Working In Teams vs. Individually

CS 169 Fall 2012

Armando Fox & David Patterson

It Takes a Team: Size, Scrum, and Documentation*(ELLS §9.1)*

David Patterson

*

Teams: On Wrestling Team in High School & College

*

SW Eng now a Team Sport

- Now in Post-Superhero-Programmer Era

- Rising bar of functionality/quality => cannot do SW breakthrough alone

- Successful SW career => programming chops AND plays well with others AND can help make *team* win

- *"There are no winners on a losing team, and no losers on a winning team."*

- Fred Brooks, Jr.

Scrum: Team Organization

- "2 Pizza" team size (4 to 9 people)
- "Scrum" inspired by frequent short meetings
- 15 minutes every day at same place and time
- To learn more: *Agile Software Development with Scrum by Schwaber & Beedle*

Daily Scrum Agenda

•Answers 3 questions at “daily scrums”:

1. What have you done since yesterday?
2. What are you planning to do today?
3. Are there any impediments or stumbling blocks?

•Help individuals by identify what they need

Scrum roles

- Team: 2-pizza size team that delivers SW
- ScrumMaster: team member who
- Acts as buffer between the Team and external distractions
- Keeps team focused on task at hand
- Enforces team rules
- Removes impediments that prevent team from making progress

Scrum roles (cont'd)

- Product Owner: A team member (not the ScrumMaster) who represents the voice of the customer and prioritizes user stories

*

Scrum Summary

Basically, self-organizing small team with daily short standup meetings

Work in “sprints” of 2-4 weeks

Suggest member rotate through roles (especially Product Owner) each iteration

Which expression statement regarding SCRUM is TRUE?

□ Scrum is at its best when it is difficult to plan ahead

□ Scrum is good for safety critical software as well as SaaS

□ Scrum implies Agile software development practices like TDD

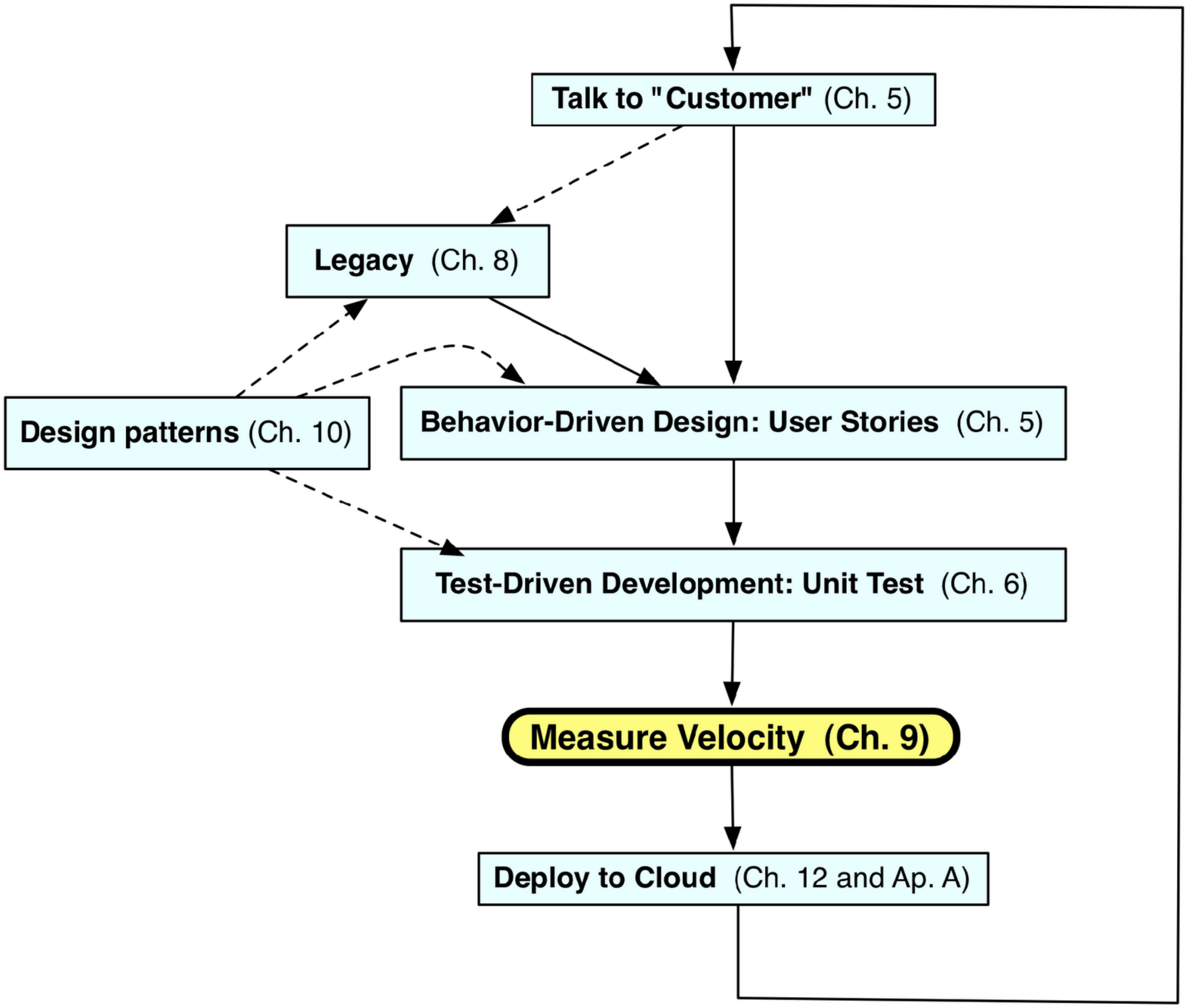
□ All the above are true

Points, Velocity, and Pivotal Tracker *(ELLS §9.2)*

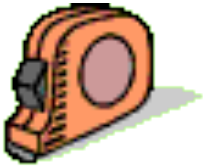
David Patterson

*

Agile Iteration: Where Are We?


Talk to "Customer" (Ch. 5)
Legacy (Ch. 8)
Design patterns (Ch. 10)
Behavior-Driven Design: User Stories (Ch. 5)
Test-Driven Development: Unit Test (Ch. 6)
Measure Velocity (Ch. 9)
Deploy to Cloud (Ch. 12 and Ap. A)

Measuring Productivity



- A measure of team productivity: calculate avg no. stories / week?
  - But some stories much harder than others
- Rate each user story in advance on a simple integer scale
  - 1 for straightforward stories, 2 for medium stories, 3 for very complex stories
- Velocity: avg number of *points* / week

More on Points

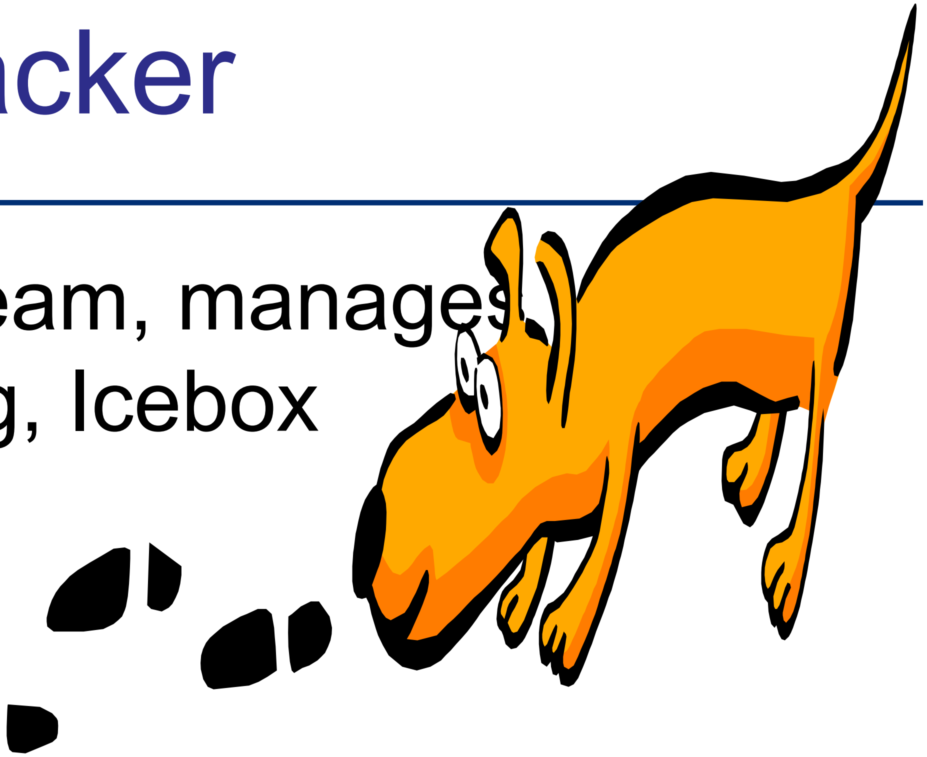
- Once get experience, Fibonnaci scale is commonly used: 1, 2, 3, 5, 8
- (Each new number is sum of previous 2)
- At Pivotal Labs, 8 is extremely rare
- Teams assign value: vote by holding up fingers simultaneously, take average
- If a big disagreement (2 and 5), discuss more

More on Points

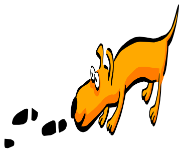
- ≥5 => user story should be broken up into simpler stories so that the backlog never has anything that's too demanding
- As long as team is consistent in evaluating stories, doesn’t matter if do 5 or 10 points per iteration

Pivotal Tracker

Calculates velocity for team, manages stories: Current, Backlog, Icebox

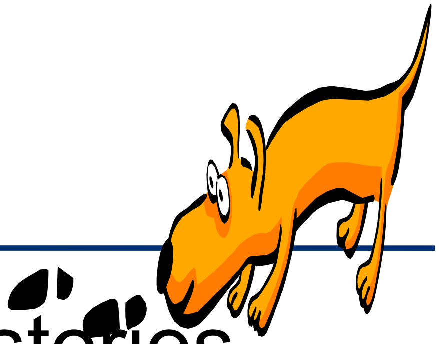


Pivotal Tracker



- Prioritize user stories by where place them in Current, Backlog, Icebox panels
- When completed, move to Done panel
- Can add logical Release points, so can figure out when a Release will really happen
  - Remaining points/Velocity
- Added *Epic* (with own panel)
  - Combine related user stories together
  - Ordered independent of user story in Backlog

Tracker Roles



- Developers don’t decide when user stories completed
- Pushes Finish button, which sends to “Product Owner”
- Product Owner tries out the user story and then either hits
- Accept, which marks user story as done, or
- Reject, which marks story as needing to be Restarted by developer

Team Cyberspace Whiteboard

Tracker allows attaching documents to User stories (e.g., LoFi UI)

Wiki with Github repository

Google Documents: joint creation and viewing of drawings, presentations, spreadsheets, and text documents

Campfire: web-based service for password-protected online chat rooms

Which expression statement regarding Points, Velocity, and Tracker is TRUE?

□ When comparing two teams, the one with the higher velocity is more productive

□ When you don’t know how to approach a given user story, just give it 3 points

□ With Tracker, developers pick the user stories and mark as Accepted when done

□ Tracker helps prioritize and keep track user stories and their status, calculates velocity, and predicts software development time

Design Reviews and Code Reviews*(ELLS §9.4)*

David Patterson

*

Design/Code Reviews

- Design review: meeting where authors present design with goal of quality by benefiting from the experience of the people attending the meeting
- Code review: held once the design has been implemented
- In the Agile/Scrum context, since design and implementation occur together, they might be held every few iterations

Design/Code Reviews

•Shalloway*: formal design and code reviews often too late in process to make big impact

•Recommends instead have earlier, smaller meetings: "approach reviews".

•A few senior developers assist team in coming up with an approach to solve the problem

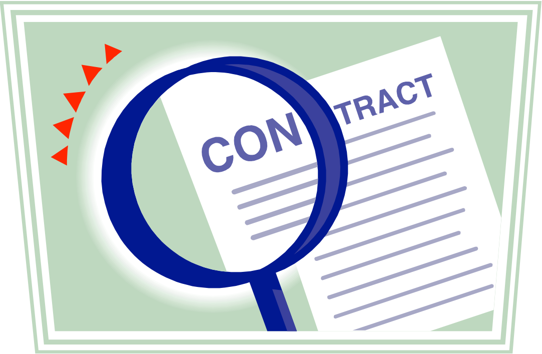
•Group brainstorms about different approaches

•If do a formal design review, suggests 1st hold a "mini-design review" to prepare

*Alan Shalloway, *Agile Design and Code Reviews*, 2002, www.netobjectives.com/download/designreviews.pdf

Code Reviews Can Check Comments too

- Challenge: keeping comments consistent with changes, refactoring



- Code review is one place to ensure comments make sense
- Advice on comments:
- Document what is not obvious from code
- Raise level of abstraction
- Explain why did something

Good Meetings: SAMOSAS

(Photo by K.S. Poddar. Used by permission under CC-BY-SA-2.0.)

- Start and stop meeting promptly
- Agenda created in advance; no agenda, no meeting
- Minutes recorded so everyone can recall results
- One speaker at a time; no interrupting talker
- Send material in advance, since reading is faster
- Action items at end of meeting, so know what each should do as a result of the meeting
- Set the date and time of the next meeting

Minutes and action items record results of meeting, start next meeting by reviewing action items

Reviews and Agile?

- Pivotal Labs – pair programming makes review superflous
- GitHub – *pull requests* replace code reviews
- 1 developer requests her code to be integrated with code base
- Rest of team see request and determines how affect their code
- Any concern leads to online discussion
- Many pull requests/day
=> many minireviews/day

Which expression statement regarding Design Reviews and Meetings is FALSE?

□ Intended to improve the quality of the software product using the wisdom of the attendees

□ They result in technical information exchange and can be highly educational for junior people

□ Design reviews can be beneficial to both presenters *and* attendees

□ Serving food like Samosas is vital to success of a good meeting