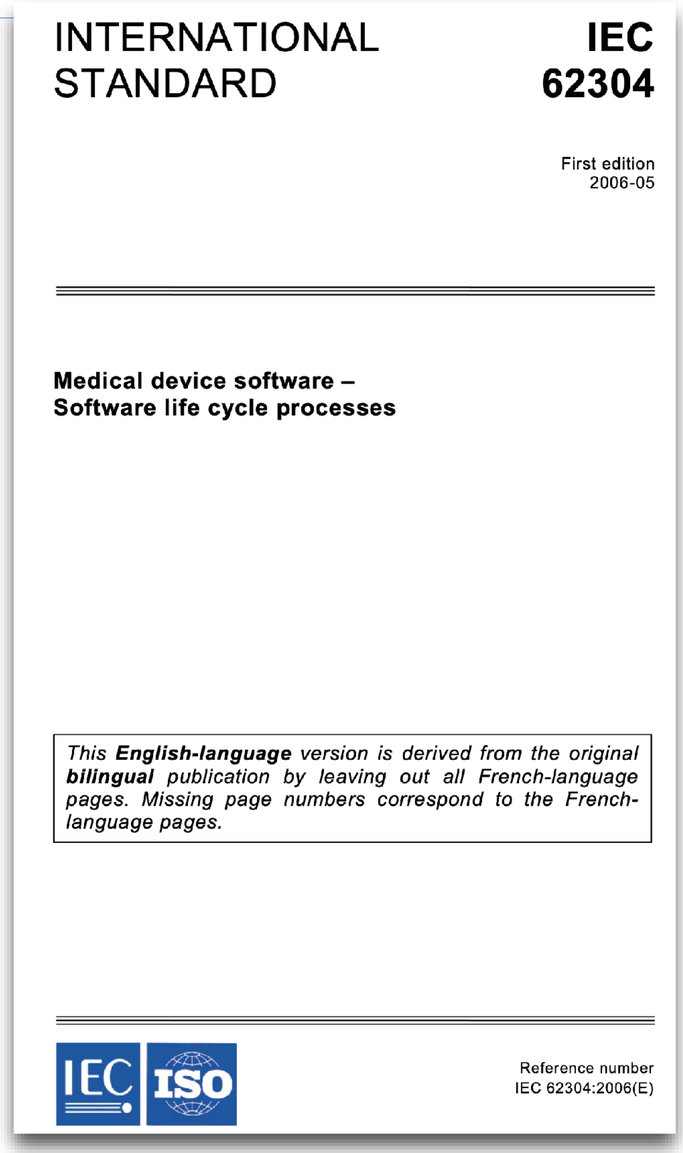
- Work for systemic change?
  - political, social, etc.
- Work to convince developers to consider human values?

What affects developer behavior?

- Standards?
- Codes of ethics?
- Hippocratic oath?

Standards can give guidance.

- International bodies define standard processes that are designed to protect the public
- By (correctly) following such a standard, you can reduce the chance of harm to users, as well as your ethical (and legal) liability
- You can work to update/expand existing and new standards



INTERNATIONAL STANDARD

IEC 62304

First edition
2006-05

**Medical device software –
Software life cycle processes**

*This **English-language** version is derived from the original **bilingual** publication by leaving out all French-language pages. Missing page numbers correspond to the French-language pages.*

IEC ISO

Reference number
IEC 62304:2006(E)

https://ethics.acm.org/code-of-ethics/software-engineering-code/

ACM Software Engineering Code of Ethics

. PUBLIC – Software engineers shall act consistently with the public interest.

2. CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.

3. PRODUCT – Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

4. JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.

5. MANAGEMENT – Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

6. PROFESSION – Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. COLLEAGUES – Software engineers shall be fair to and supportive of their colleagues.

8. SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

Does this code change developer behavior?

Does ACM's Code of Ethics Change Ethical Decision Making in Software Development?

Andrew McNamara
North Carolina State University
Raleigh, North Carolina, USA
ajmcnama@ncsu.edu

Justin Smith
North Carolina State University
Raleigh, North Carolina, USA
jssmit11@ncsu.edu

Emerson Murphy-Hill
North Carolina State University
Raleigh, North Carolina, USA
emerson@csc.ncsu.edu

ABSTRACT

Ethical decisions in software development can substantially impact end-users, organizations, and our environment, as is evidenced by recent ethics scandals in the news. Organizations, like the ACM, publish codes of ethics to guide software-related ethical decisions. In fact, the ACM has recently demonstrated renewed interest in its code of ethics and made updates for the first time since 1992. To better understand how the ACM code of ethics changes software-

The first example is the Uber versus Waymo dispute [26], in which a software engineer at Waymo took self-driving car code to his home. Shortly thereafter, the engineer left Waymo to work for a competing company with a self-driving car business, Uber. When Waymo realized that their own code had been taken by their former employee, Waymo sued Uber. Even though the code was not apparently used for Uber's competitive advantage, the two companies settled the lawsuit for $245 million dollars.

TLDR: No

What about a Hippocratic Oath for Software Developers?

- Would be formulated collectively by practitioners, not by a professional body
- Reinforced by social incentives
- Might this work? Who knows? Software Engineering Needs A Hippocratic Oath

Where does this leave us?

- **So that we can sleep at night**
  - Consider the different ways that our software may impact others
  - Consider the ways in which our software interacts with the political, social, and economic systems in which we and our users live
  - Follow best practices, and actively push to improve them
  - Encourage diversity in our development teams
  - Engage in honest conversations with our co-workers and supervisors to explore possible ethical issues and their implications.