CSE 403 Software Engineering

Course Introduction

Winter 2025

Today's Outline

- The CSE 403 staff
- What is software engineering
- Course overview and expectations
- Assignment 1 – project proposals

About me

Gail Alverson, Ph.D., UW Affiliate Professor

About our Wi25 TAs

Celestine Buendia

Melanie Kneitmix

Taryn Neal

Connor Reinholdtsen

About Wi25 CSE 403 students

Let's hear about you!

So just what is Software Engineering?

Consider what you, a software engineer, must do to **deliver** a product/service that **delights your customer** on an ongoing basis

So just what is Software Engineering?

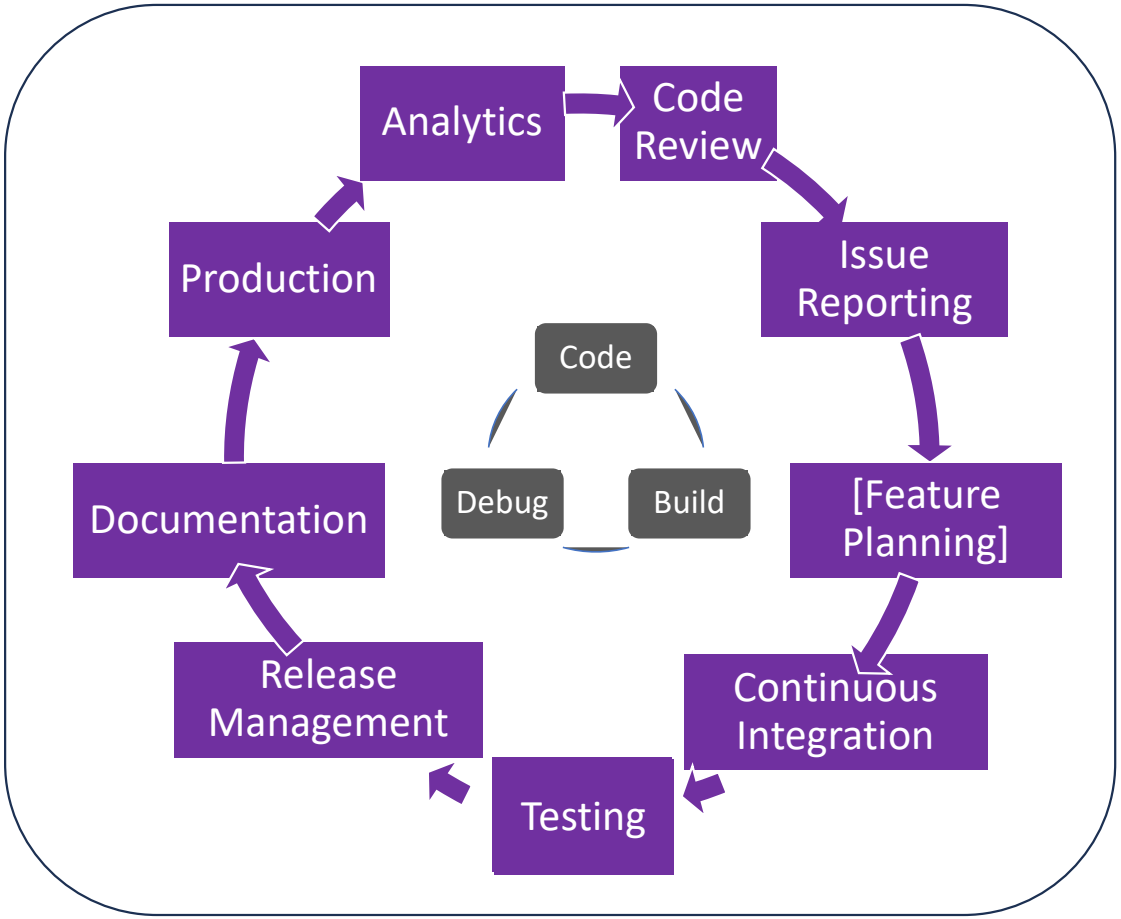
An **engineering discipline** concerned with all aspects of **software production** from the early stages of system specification through to maintaining [evolving] the system after it has gone into use. — Ian Sommerville

Software Engineering tasks include:

- Requirements engineering
- Specification writing and documentation
- Architecture and design
- Programming (Just one out of many important tasks! 🤯)
- Testing and debugging
- Deploying, operating, evaluating, refactoring and evolving
- Planning, teamwork and communication

Software Engineering in practice

Sample engineering workflow at Microsoft 2019



CSE intro-level courses focus on the inner loop

Our focus is largely the outer loop

Why is Software Engineering important?

Software is everywhere -- our lives depend on it

Good software engineering allows us to **deliver^2**

Why is Software Engineering important?

Software engineering is the complete process of specifying, designing, developing, analyzing and maintaining a software system

It is the path to a successful product

- Decomposes a complex engineering problem
- Organizes processes and efforts
- Improves software reliability
- Improves developer productivity
- Improves delivery of a solution that delights your customer

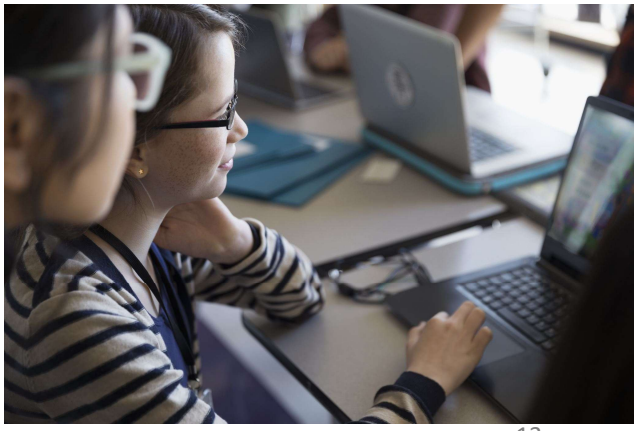
Course overview

Course overview: topics

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

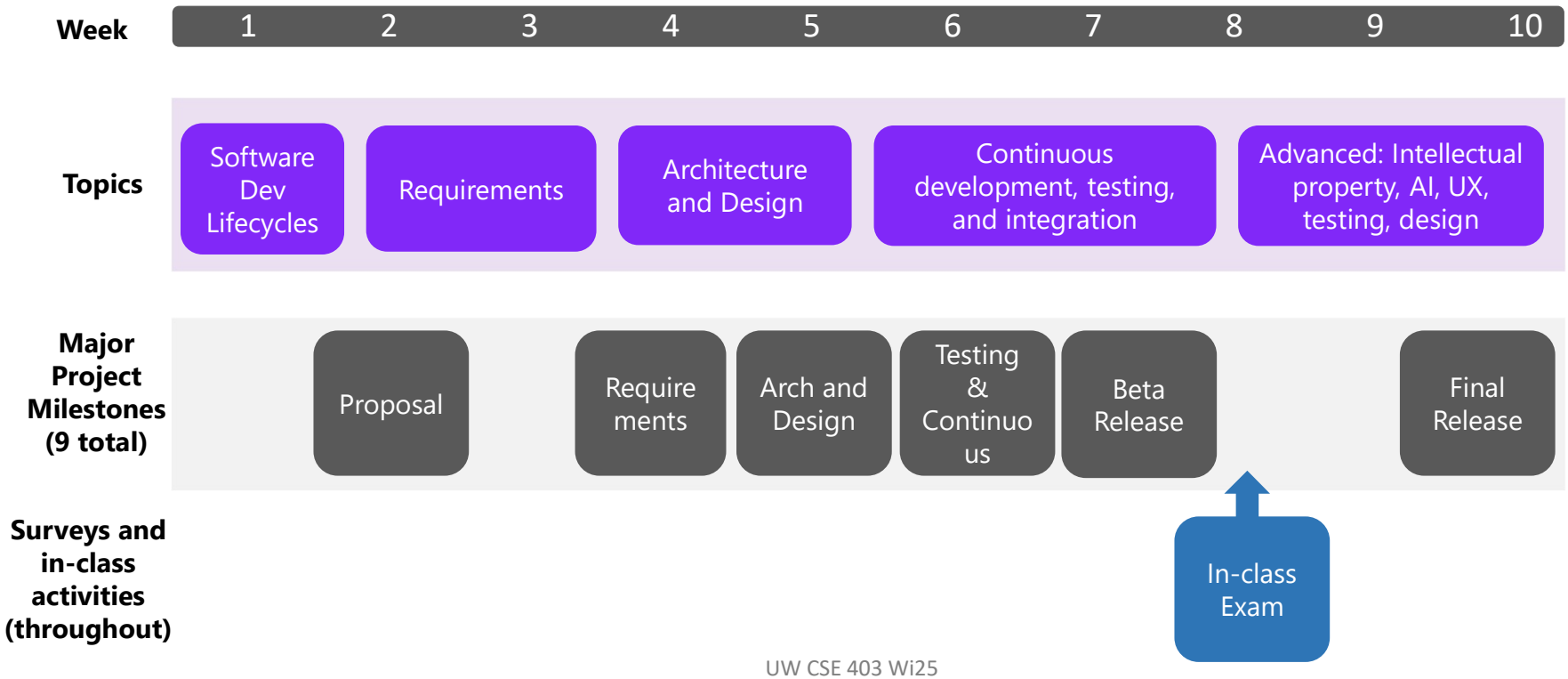
By the end of the quarter, you'll have…

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



Course overview: schedule

Important: See Calendar and Canvas for current details of topics and assignments



Course overview: grading

- 60% **Project milestones** (team)
  - 10% each (3): Requirements, Architecture & Design, Beta release
  - 5% each (3): Testing & Continuous Integration, Beta++ release, Peer review
  - 15% Final release & team retrospective
- 10% **Project assignments**
  - 5% Project proposal (small group)
  - 5% Individual retrospective (individual)
- 5% **In-class exercises** (small group)
- 15% **Later-in-term exam** (individual)
- 10% **Participation** (individual)
  - Project team and meeting engagement
  - In-class polls, feedback-requests, Q&A, etc.

Class and team expectations

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

The project and assignment 1

What's the difference between a PROJECT and a PRODUCT

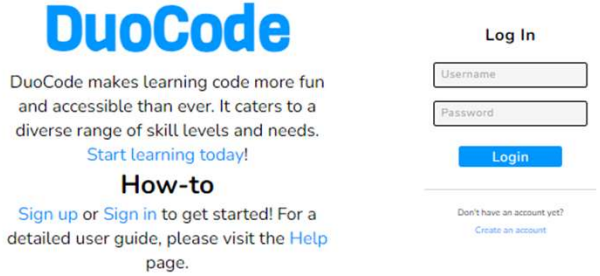
Double click on the CSE 403 project

- 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 (think, sharktank)
- You'll be assigned to a project team (you'll have input)
- 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)

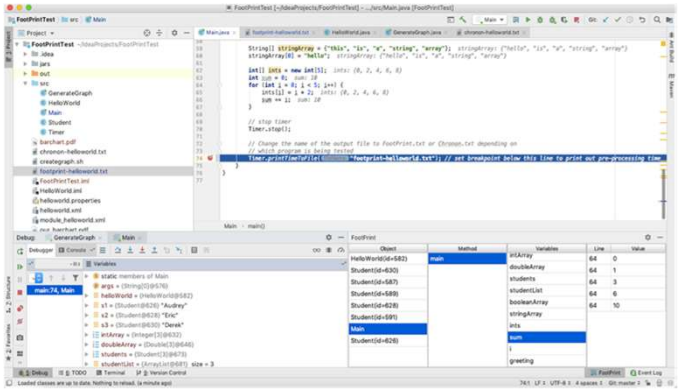
Some example products

- All-in-one student to-do list (canvas x class websites x calendar)
  - Plugin using AI to automatically add comments to code
    - CallHome reminder with topic ideas from calendar, news, etc.
      - 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 project challenges for 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

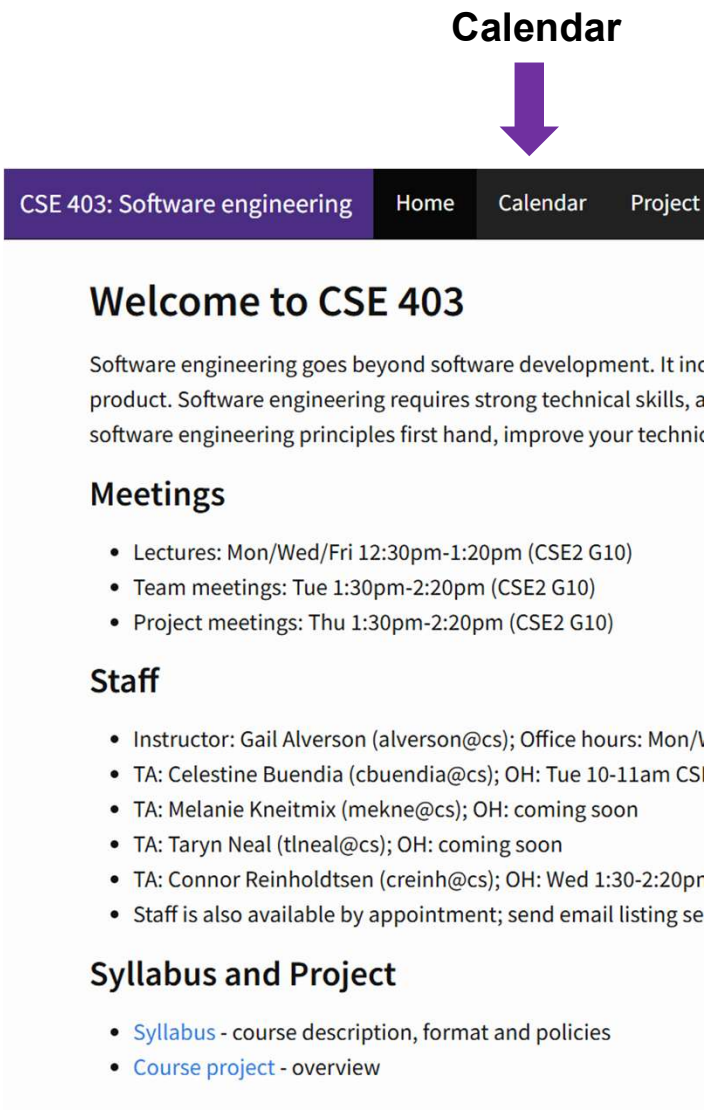
Assignment 1 – Project proposal

- 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 proposal: Monday 1/13, 11:59pm
- Present in-class and section: Tues-Thurs 1/14 to 1/16
- Rank your preferences: Fri 1/17, ~~11:59pm~~ 12:30pm (classtime) due to Mon holiday
- Learn your team and project! Tues 1/21 in section

See Canvas and the class calendar for more specifics

Course communication

- Website: https://courses.cs.washington.edu/courses/cse403/25wi
- Class discussions and announcements (Ed): https://edstem.org/us/courses/70255/discussion
- Assignments and turnin (Canvas): https://canvas.uw.edu/courses/1779835
- Direct questions to staff:
  - cse403-staff@cs.washington.edu
  - Email or Ed Chat
- Office hours



Questions

Additional material

CSE 403 vs internship

There are many commonalities!

Internship
- Get paid (usually)
- Get experience the real world and with real customers (+/-)

CSE 403
- Get significant input on what is the product you'll deliver
- Get significant input on your role in its development
- Get detailed feedback and support with learnings encouraged

You typically have less control of your path in an internship and lots of control in CSE403 but both are valuable experiences