

CMPT 276 Class 06: Scrum and Agile Development

Dr. Jack Thomas

Simon Fraser University

Fall 2020

Scrum!



Image credit: [https://en.wikipedia.org/wiki/Scrum_\(rugby\)](https://en.wikipedia.org/wiki/Scrum_(rugby))

Today's Topics

1. Who does what in a **Scrum team**?
2. What does a **week** as a Scrum developer **look like**?
3. How does a Scrum team **estimate** work?

Image credit: <https://www.businessinsider.com/every-winning-slogan-from-us-presidential-campaigns-1948-2016-2019-5>

“Plans are worthless, but planning is everything.”
-President Eisenhower



Scrum: The Big Picture

- **BDUF is plan-driven:** first you create the plan, then development works to the plan.
- **Agile is planning driven:** each iteration you are constantly planning for the best path.
- **Some Changes in Agile (vs BDUF):**
 - Document as you go; only as needed
 - Deliver product early and often
 - Plan as you go: not just at the start of the project
 - Test as you go: not just at the end of project
- **Scrum** is one of many agile processes.

Key Items In Scrum

- **Backlog:** List of feature requests (“**User stories**”)
 - Ex: “As a cashier, I want to be able to apply a dollar amount discount to a single item so that I can accept a coupon.”
 - Highest priority features ("stories") are at the top of the backlog.
 - The team picks stories to complete from the top of the backlog.
- **Iteration:** a week (or 2-3) where team commits to deliver some user stories.
 - At end of iteration, team delivers working software.
 - The customer then tries out software & gives feedback
 - Use their feedback to plan the next iteration.

Scrum Roles: The A(gile)-Team



“I pity the fool
who [breaks
the build]”

I swear this reference was timely once

- There was a remake in 2010.
- It had Liam Neeson and Bradley Cooper and everything.

“I love it when a plan comes together”? Anybody?



Scrum Roles

- The scrum team identifies a few roles:
 - 1. Product Owner**
 - 2. Scrum Master**
 - 3. Team member**
- And we'll add in
 - 4. Repository Manager**

Role: Product Owner

- **Keeper of the Product Vision**
 - Understands needs of the customer
 - Works closely with the stakeholders to determine what needs to be built, and feature priorities
- **Manages Backlog**
 - Adds new feature requests
 - Prioritizes features: directs the team towards most valued work
- **Responsibility:** to maximize value to the business

Role: Scrum Master

- **The Coach**
 - Guides team towards more cohesion, self-organizing, performance
 - Scrum master's deliverable is the self-organizing team.
- **Not the boss:** Is a trusted advisor, and team-advocate
 - Pushes decision making back to the team
 - Removes impediments (Ex: getting equipment)
 - Expert on scrum process; facilitates meetings.
- **Responsibility:** a well-functioning Agile team.

Role: Team Member

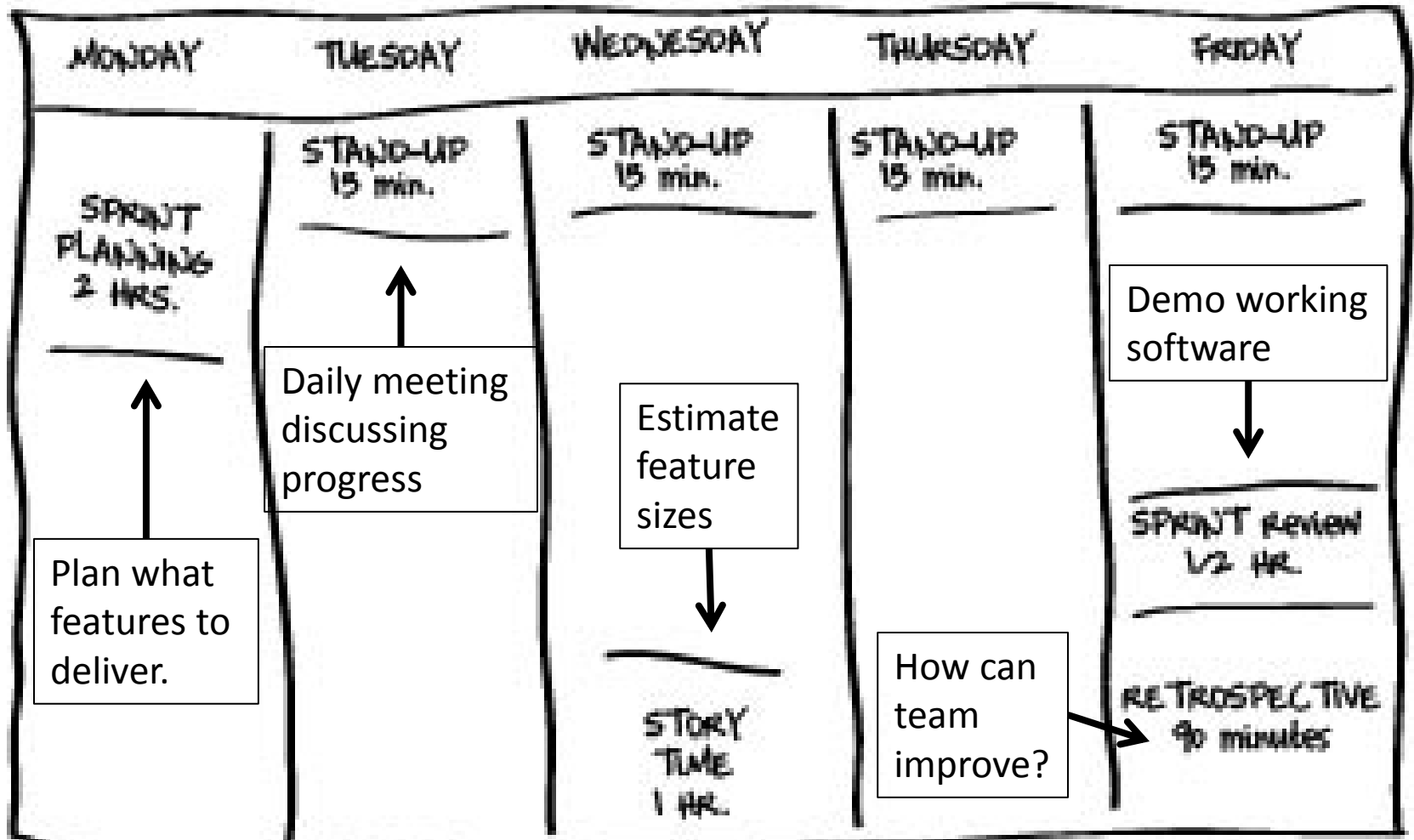
- **Teams are collaborative, self-organizing.**
 - Have total authority over how to get work done
 - Estimates work for each feature
 - 7 +/-2 team members
- **Responsibility:** Delivering stories that the team committed to.
 - No singled out "experts"; everyone does what is needed for this iteration.
 - No “silos of knowledge”

Optional Role: Repository Manager

- Not part of normal Scrum, added as part of a GitLab development work-flow.
- **Accepts merge requests when they are ready**
 - Helps resolve Git problems
 - Helps enforce code quality
- **Responsibility:** Ensures team's process is followed to commit code.
 - It must meet coding style; have unit tests, system pass all unit tests, code reviewed, etc.

Sprint "Ceremonies"

Daily Schedule for a One-Week Sprint



Sprint Planning (2 hours)

- Team pick **what stories to commit to** this iteration
 - **Backlog** must be **well maintained**: Each story has a "size" estimate.
- **Velocity**: Amount of work a team finished last iteration.
 - Team generally picks to do as much work this iteration as they completed last iteration.
 - Self correcting to become accurate at predicting performance.

Stand-Up Meeting (15 minutes)

- Daily. Only team members. Brief.
- Each team member briefly answers:
 1. What I **accomplished** since the last stand-up
 2. What I **expect to accomplish** by next stand-up
 3. What **obstacles** are slowing me down
- **Benefits:**
 - If "accomplished" < yesterday's "expected", then we know we are slipping.
 - Others jump in with brief, "I can help you with that"; but no long discussions of the details.
- Scrum master facilitates meeting, takes notes.

Sprint Review (30 minutes)

- **Demonstrate** working software to **stakeholders**
 - Report on stories that **did not get finished**.
- Product owner gathers **feedback** from **stakeholders**:
 - New **feature requests** or changes
 - Clarifies **priorities**
- **Make no promises!**

Retrospective (1.5 Hours)

- **Goals**
 - Identify one or two specific things to improve in the team.
 - Create an action plan to implement those changes.
- Team **reviews** what happened **during iteration**
 - Generates insights into what happened (**no finger pointing**)
 - Choose 1-2 **specific** improvements.
- **Appreciate** each other's work.

Story Time:

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?

For Example, I Thought I Could Fit All Of This Onto One Slide

- Scrum estimates work in terms of relative “story points”
- Start by coding some of the **smallest tasks**.
 - 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 are not based on hours, as this is hard to give absolute estimates, but based on effort.

Story Time And Estimation Game

- Team meets each week for "**Story Time**":
 - Assign "story points" to up-coming user stories
- **Estimation Game Steps:**
 1. Pin up-coming stories to wall, ordering by size. Each team member takes turns doing one of:
 - Put 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 when whole team agrees on the estimates.
- User stories will be pre-chosen in the group project.

Recap – Quick Notes On Being Agile

- **Scrum Team**

- Product Owner, Scrum Master, Team Member, (extra) Repository Manager.

- **Ceremonies**

- Sprint planning, Daily standup, Review, Retrospective, Story time

- **Estimation**

- Story points
 - Estimation game