

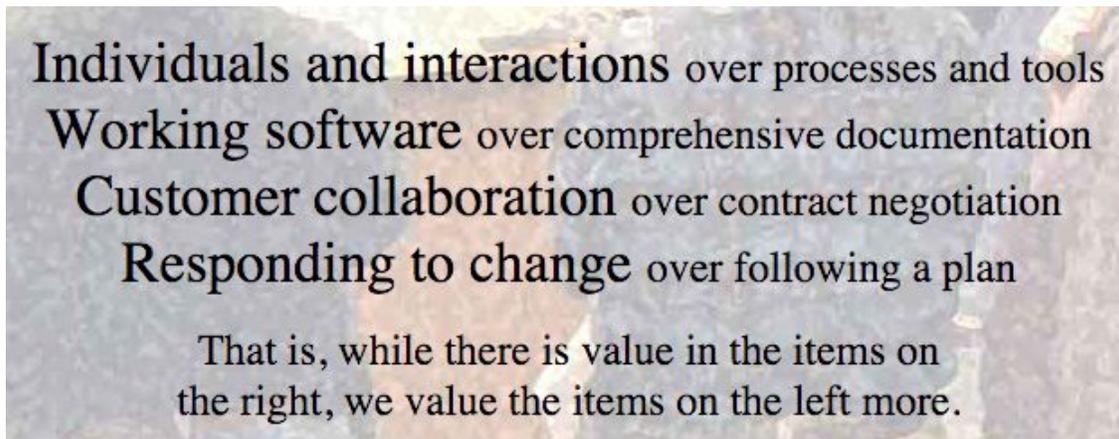
Agile notes

Tech notes describing an agile development methodology that we'll use in class

Agile and Scrum

Agile software development is a set of principles for software development as defined by the Agile Manifesto. Scrum is one example of an agile methodology.

Agile is a development process defined by its manifesto, www.agilemanifesto.org/:



www.agilemanifesto.org/principles.html - there are also 12 lower-level agile principles; these are worthy!

CSC 694 Project

We are small, so we'll be simple-agile in 694. This means:

- **Sprint** - We'll do three coding sprints of 2 weeks each.
- **Deliver** - Each coder will sign up for specific deliverables for the sprint.
- **Stoplight** - During the spring, I'll frequently ask you the stoplight question: Are you **red** (help!), **yellow** (worried, but OK), or **green** (all good)?
- **Pairs** - I'll try to sneak in at least some pairs programming, so you can give it a try. Maybe we'll do it in class one Tuesday. We'll see.
- **Code review** - Every line of code will be reviewed by Prof Bill and a peer. We'll also look at code in class. The answer here: only submit quality, code with great comments and formatting.
- **Test review** - I will review your tests as well.

That's it, pretty much.

Agile is supposed to be easy. And it is.

We'll succeed if we each own this project and work together as a team.



Stoplight Status:

Red=HELP! Yellow=worried. Green=All good.

The rest of this

You can stop here.

The rest of this is from my 495 class last term. It's overkill. But if you want to read more about agile and scrum, continue on!

www.amazon.com/B007P5N8D4 - "Scrum: a Breathtakingly Brief and Agile Introduction" by Chris Sims and Hillary Louise Johnson; I have it (\$1 Kindle version); it's excellent and only 54 pages, as it should be!

www.amazon.com/B015QA141W - "Scrum for Newbies" by Jeremy Wilson; I have it (Kindle \$3), but haven't read it yet

Other book possibilities:

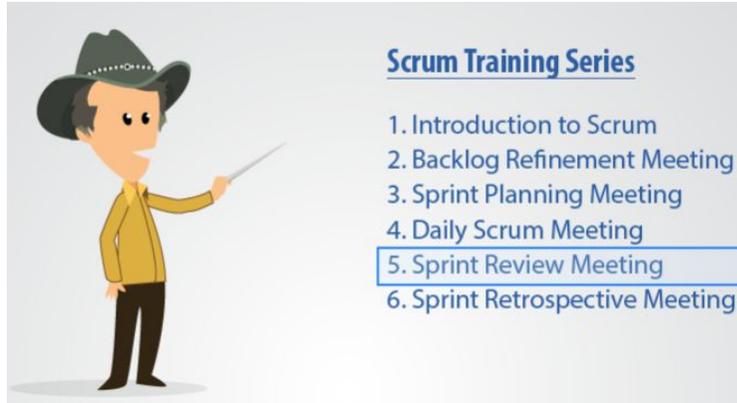
- www.amazon.com/B008NAKA5O - "Essential Scrum" by K Rubin; looks good and all, but why is it 500 pages to describe an agile process?
- www.amazon.com/B00JI54HCU - Scrum: The Art of Doing Twice the Work in Half the Time by Sutherland, 2014. Only 258 pages, but dinged in reviews for not actually covering Scrum?!?

www.scrumguides.org - the "official" definition of Scrum

www.scrummethodology.com/ - nice overview; watch these videos when I get a chance

www.scrumtrainingseries.com/ - a good video series; watched these and here are my notes:

1. **Introduction to Scrum** - Strong intro! "Today is the dumbest day of the rest of your project", desired team size is (7 +- 2), "a PowerPoint presentation about hypothetical things"
2. **Backlog Refinement Meeting** - INVEST= {Independent, Negotiable, Valuable, Estimable, Small, Testable}, timeboxed meetings, "potentially shippable form", user story/use case, very funny!
3. **Sprint Planning Meeting** - 2 hours, plan tasks for 2 week sprint
4. **Daily Scrum Meeting** - No phones, standing up, ideal sprint task is 1 man-day or less
5. **Sprint Review Meeting** - agenda={demo, product owner declares what's done, measure velocity (optional), stakeholder feedback}, no "partially done" tasks
6. **Sprint Retrospective Meeting** - team growth, "a learning team",



www.scrumreferencecard.com/ - 6 page reference card? Not good

www.agilenutshell.com/ - this guy has a book: The Agile Samurai,
www.amzn.com/1934356581; I have it but haven't read it yet

en.wikipedia.org/wiki/Pair_programming - Pair programming can be a part of Agile, "Pairs spend about 15% more time on programs than individuals. However, the resulting code has about 15% fewer defects." and "survey: 95% of programmers enjoyed working in pairs, 95% were more confident in their solutions"; So, the question is: try something new/fun/different because that's what school is for? Let teams decide?