

**15-412**

**Scrum**  
**Oct. 4, 2019**

**Dave Eckhardt**

# Outline

## Scrum methodology

# Scrum Methodology

## Scrum is an “agile software development methodology”

- 1986, Takeuchi & Nonaka, “New New Product Development Game”
- Goals
  - Replace older models such as “waterfall”
  - Generic “agile development” goals
    - » *Reveal* requirements via iteratively producing software artifacts
  - Increase communication among group members
  - Lots of “demo days”

# Key Parts

## People

- “Product Owner” - customer or Marketing
- Scrum Master – “in lieu of” manager / “not a leader”
  - Runs process, maintains data structures
- Team
  - Talents vary (UI, language expert, coder, documenter)

## Data structures

- Project backlog, sprint backlog, issue list
- Burn-down graph

## Time

- Sprint
- Scrum

# Time

## Sprint

- “Time-boxed” development cycle – 2 to 4 weeks
  - Goal: limit time-slip by *incrementally* determining extent
- Begins with a Planning Meeting
- Includes N Scrum Meetings, one per day
- Ends with Review Meeting (“demo day”), Retrospective

## Scrum

- “Extremely time-boxed”
- Daily 15-minute mini-meeting
  - What have you done since yesterday?
  - What do you plan to do today?
  - What is blocking you?

# Data Structures

## Project backlog

- List of desirable features
- Sorted by (customer) priority
- Dynamic!
  - Customers will modify their needs when they see demos

## Sprint backlog

- Team picks features from project backlog
- Features broken into 4-hour to 16-hour tasks
- Team members sign up for tasks

## Burn-down chart

- Daily chart of work remaining in current sprint

# 15-412???

## Suggested features

- High-level todo list, with minimum/hope/ideal breakdown
- Short-term todo list
  - Week is probably more appropriate than month for 412!
- Per-hack-session “did/expect/block” model
- Written “log” of “what we thought we could do in a week”
- Emphasis on thinking ahead to a “demo” (usable system)
  - Avoid the “pile of tasteful bits that doesn't do anything” outcome

# Summary

## The Scrum model

### Suggestions for applying similar ideas

- Think ahead
- Re-prioritize as you learn more about your work
- Use recurring time-based triggers to avoid getting bogged down