CSE 403

Software Engineering

Course introduction

Today

- The CSE 403 team
- Logistics and resources
- What is Software Engineering
- Course overview and expectations

The CSE 403 team

Instructor

- Michael Ernst; office hours: after class and by appointment
- Best email: cse403-staff@cs; you may also use mernst@cs

TAs

- Thomas Chen
- Hady Fawal
- Saket Gollapudi
- Hannah Potter
- Yixuan Wang

Your TA has multiple roles:

- Venture capitalist who expects results
- Manager who helps when you have difficulty
- Grader

Today

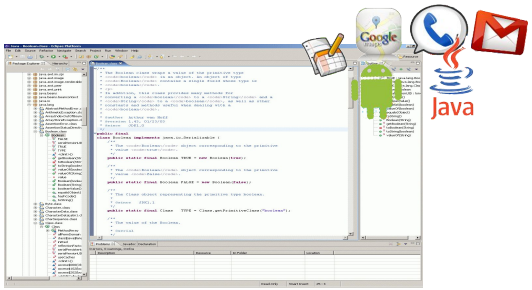
- The CSE 403 team
- Logistics and Background
- What is Software Engineering
- Course overview and expectations

What is Software Engineering?

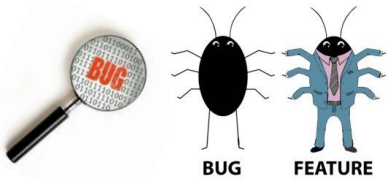
How can you, a software engineer,
deliver a product/service
that **delights your customer** on an ongoing basis?

What is Software Engineering?

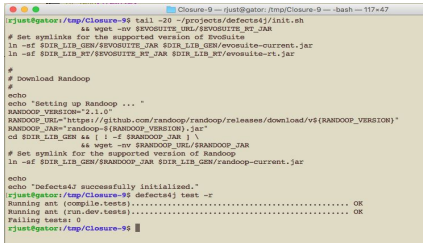
- Developing in an IDE
  and software ecosystem?



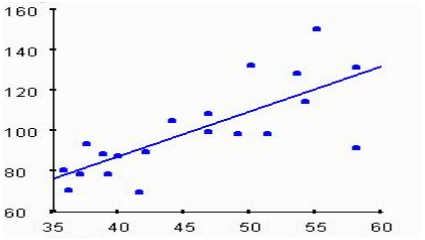
- Debugging and maintaining a software system?



- Deploying and running
  a software system?



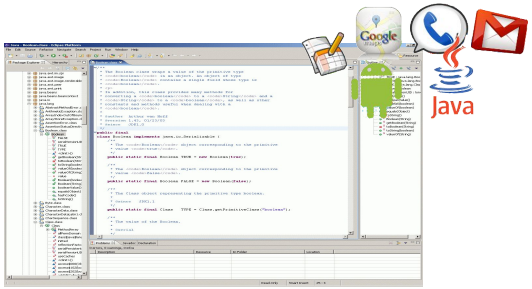
- Empirically evaluating a software system?



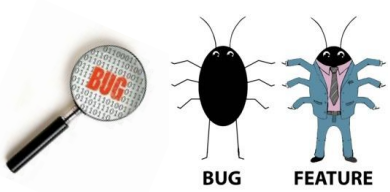
- Writing (design) docs?

What is Software Engineering?

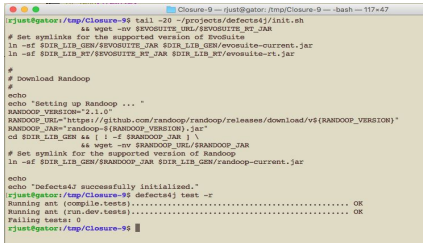
- Developing in an IDE and software ecosystem?



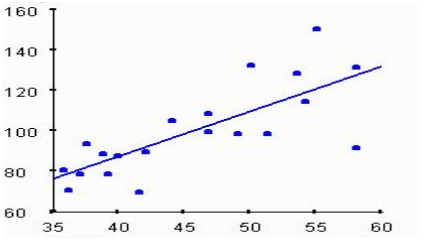
- Debugging and maintaining a software system?



- Deploying and running a software system?



- Empirically evaluating a software system?



- Writing (design) docs?

All of the above and much more!

Software Engineering is more than writing code

Software Engineering is the complete process of

specifying,
requirements engineering, specifications, documentation

designing, software architecture and design, UI

developing, programming (just one of many important tasks)

analyzing,
testing, debugging, linting, verification, performance engineering

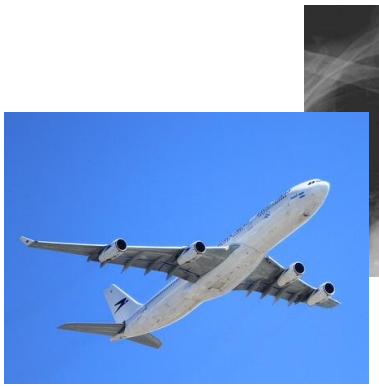
deploying,
DevOps, CI, packaging, operation, remote diagnostics,
documentation, websites

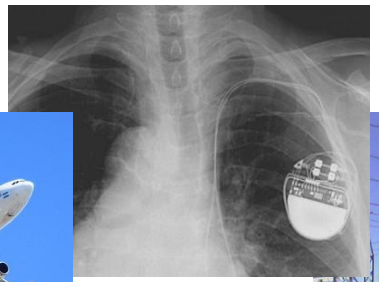
& maintaining refactoring, extensions, adaptation, issue tracking

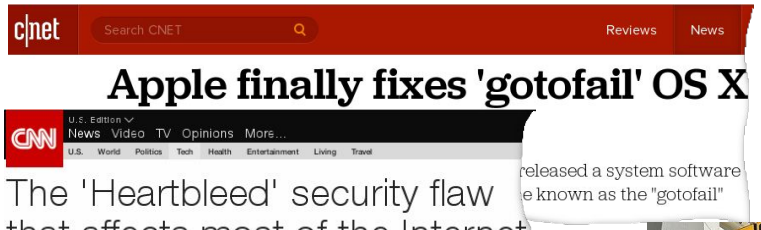
a software system. nearly every system contains software

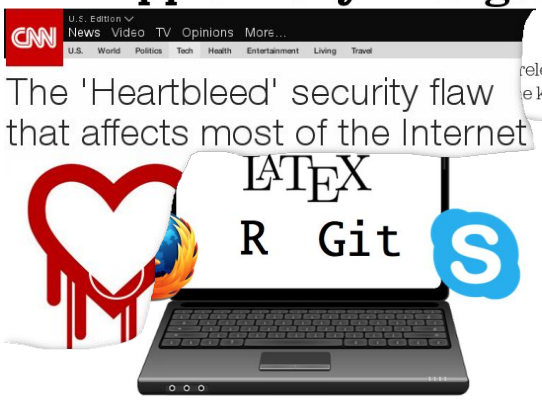
Why is Software Engineering important?

Software is everywhere!



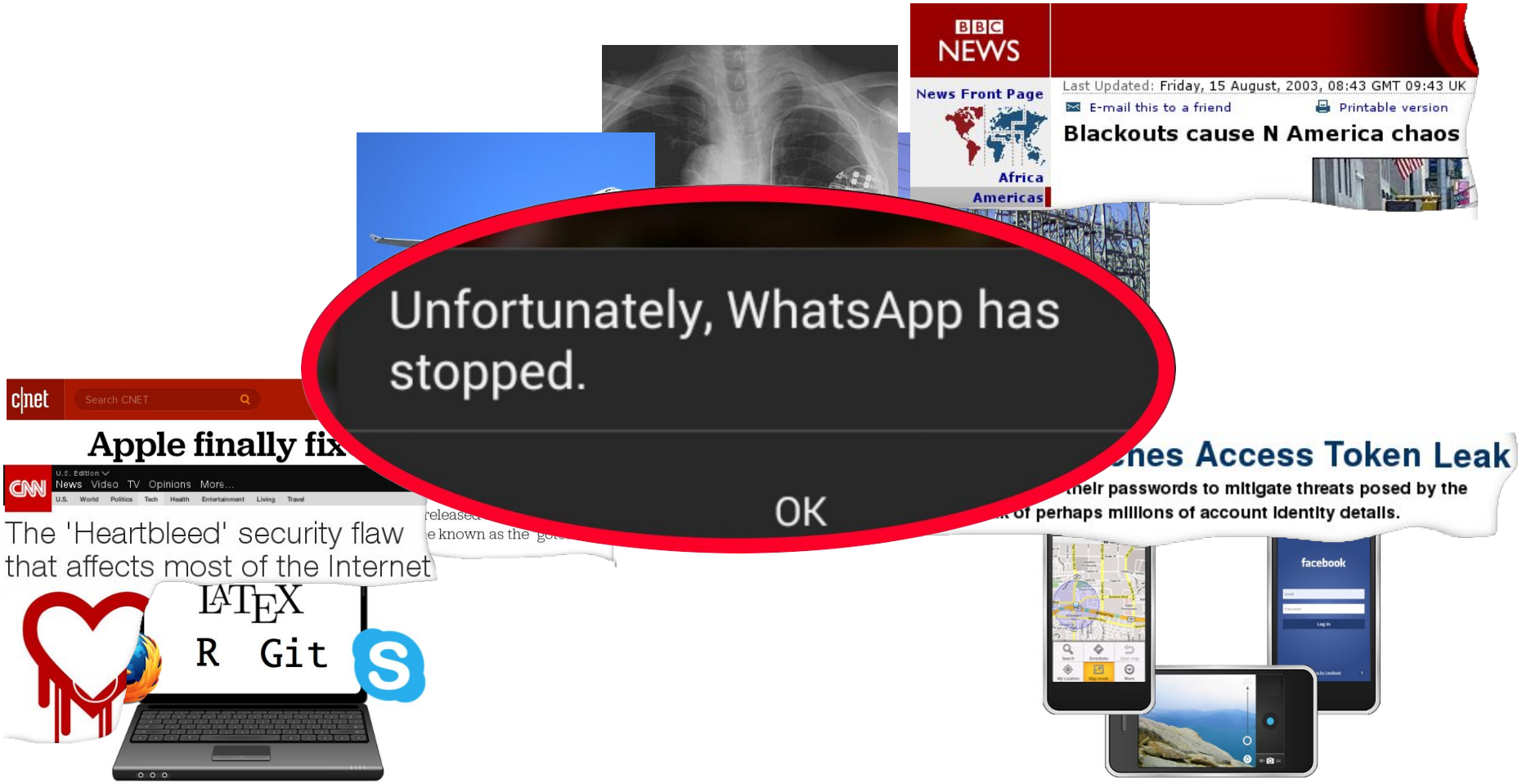






Why is Software Engineering important?

Software is everywhere!



Summary: Software Engineering

What is Software Engineering?

- The complete process of specifying, designing, developing, analyzing, and maintaining a software system.

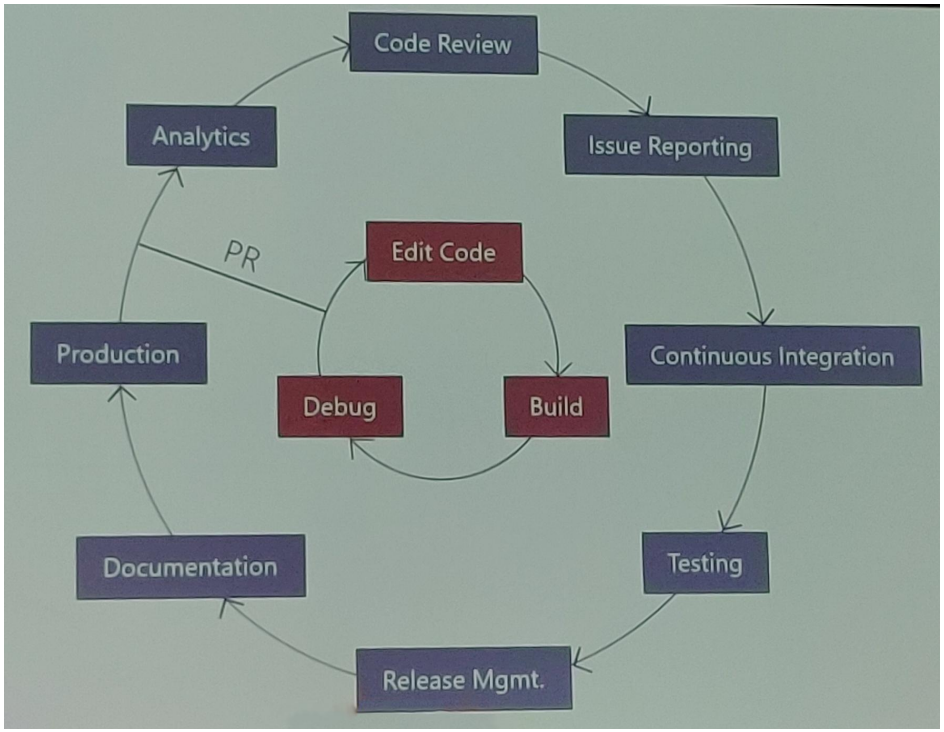
Why is it important?

It is a path to a successful product!

- Decomposes a complex engineering problem.
- Organizes processes and effort.
- Improves software reliability.
- Improves developer productivity.

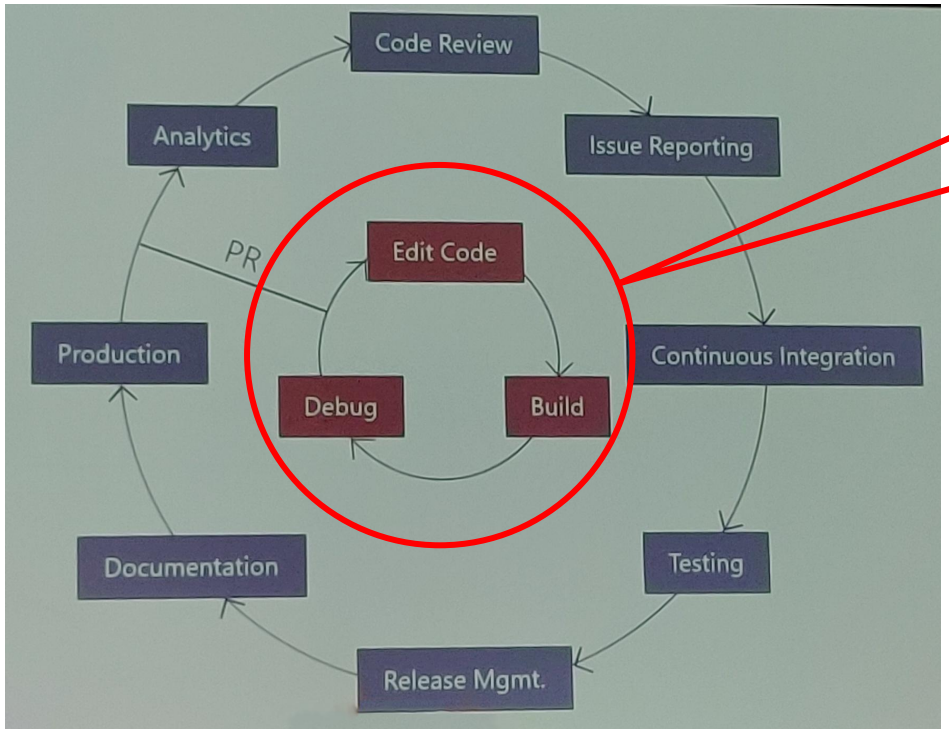
Both **technical** and **management** contributions are essential.

The Role of Software Engineering in Practice



(Engineering workflow at Microsoft, Big Code summit 2019)

The Role of Software Engineering in Practice



Intro-level courses focus on the inner loop.

(Engineering workflow at Microsoft, Big Code summit 2019)

CSE 403 largely focuses on the outer loop.

What can you learn in CSE 403?

- Learn software development best practices
- Understand how software is produced – from conception to continuous development and release
- Develop skills to effectively collaborate with others towards a common delivery goal
- Experience the responsibilities, issues and tradeoffs involved in making decisions as software engineers

Grounded by working as a team to incrementally deliver a real software product/service

CSE 403 vs. an internship

Internship:

- Get paid
- See the real world, warts and all

CSE 403:

- Choose your own project
- Significant input into the development process
- All aspects of the project: conception to deployment
- Detailed feedback and support

After an internship, there is still more to learn and practice. Both are valuable experiences. With a good attitude, you will benefit, such as honing your mentorship skills.

Today

- The CSE 403 team
- Logistics and Background
- What is Software Engineering
- Course overview and expectations

5 meetings per week

- **Lectures**: MWF; some in-class activities, especially on Fridays
- **Team meeting**: Tuesday
- **Meet with your TA**: Thursday

The first week and a half have a different schedule.

Logistics: resources

- **Course website**: cs.uw.edu/403

  All relevant information is on the website, or linked from it.
- Submit assignments via **Canvas**: https://canvas.uw.edu

Course overview: grading

Grading

- 50%: Course project
  - 70% project milestones
  - 30% final project review
- 20%: In-class exercises and individual assignments
- 15%: Midterm exam
- 15%: Participation
  - Engagement in project meetings
  - In-class discussions and activities (polls, small-group activities, etc.)
  - Discussion board contributions
- No final exam
  - Final presentations in final exam slot

Course overview: workload and topics

Workload

- One project assignment each week
- Readings
- In-class exercises

Course overview: topics and project

- **Software processes, requirements, and specification**
  - Range of software development processes.
  - Precise capture of requirements and specifications.
- **Software development**
  - Decompose a complex problem and build abstractions.
  - Improve your coding skills.
  - Effectively use version control, build systems, and code review.
  - Continuously develop and integrate code.
- **Software testing and debugging**
  - Effective, complete, and automated tests
  - Modern testing and debugging techniques
- **Deliver a significant product as part of a technical team**

You will experience

- Be exposed to some of the best software development practices in use today
- Understand how software is produced – from conception to continuous development and release
- Develop skills to effectively collaborate with others towards a common delivery goal
- Experience the responsibilities, issues and tradeoffs involved in making decisions as software engineers

Class and team expectations

Participate
Engage
Take initiative
Be respectful
Be responsible
Communicate
Reflect
Improve
Deliver

Course project overview

Teamwork

For 1 week, you will work on a pitch with a randomly-assigned partner.

For 9 weeks, you will work in a 6-person team. You may choose teammates, but that is not required.
Each person or team will rank the pitched projects.
The staff will assign people/teams to projects.
If a team is less than 6 people, then the staff will add more team members who are interested in the same project.

Course project: example categories

Create a *new* program/app/feature that scratches your itch.

Example categories

- Productivity and convenience apps
- Optimization problems and data science
- Gaming and making
- Extensions or plugins to open-source software
- Software engineering research (prototypes & experiments)

Pitching a product

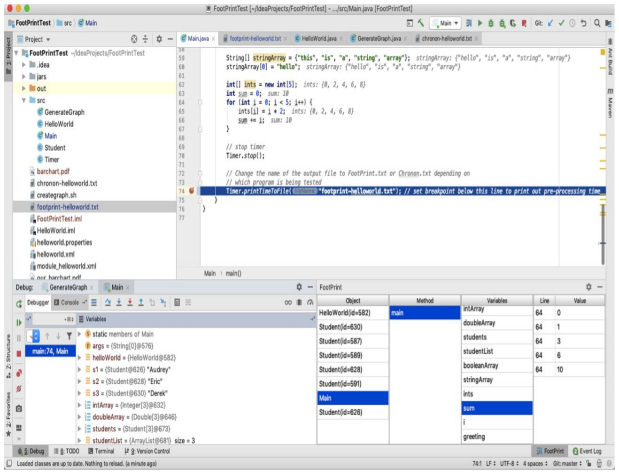
- This week you'll develop a product pitch
  - Identify a problem, a pain point, for some target customer set, in some area, that you can solve with technology – what's the value proposition of your solution?
- A subset will be selected to move forward (Shark Tank)
- You'll join a project team
- The rest of the quarter, you'll work to develop the product with your team, with incremental deliveries including demos
  - Weekly milestone deliverables
  - Tues section for team meetings
  - Thurs section for project meetings (with TA)

Some example products

- All-in-one student to-do list (canvas, class websites, calendar)
  - Plugin using AI to automatically add code comments
  - CallHome reminder; topic ideas from calendar, news
  - HowTheyVote tool to identify Congress voting history
  - Smart music or video recommendations

- App to split roommate costs/payments

- DuoCode (inspired by Duolingo) to learn coding



- Plugin to view history of variables

Common challenges for CSE 403 students

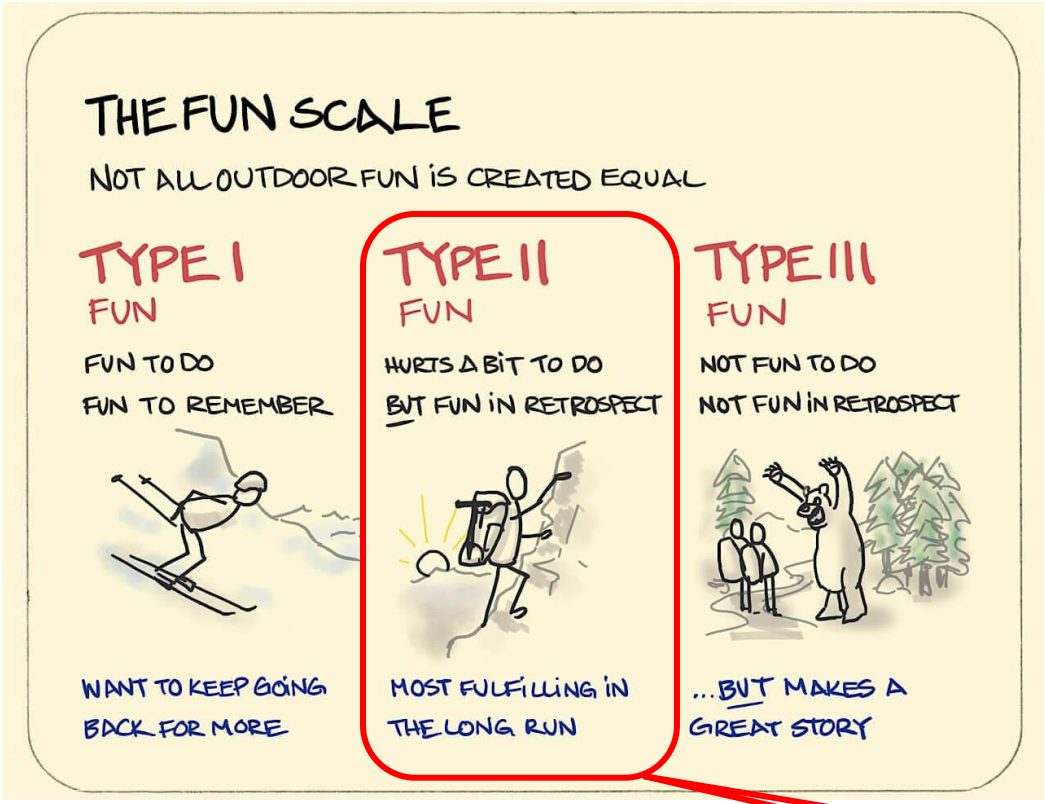
- Teamwork
  - Effective communication and coordination (#1 challenge)
  - Different backgrounds, skills, and incentives
- Complexity
  - Tooling and technology stacks
  - Scale of code base and code integration
- Uncertainty
  - Trade-offs, decisions, and justifications
  - No simple check-box grading

Assignment 1 – Project Proposals

- Prepare a product pitch in teams of 2-3
  - Think about a problem you'd like to solve
  - Consider what's already in play and available
  - Pitch a solution and its high-level technical approach
  - Use today and tomorrow's section to [form a group and] work together;
    Identify your group via class form by 1/7 11:59pm (see Calendar)
- Turn in pitch: Monday, 11:59pm
- View others' pitches
- Rank your preferences and join a team

See Canvas and the class calendar for more specifics

CSE 403: mostly type II fun



Sweet spot for teaching

Expectations

- Ability to program (in any programming language).
- Active participation in discussions.
- Teamwork and communication.
- Reflect on and improve your submissions.
- Go beyond adequate.

CSE 403: challenges for students

Team work

- Effective communication and coordination
- Different backgrounds, skills, and incentives

Complexity

- Tooling and technology stacks
- Scale of code base

Uncertainty

- No simple check-box grading
- Trade-offs, decisions, and justifications

CSE 403: challenges for students and staff

The Week-1 rush

Lecture time (12:30)

What's next?

- The Joel Test: basic software engineering processes
- Work on project proposal

Find your pitch partner