

Scrum Part 2

The End and the Beginning (of a sprint)

Topics & Ceremonies

- 1) Ending a Sprint
 - a) Sprint Demo
 - b) (Sprint) Retrospective
- 2) Starting a Sprint
 - a) Sprint Planning
- 3) Managing Stories
 - a) Story Time

Sprint "Ceremonies"

Daily Schedule for a One-Week Sprint

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
Sprint Planning 2hrs.	Stand-Up 15 min.	Stand-Up 15 min.	Stand-Up 15 min.	Stand-Up 15 min.
		Story Time 1hr		Sprint Review 1/2 hr
				Retrospective 1.5hrs

Ending a Sprint

Sprint Demo

- ..
- Report on stories that did not get finished
- Product owner gathers feedback from stakeholders
 - Do not make promises
 - Demo is lead by the product owner, but everyone contributes

Retrospective Steps

- Retrospective's Purpose

..

- 5 Steps:

1. ..

- State the goal:
"to identify one or two specific things to improve, and to create an action plan to implement those changes".
- Don't finger point, that shuts people down.
- Ensure everyone feels comfortable contributing.

Retrospective Steps (cont)

2. ..

- Create a time-line to identify what happened during the sprint.
- Can use sticky notes/index cards to order events.
- Helps remember whole sprint, from everyone's point of view.

3. ..

- Determine why things happened that way, avoiding finger-pointing.

Retrospective Steps (cont)

4. ..

- ..

- Just pick a couple:

- solving one makes you feel good;
- doing 20% on 5 makes you feel disheartened

- Try change for 1 sprint; review at next retrospective.

5. ..

- Appreciate team members

- Team members state something they appreciated:

"I appreciate <person> for <something>"

(voluntary!)

Inspect & Adapt

- Scrum is about inspecting and adapting.
 - Daily standup:
 - ..
 - keep on track.
 - Sprint demo:
 - ..
 - ensure most valuable features being added.
 - Retrospective:
 - ..
 - continuous improvement to the team.

Starting an Iteration

Sprint Planning Ceremony

- Sprint Planning Preconditions
 - ..
 - User stories have acceptance tests
 - These clarify what is expected
 - User story's sizes are estimated in terms of "points"
(from "story time" ceremony)

Sprint Planning Ceremony (cont)

- 1st half: What will we do?
 - ..
 - Choose amount of work = ..
 - Needs a solid, well groomed, well estimated backlog to be effective (story time)
 - For each story, review acceptance criteria to ensure shared understanding

Sprint Planning Ceremony (cont)

- 2nd half: How will we do it?
 - ..
 - Stories:
 - ..
 - Tasks:
 - ..
 - Tasks > 1/2 day likely too big; decompose further
 - Estimate tasks in "hours", "task points", or just "task count"
 - Experienced team can identify ~50% of required tasks.

Sprint Planning Ceremony (cont)

- Sprint Backlog
 - = ..
 - Committed stories for sprint
 - Sprint's tasks
 - Team improvement tasks (from retrospective)

Story Points and Estimation

Relative, not Absolute Estimates

- People are better at gauging relative size vs absolute size:
- Which is easier to answer?
 1. Which is taller, the CN tower or the Surrey's Central City tower?
 2. How tall is the Surrey Central City tower?

Relative, not Absolute Estimates

- Scrum estimates work..
- Start by coding some smallest task
 - Ex: a log-in screen, or a config file, etc.
 - Give it one point.
- Each other story's size estimated in terms of points:
 - "This seems like it's about 3 times as much work as the config file, so 3 points"
- Points not based on hours, as this is hard to give absolute estimate, but..

Story Time and Estimation Game

- Team meets each week for "Story Time":
 - Assign "story points" to up-coming user stories
- Estimation Game steps
 1. ...
 - Each team member takes turns doing one of:
 - Put up a new story up where they think best
 - Move a story that's up to improve order
 - Pass (if no changes needed)
 2. Team decides how many points each story is.
 3. Done!
 - Whole team agrees on the estimates!

In Class Exercise: Estimation Game

- Estimation Game

Let's play to assign story points

- At the front are user stories from an application
- When selected, you may:
 - put a new story on board, updating effort-order
 - move an existing story
 - pass (if all stories up and no corrections)

- Story Points:

- Left-most story = 1 point
- Let's assign thresholds for points at Fibonacci numbers (1, 2, 3, 5, 8, 13, 21, 34, ...)

Summary

- Sprint Demo
 - Demonstrate working software to stakeholders.
- Retrospective
 - Improve the team with 2 changes.
- Sprint Planning
 - Pick stories for the sprint.
 - Break stories into tasks.
- Story Time
 - Estimate size of stories on backlog