Software Development Lifecycles – Agile Models

CSE 403 Software Engineering

Autumn 2023

Today's Outline

- Quick recap
  - Project Proposals
  - Software Engineering Lifecycles (SDLC)

- Agile SDLC Models
  - Agile
  - XP
  - Scrum

Project proposals

- Today (Mon) 11:59pm
  - Proposals due in Canvas (one submission per proposal-group)
  - Form with Project Name and Abstract due (see Ed Announcements for Link)
- Tues, Wed, Thurs
  - Pitches in class
  - Staff will publish the order by noon Tues (see Ed Announcements for list)
  - Staff will run the slides (from the submission)
- Thurs 11:59pm
  - Preferences survey due (see Ed Announcements for link)

Student preferences survey

1. Rank (highest to lowest) the projects you'd like to work on

   Top entry == project you'd most like to work on

2. [Optional] Identify one (or max two) other students that you'd like to be on a team with

   Note 1: Your requests and ranking must match the other students

   Note 2: This may affect which project you'll be placed on as there will need to be space

How we form the project teams

1. Staff first select the set of projects, those:
   - That students have found most interesting (higher ranked) and
   - That we think will be successful in our quarter class and
   - That balance the types of projects done in the class, so that you can see a range of projects developed.

2. If a project is selected to go forward, then students who proposed that project have priority for it (assuming that they ranked it their top preference).

3. Next, we will place other students on the selected projects:
   - We aim for groups of about 4-6 students per project.
   - We will try to assign you to a project with at least one student you have requested to work with, as long as that request was mutual.
   - We will try to give you one of your top ranked projects. But, just as in the real world, you may not get your first choice.

Back to SDLC - some traditional models

- Code and fix
- Waterfall model
- Prototyping
- Spiral model
- Staged delivery

Common stages

- Requirements
- Design
- Implementation
- Testing
- Release
- Maintenance

Let's try a poll in Poll Everywhere

pollev.com/cse403au

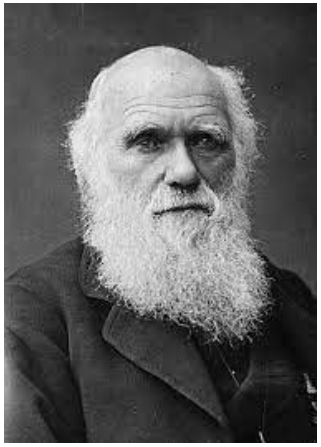
Rank the traditional SDLCs - from highest to lowest choice - that you'd choose to use for the class project (if you had to!). Team of 4-6, 9 weeks to develop, flexible requirements.

Total Results: 1

Onto Agile models

What is Agile all about?

Premise: the world is too uncertain, and we must be flexible and responsive to changes



There is nothing permanent except change -Heraclitus (Greek philosopher)

It is not the strongest or the most intelligent who will survive but those who can best manage change -Charles Darwin (English naturalist)

Agile Manifesto

A Behind the Scenes Look at the Writing of the Agile Manifesto

Agile Manifesto (http://agilemanifesto.org/):

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

See: Reading assignment 1 (due Tues 10/10/23)

While there is value in the items on the right, we value the items on the left more.

Agile models

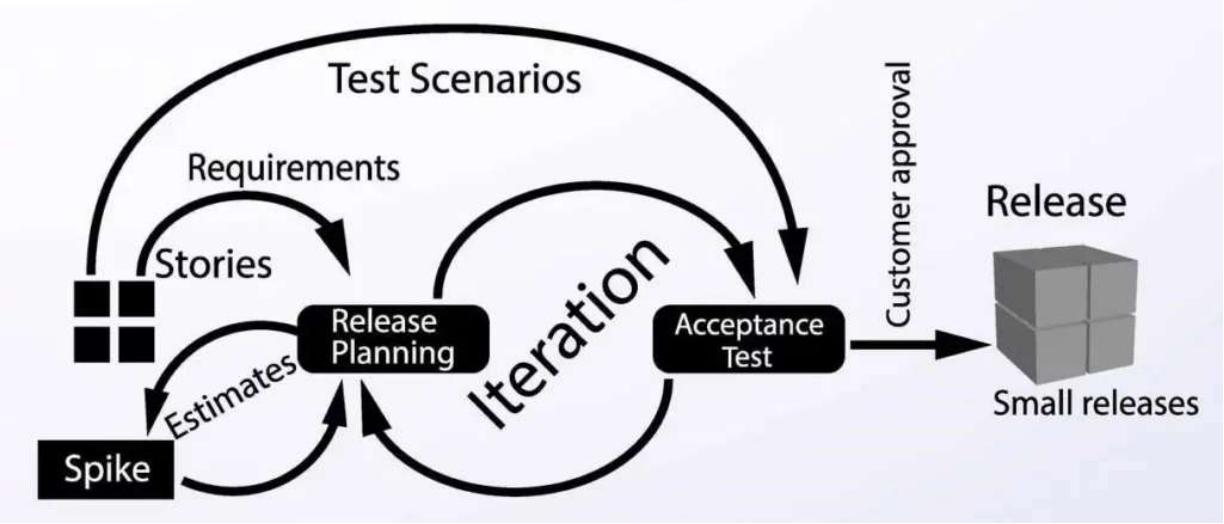
"Agile software development" is a general term for frameworks and practices outlined in the Agile Manifesto

Agile models

- Aim to deliver a high-quality product to the customer as fast as possible
- Focus on simplicity, excellence, continuous testing, integration
- Incremental and frequent delivery of working software
- Continuous customer involvement
- Expect requirements to change

http://agilemanifesto.org/principles.html

Agile SDLC: Extreme Programming (XP)



https://www.nimblework.com/agile/extreme-programming-xp/

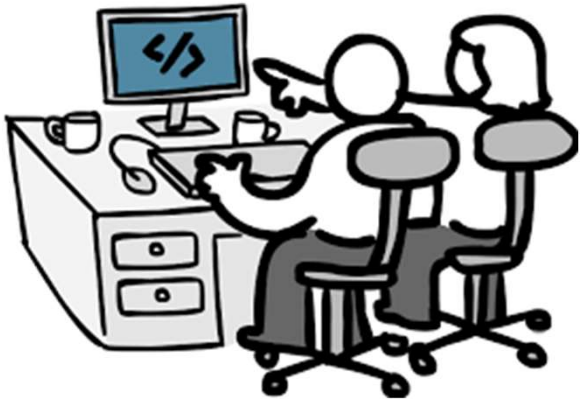
- SDLC emphasizes how engineers should work – good practices taken to an extreme
- Examples:
  - Continuous testing and integration
  - 10-minute build
  - Constant discussions with customers
  - Full flexibility to change requirements anytime
  - Pair programming
  - Test-driven development

XP Practice: Pair Programming

Pair programming – All production software is developed by two people sitting at the same machine

Provides for continuous code development, collaboration and review

Thoughts?

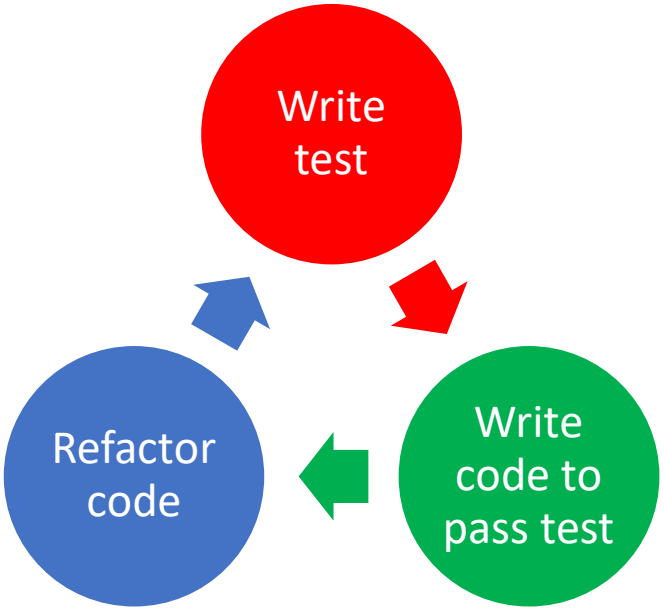


XP Practice: Test driven development

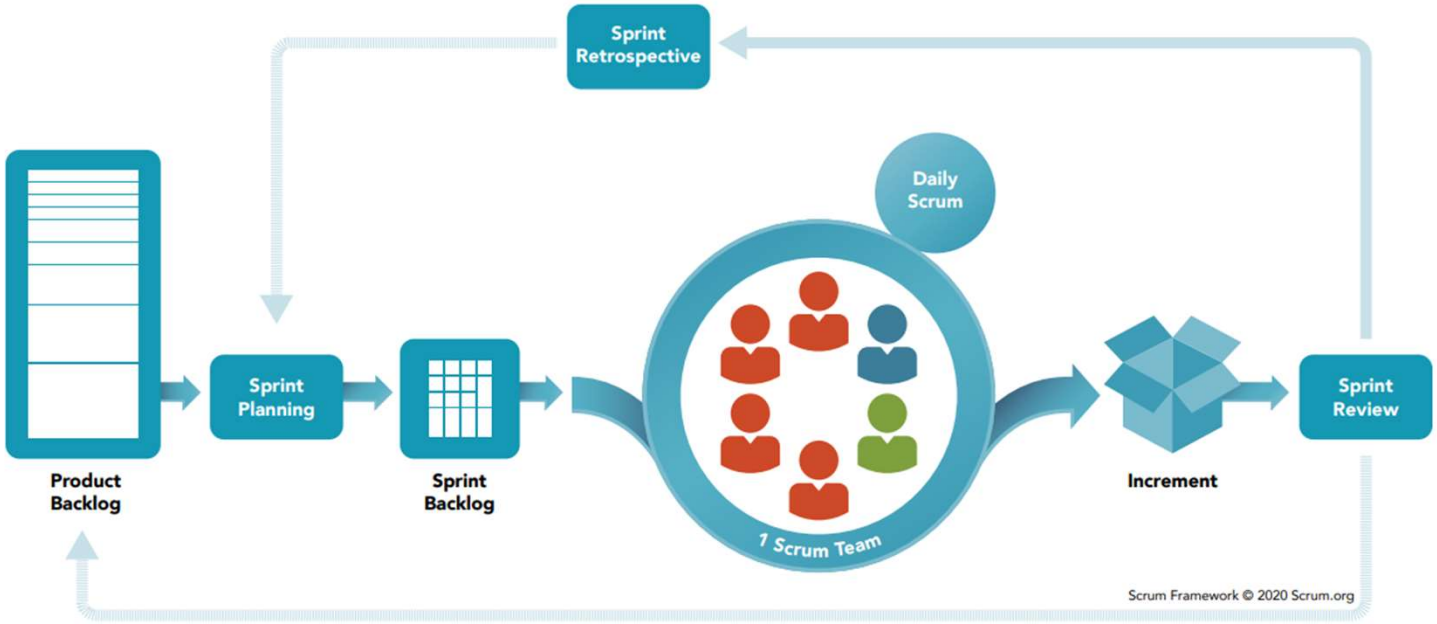
Write tests based on the requirements - before the production code is even written - and then develop code to make the tests pass

Tests run early and often

Thoughts?



Agile SDLC: Scrum



- See the analogies with XP?
- Scrum focuses on management and productivity
- XP addresses software quality and engineering techniques

Agile Summary

Pros

- Flexibility (changes are expected)
- Focus on quality (continuous testing)
- Focus on communication – with customers – with team

Cons

- Requires experienced management and skilled developers
  (e.g., responsible, proactive, communicate well)
- Prioritizing requirements can be difficult when there are multiple stakeholders
- Needs customer to be flexible in delivery (what / when)

Back to Poll Everywhere – pollev.com/cse403au

What SDLC would you pick and why?

- A control system for anti-lock braking in a car
- A hospital accounting system that replaces an existing one
- An interactive system that allows airline passengers to quickly find replacement flights
- New innovative but tbd features for a social media app
- Your 403 class project

⚠ When survey is active, respond at **pollev.com/cse403au**

What SDLC would you choose?

1 done

0 underway

When poll is active, respond at **pollev.com/cse403au**

A control system for anti-lock braking in a car

- Waterfall | Staged Delivery
- Prototyping
- Spiral
- Agile - XP | Scrum

Total Results: 1

When poll is active, respond at **pollev.com/cse403au**

A hospital accounting system that replaces an existing one

- Waterfall | Staged Delivery
- Prototyping
- Spiral
- Agile - XP | Scrum

Total Results: 1

When poll is active, respond at **pollev.com/cse403au**

An interactive system that allows airline passengers to quickly find replacement flights

- Waterfall | Staged Delivery
- Prototyping
- Spiral
- Agile - XP | Scrum

Total Results: 1

When poll is active, respond at **pollev.com/cse403au**

New innovative but tbd feature for a social media app

- Waterfall | Staged Delivery
- Prototyping
- Spiral
- Agile - XP | Scrum

Total Results: 1

When poll is active, respond at **pollev.com/cse403au**

Your 403 class project (ok to change)

- Waterfall | Staged Delivery
- Prototyping
- Spiral
- Agile - XP | Scrum

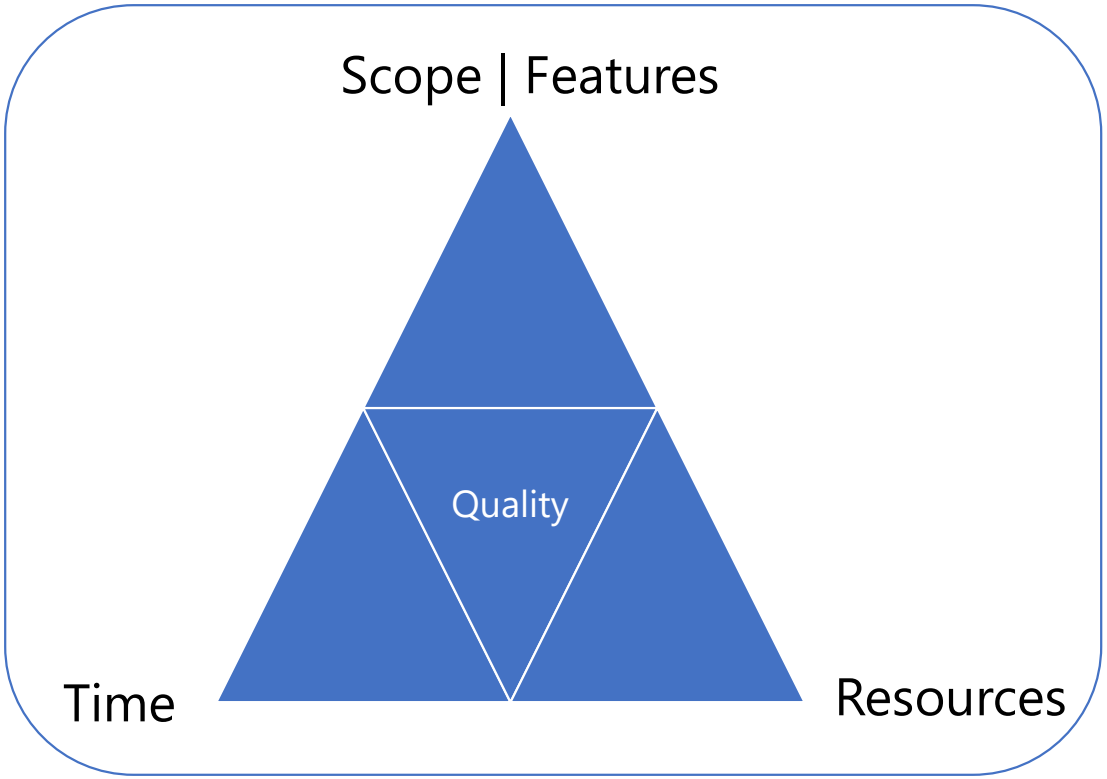
Total Results: 1

Why are there so many SDLC models?!

Choices are good ☺!

- The choice depends on the project context and requirements
- All models have the same goals: manage risks and produce high quality software
- All models involve the same general activities and stages (e.g., specification, design, implementation, and testing) and can be tailored
- Today's models involve customer feedback and the ability to adapt to changing requirements

A last note – a project management tool



- Software projects must balance what's delivered, when, and with what resources
- When there are changes to one axis, at least one other has to adapt
- These are also good considerations when choosing a SDLC model or adapting to a changing environment

Questions?