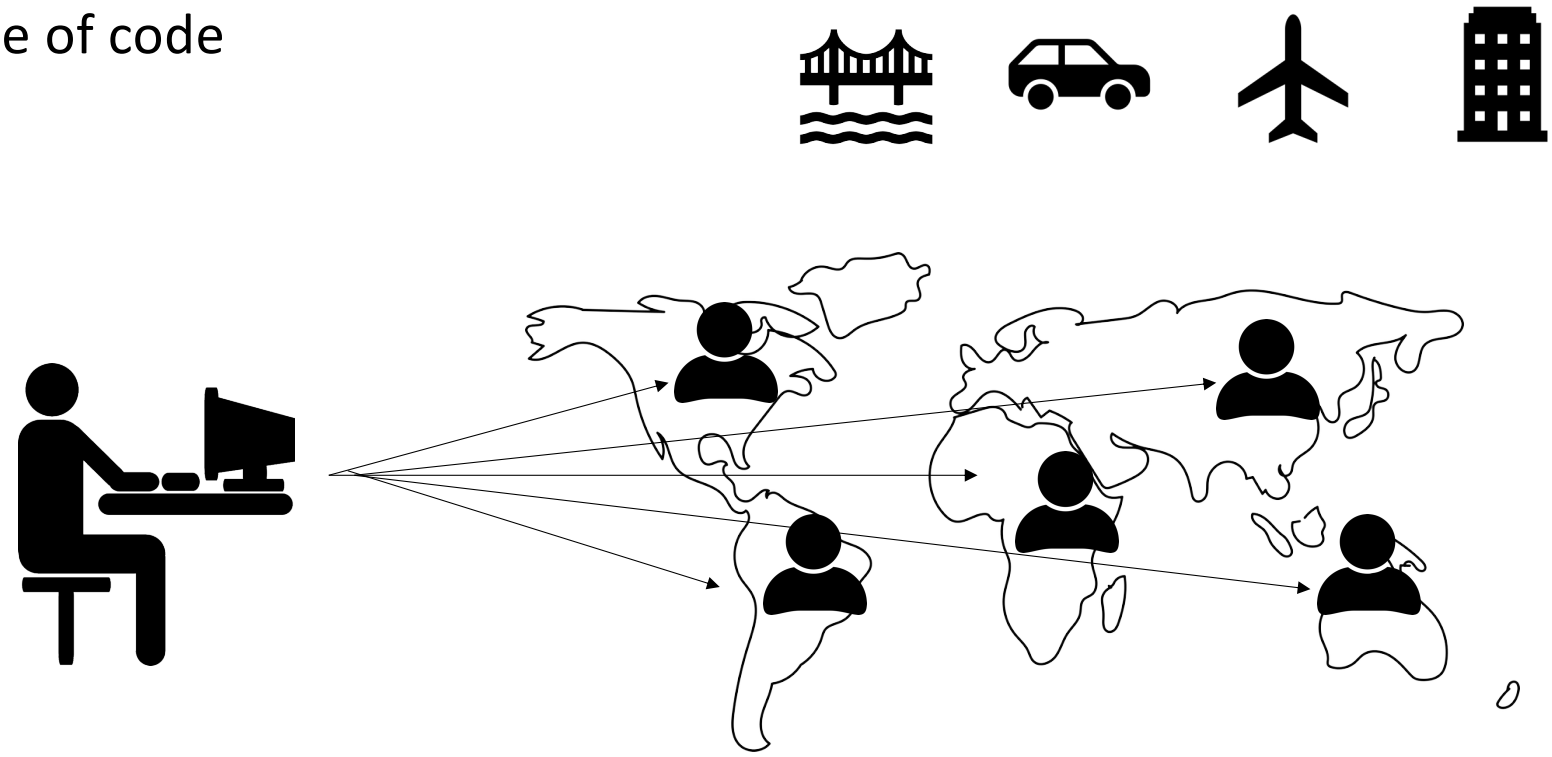
Software Engineering Ethics

Jin L.C. Guo

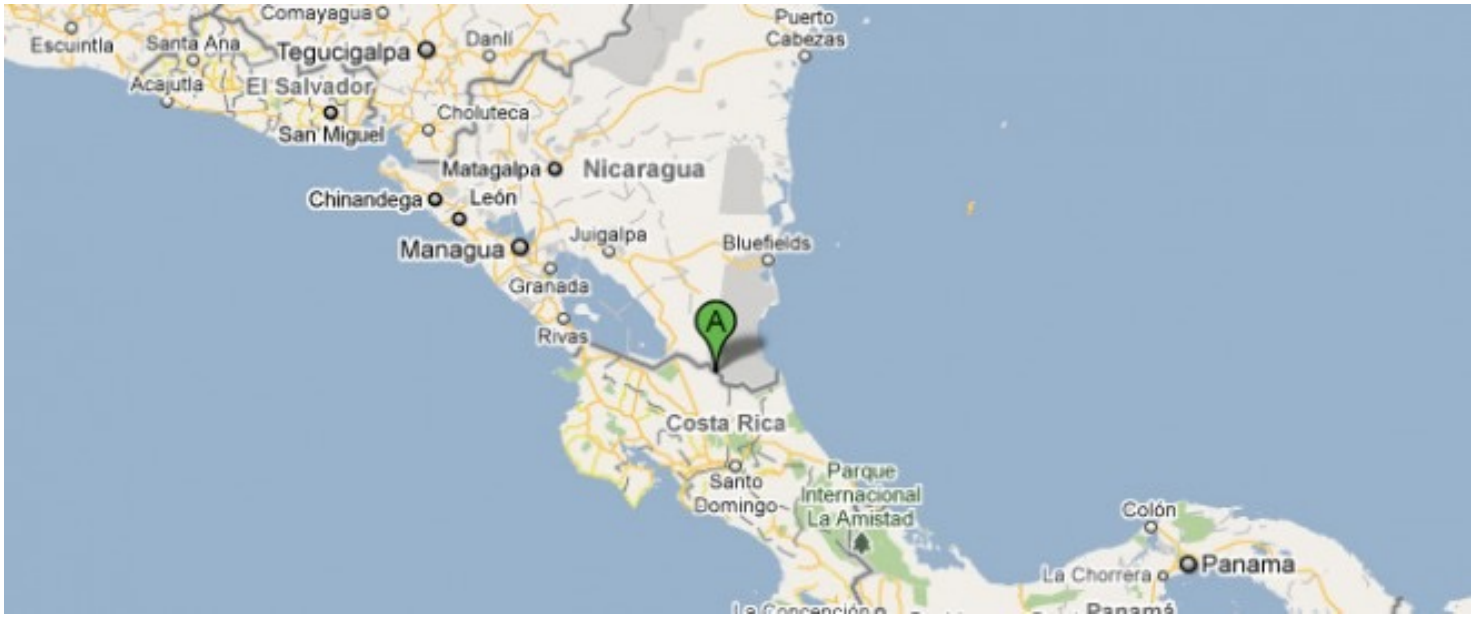

What does ethics
mean to you?

Software Engineer Ethics

- Build line of code



Nicaragua to keep troops in disputed territory after Google Maps error



More common issues

- Addictive design;
- Questionable personal data ownership;
- Algorithmic bias;
- Weak cyber security;
- Overemphasis on features;
- ….

Ethical Obligations of Software Engineers

- Personal level

  Take full responsibility for our moral choices and their consequences

- Professional level

  What does integrity look like in a software engineer?
  What sort of specific coding practices demonstrate integrity, or a lack of it?

Professional Ethics - Medical Ethics

- Respect for Autonomy
- Beneficence and non-maleficence
- Justice

Gillon, Raanan. "Medical ethics: four principles plus attention to scope." *Bmj* 309, no. 6948 (1994): 184.

Case Study 1:

- Karen
- Lives with an incredibly hectic and stressful schedule
- Using an App Errand Whiz

App Errand Whiz

- Merges information from to-do list, retail stores, and GPS software
- Produce the most efficient map and directions for running errands on a given day
- To accomplish this, the app aggregates information about
  - where Karen lives and shops,
  - what she typically buys in each store,
  - how much she buys,
  - what she typically pays for each item.
- This collected data is stored on a separate server that the app links to when it needs to create a shopping map.
- The app encourages users to log in via Facebook, as the developers have made a deal with Facebook to sell this data to third-party advertisers, for the purpose of targeting Facebook ads to Karen and her friends.

Questions

1. In what ways could Karen potentially be harmed by this app, depending on how it is designed and how her shopping data is handled and used?

2. Which if any of these harms could result from ethical failings on the part of the people who developed Errand Whiz? How, specifically?

3. What actions could the people behind Errand Whiz take to prevent these harms? Are they ethically obligated to prevent them? Why or why not?

Doing good

- What things can excellent Software Engineers contribute to the good life?

- What kinds of character traits, qualities, behaviors and/or habits do you think mark the kinds of people who tend to contribute most in these ways?

Dilemmas

"Mission impossible"
- being asked to create or accept a product schedule that is clearly impossible to meet

"Mea Culpa"
- delivering products without key functionality or with known defects

"Rush Jobs"
- delivering products of subpar quality to meet schedule pressures

"Red Lies"
- telling clients or management known falsehoods about product schedule or performance

"Fictionware"
- promising features that are infeasible

"Nondiligence"
- inadequate review of requests for proposals, contracts or specifications

Brian Berenbach and Manfred Broy (2009), "Professional and Ethical Dilemmas in Software Engineering," Computer, 42:1, 74-80.

Case Study 2

- Read the article "Whistleblowers: Software bug keeping hundreds of inmates in Arizona prisons beyond release dates"

Questions

- What kind of problems directly related to the software are mentioned in the article?

Questions

- Who were the various **stakeholders** whose interests were ethically significant here? What were the interests of each stakeholder? Whose interests should have taken precedence in the minds of the relevant actors, and why?

- Who might have the power to prevent this outcome?

Questions

- Put yourselves in the shoes of the software engineers employed on this project. How the outcome of this case would affect you, personally and professionally.
- How would you feel about your friends, family, neighbors and mentors learning that you were involved in the issue, and that your work was implicated in the suffering and potential deaths of many innocent people?
- What would you say to them to explain yourself? Would any explanation be adequate?

ACM/IEEE-CS Software Engineering Code

- For anyone that is a member of the software engineering profession

Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession. In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the Eight Principles.

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

PUBLIC

Software engineers shall act consistently with the public interest.

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.

PRODUCT

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

JUDGMENT

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

MANAGEMENT

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

PROFESSION

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

COLLEAGUES

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

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.

PRODUCT

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

3.06. Work to follow professional standards, when available, that are most appropriate for the task at hand, departing from these only when ethically or technically justified.

3.08. Ensure that specifications for software on which they work have been well documented, satisfy the users' requirements and have the appropriate approvals.

3.10. Ensure adequate testing, debugging, and review of software and related documents on which they work.

3.13. Be careful to use only accurate data derived by ethical and lawful means, and use it only in ways properly authorized.

3.15 Treat all forms of software maintenance with the same professionalism as new development.

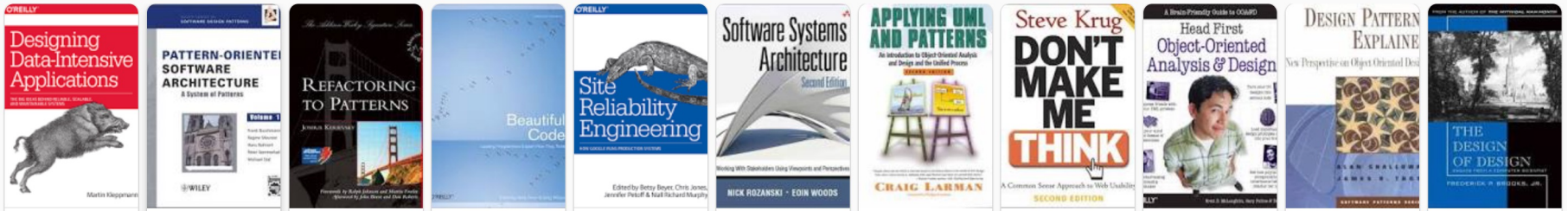
Acknowledgement

- Some content in this module is adapted from “An Introduction to Software Engineering Ethics” by the Markkula Center for Applied Ethics at Santa Clara University with permission.

COMP 303 Software Design Summary

Why are we here?

Design Patterns: Ele… 1994 | Domain-driven design Eric J. Evan… | Code Complete Steve McCo… | Head First Design Patte… 2004 | Refactoring 1999 | Clean Architecture:… Robert Cecil… | Agile Principles, P… 2002 | Code: The Hidden Lang… 1999 | A Philosophy of Software … John Ouster… | Building Microservice… Sam Newma… | The Algorithm Design Manual Steven Skie…

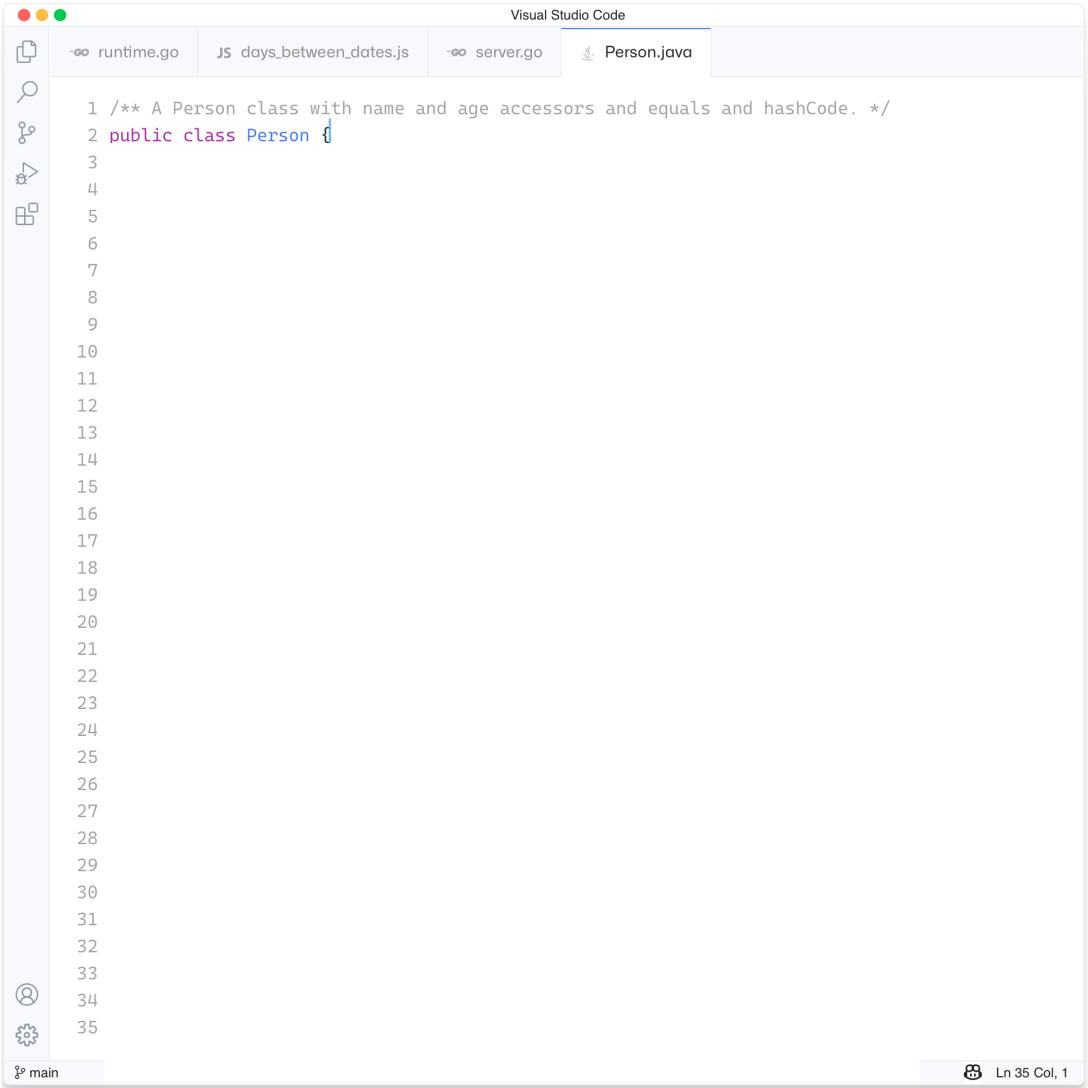


Designing Data-Intensi… Martin Klepp… | Pattern-oriented Soft… 1995 | Refactoring to Patterns Joshua Keri… | Beautiful Code 2007 | Site Reliability Engineering:… 2016 | Software Systems Arc… 2005 | Applying UML and patterns Craig Larma… | Don't Make Me Think Steve Krug, … | Head First Object-Orien… Brett McLau… | Design patterns expl… Alan Shallo… | The Design of Design Fred Brooks,…

… …

Why are we here?

"GitHub Copilot is powered by Codex, the new AI system created by OpenAI. GitHub Copilot understands significantly more context than most code assistants. So, whether it's in a docstring, comment, function name, or the code itself, GitHub Copilot uses the context you've provided and synthesizes code to match. Together with OpenAI, we're designing GitHub Copilot to get smarter at producing safe and effective code as developers use it."



https://copilot.github.com

https://youtu.be/SGUCcjHTmGY

This course is about?

- Design Principles (separation of concerns, encapsulation, substitutability, interface segregation, etc.)
- Design patterns;
- Design techniques such as UML Diagrams and Design by Contract;

Properly *Explain* and *Apply*

This course is about?

- Programming language mechanisms such as exception handling, concurrency and synchronization, reflection, concurrency;

Effectively *Use*

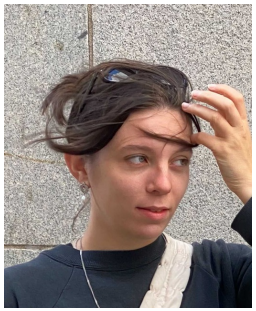
This course is about?

- The quality of design solutions *Analyze* and *Evaluate*
- Design smells *Identify*

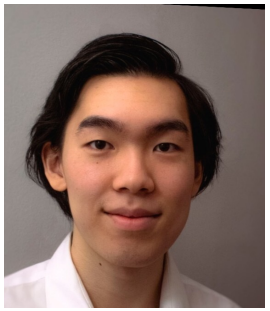
Huge Thanks to **Our TA Team**

Deeksha Arya

Avinash Bhat



Béatrice Duval



Peter Rong

Violet Shi

Jiahao Chen

Beyza Yıldırım

Anthony Ong

Linhui (Malinda) Huang

Shawn Hu

Xinran Xiong

Huge Thanks to **YOU**

During Class

Individual Interaction

Ed discussion forum

What’s missing

- Refactoring
- Usability

What's missing

- Techniques for larger systems and development team
  - Development and Deployment Tools
  - Scheduling
  - Team

What's missing

COMP 529 Software Architecture
COMP 533 Model-Driven Software Development
COMP 585 (599) Intelligent Software Systems
ECSE 428 Software Engineering Practice
ECSE 429 Software Validation
ECSE 557: Introduction to Ethics of Autonomous Intelligent Systems
… …

Question For you

- Five years from now on, what will you remember from this class?

Last Requests

- Ask ethical questions throughout your professional career;
- Use the techniques and tools of COMP 303 effectively;
- Course Evaluation