Slide 1

Agile Project Management – Applying *Scrum* to Botball

David Mutchler
Professor of Computer Science and Software Engineering

Rose-Hulman Institute of Technology
5500 Wabash Ave.
Terre Haute, IN 47803

mutchler@rose-hulman.edu
(812) 877-8426

www.rose-hulman.edu/~mutchler/gcer2012

Slide 2

Take-away (preview)

- Botball teams need Project Management
  - But Project Management is hard!
- Current focus:
  - Gantt Charts
  - Waterfall processes

- Another way:
  - *Agile* processes
  - **Simplified *Scrum* for Botball**

www.rose-hulman.edu/~mutchler/gcer2012

Slide 3

What is *Project Management*?

Henry Gantt (1861-1919), the father of planning and control techniques

Typical phases of an engineering project

Photo and diagram are from Wikipedia, http://en.wikipedia.org/wiki/Project_management

www.rose-hulman.edu/~mutchler/gcer2012

Slide 4

Project Management in Botball – *Gantt Chart*

www.rose-hulman.edu/~mutchler/gcer2012

Slide 5

Waterfall Project Management

Building:
- Cars ☺
- Software ☹

The diagram is from Wikipedia, http://en.wikipedia.org/wiki/Project_management. It was adapted from *Managing the Development of Large Software Systems*, by Winston W. Royce, 1970, http://leadinganswers.typepad.com/leading_answers/files/original_waterfall_paper_winston_royce.pdf

Waterfall was incorporated in DOD standards like the one shown above, http://www.product-lifecycle-management.com/download/DOD-STD-2167A.pdf

www.rose-hulman.edu/~mutchler/gcer2012

Slide 6

Agile Project Management

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
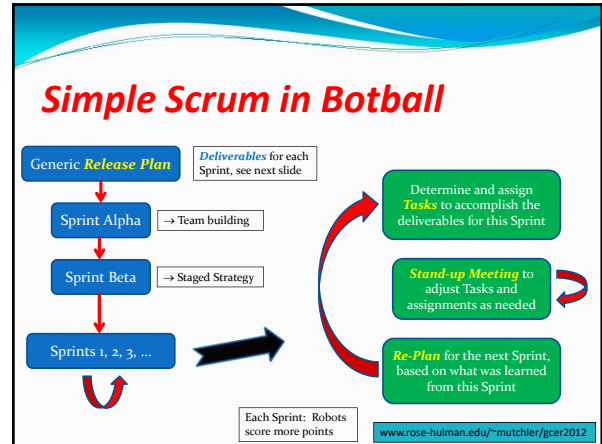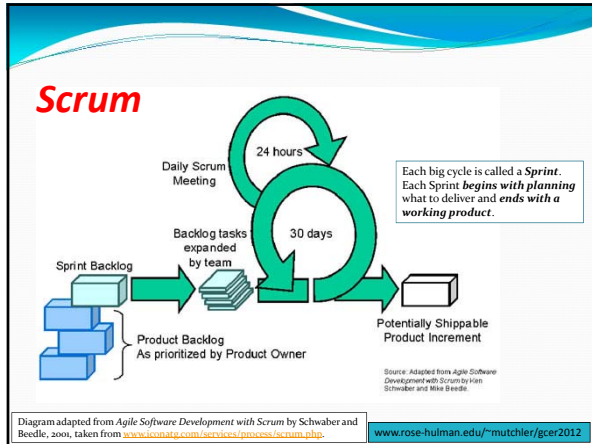Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

We follow these principles:

... **Deliver working software frequently** ...

... **Build projects around motivated individuals** ...

... **Working software is the primary measure of success** ...

... **Simplicity – the art of maximizing the amount of work not done – is essential** ...

These sound like good fits for Botball!

From http://agilemanifesto.org/ and http://agilemanifesto.org/principles.html.

www.rose-hulman.edu/~mutchler/gcer2012

Slide 1: Scrum

Scrum

24 hours

Daily Scrum Meeting

30 days

Backlog tasks expanded by team

Sprint Backlog

Product Backlog As prioritized by Product Owner

Potentially Shippable Product Increment

Each big cycle is called a *Sprint*. Each Sprint *begins with planning* what to deliver and *ends with a working product*.

Source: Adapted from *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle

Diagram adapted from *Agile Software Development with Scrum* by Schwaber and Beedle, 2001, taken from www.iconatg.com/services/process/scrum.php.

Slide 2: Simple Scrum in Botball

Simple Scrum in Botball

Generic *Release Plan*

Deliverables for each Sprint, see next slide

Sprint Alpha → Team building

Sprint Beta → Staged Strategy

Sprints 1, 2, 3, …

Determine and assign *Tasks* to accomplish the deliverables for this Sprint

Stand-up Meeting to adjust Tasks and assignments as needed

Re-Plan for the next Sprint, based on what was learned from this Sprint

Each Sprint: Robots score more points

Slide 3: A generic Release Plan for Botball

A generic *Release Plan* for Botball

- Sprint Alpha:
  - Plan for how to **build the team**
  - **Team rules**: participation, decision-making, conflict-handling, …
- Sprint Beta:
  - **Staged Strategies** for Seeding and Double-Elimination
- Sprint 1:
  - Robots **accomplish Stage 1** of Seeding Strategy
  - Robots **accomplish Stage 1** of Double-Elimination Strategy
- Sprints 2, 3, 4…
  - Robots **accomplish Stage 2…** of Seeding Strategy
  - Robots **accomplish Stage 2…** of Double-Elimination Strategy

Slide 4: After Sprint Beta

After Sprint Beta:

This and the next slides show an example of how the Release Plan might evolve. Only the Seeding Strategy is shown.

- Sprint 1:
  - Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
- Sprint 2:
  - Scores 12 x 10 = 120 points reliably by scoring 12 of 12 Tilapia.
- Sprint 3:
  - Scores 180 additional points reliably by scoring 6 (or more) of 10 Kelp.
- Sprint 4:
  - Scores 410 additional points reliably by scoring 7 (or more) of 10 polyps from opponent's side and 3 (or more) from our side into reefs and into MPA.
- Sprint 5:
  - Scores TBD to TBD points reliably by scoring TBD.

Slide 5: After Sprint 1

After Sprint 1:

Sprint 1: Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
Sprint 2: Scores 12 x 10 = 120 points reliably by scoring 12 of 12 Tilapia.
Sprint 3: Scores 180 additional points reliably by scoring 6 (or more) of 10 Kelp.
Sprint 4: Scores 410 additional points reliably by scoring 7 (or more) of 10 polyps from opponent's side and 3 (or more) from our side into reefs and into MPA.
Sprint 5: Scores TBD to TBD points reliably by scoring TBD.

- *Accomplished:*
  - Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
- Sprint 2:
  - Scores 12 x *6* = *72* points reliably by scoring *at least 6 of 12* Tilapia.
- Sprint 3:
  - Scores *60* additional points reliably by scoring *2* (or more) of 10 Kelp.
- Sprint 4:
  - *Increases above scores for Tilapia and Kelp*
  - Scores 410 additional points reliably by scoring 7 (or more) of 10 polyps from opponent's side and 3 (or more) of 10 polyps from our side into reefs and into MPA.
- Sprint 5:
  - Scores TBD to TBD points reliably by scoring TBD.

Slide 6: After Sprint 2

After Sprint 2:

Accomplished: Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
Sprint 2: Scores 12 x 6 = 72 points reliably by scoring at least 6 of 12 Tilapia.
Sprint 3: Scores 60 additional points reliably by scoring 2 (or more) of 10 Kelp.
Sprint 4: *Increases above scores for Tilapia and Kelp.* Scores 410 additional points reliably by scoring 7 (or more) of 10 polyps from opponent's side and 3 (or more) of 10 polyps from our side into reefs and into MPA.
Sprint 5: Scores TBD to TBD points reliably by scoring TBD.

- *Accomplished:*
  - Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
  - Scores 12 x *3* = *36* points reliably by scoring *at least 3 of 12* Tilapia.
- Sprint 3:
  - Scores 12 x *9* = *108* points reliably by scoring *at least 9 of 12* Tilapia.
  - Scores *60* additional points reliably by scoring *2* (or more) of 10 Kelp.
- Sprint 4:
  - Scores *60* additional points reliably by scoring *4* (instead of just 2) or more of 10 Kelp.
  - Scores *30* additional points reliably by scoring ~~7 (or more) of 10 polyps from opponent's side and~~ 3 (or more) of 10 polyps from our side ~~into reefs and~~ into MPA.
- Sprint 5:
  - *Scores additional points from polyps.*
  - Scores TBD to TBD points reliably by scoring TBD.

After Sprint 3:

- *Accomplished:*
  - Starts, leaves box, stops, moves over bumps on board and through turnstile, all very reliably.
  - Scores 12 x *11* = *132* points reliably by scoring at least *11* of 12 Tilapia.
  - Scores *30* additional points reliably by scoring 3 (or more) of 10 polyps from our side into MPA.
- Sprint 4:
  - Scores *30* additional points reliably by scoring *1* (or more) of 10 Kelp.
  - Scores *50* additional points reliably by scoring *8 (instead of just 3)* of more of 10 polyps from our side into MPA.
- Sprint 5:
  - Scores *90* additional points reliably by scoring *4* (instead of just 1) or more of 10 Kelp.
  - Scores *...*

Take-away (summary)

Consider *Simplified Scrum* (an *Agile* process)

- Make a Release Plan
- Repeatedly:
  - Do a Sprint
  - Evaluate
  - Re-Plan

KISS – don't overplan at this point

Items in the Release Plan are *Deliverables*, not Tasks. Most items should *score points*.

Do detailed planning (i.e., assign Tasks) *only* for the *current* Sprint

Be agile! What the team has accomplished so far informs what the team can accomplish in the rest of the time!

Always have working robots that score points