An Introduction to Scrum

Premek Brada, Lukas Holy
http://esecc.zcu.cz/
5.11.2013
OVERVIEW OF SCRUM
What to Expect from Scrum

- Quality?
- Speed?
- Cost?
- Agility?

Manifesto for Agile Software Development

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.
Scrum

Sprint goal
Return
Canceled
Coupons
Gift wrap
Product backlog

Sprint backlog

Sprint 2-4 weeks
24 hours

Potentially shippable product increment
Scrum framework

Roles
• Product owner
• ScrumMaster
• Team

Ceremonies
• Sprint planning
• Sprint review
• Sprint retrospective
• Daily scrum meeting

Artifacts
• Product backlog
• Sprint backlog
• Burndown charts
DETAILS ON PROCESS PARTS
Scrum framework

Roles
- Team
- Product owner
- ScrumMaster

Ceremonies
- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts
- Product backlog
- Sprint backlog
- Burndown charts
The team

• Cross-functional
  • Programmers, testers, UX designers, etc.
  • Ideally “no titles”
• Self-organizing

• Empowered + responsible
  • Ideally full-time members
• Strive for long-lived teams
The Team
Product owner

- **Responsible for the profitability of the product (ROI)**
- **Define** the features of the product
- **Decide** on release date and content
- **Prioritize** features according to market value
- **Adjust** features and priority every iteration, as needed
- **Accept** or reject work results
The ScrumMaster

- Responsible for enacting Scrum values and practices
- Represents management to the project
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles
- Shield the team from external interferences
Scrum framework

Roles
• Product owner
• ScrumMaster
• Team

Ceremonies
• Sprint planning
• Sprint review
• Sprint retrospective
• Daily scrum meeting

Artifacts
• Product backlog
• Sprint backlog
• Burndown charts
Sprints

- Sprint goal
- Sprint backlog
- 24 hours
- Sprint 2-4 weeks
- Potentially shippable product increment
Product Planning and Scrum
Sprint planning

• **PO explains** product backlog items + Team discusses (**collaboratively** => “Planning game”)

• Sprint backlog is created
  • Items from the product backlog which Team can commit to completing
  • **Tasks** are identified, each is estimated (1-16 hours)

• High-level design is considered

• **Definition of DONE** is needed

As a vacation planner, I want to see photos of the hotels.

Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)
No changes during a sprint

- Plan sprint durations around how long you can commit to keeping change out of the sprint
The Daily Scrum

- **Parameters**
  - Daily, 15-minutes
  - Stand-up

- **Not reports to SM**
  - *Synchronization* info for peers
  - *Commitment* in front of peers
  - Whole world is invited BUT only Team members (plus ScrumMaster, Product Owner) can talk
  - Not for problem solving

1. What did you do yesterday?
2. What will you do today?
3. Is anything in your way?
The Daily Scrum
The Sprint Review

• **Team presents** what it accomplished during the sprint (demo of new features)
• **PO accepts** – Definition of Done, Sprint Goal
• Informal
  • 2-hour prep time rule, No slides
• Whole team participates
• Invite the world
Sprint Review
Sprint retrospective

- Periodically take a look back at **what is and is not working**
- After every sprint

- Whole team
  - ScrumMaster
  - Team
  - + Product owner
  - + Possibly customers and others

Start doing

Stop doing

Continue doing

This is just one of many ways to do a sprint retrospective.
Sprint Retrospective

DIDN'T WORK

* working with documentation - Word as first traceability to dev progress,
  * sales, marketing etc don't know
  * how and why to use JIRA and Confluence
  * e.g. icons

* planning meetups for DEV redundant - flow of most clear, cycles of (re)work
  * handled regularly in the Flow (even on closed tasks, needed on box/web/at/model level)
  * closing tasks looks needless at first sight → real issue = traceability to requests
    *速率: each change request on requests (will be handled according to schedule on demand)
    * assign them to all affected requests + reopen those + some form of regression testing

* vertical - slice & modular development point to a large part due to front-end framework (React) + SDC + box incompatibilities

* people sync/planning issues ⇒ e.g. workload reviews
  * exacerbated by multiple projects running in parallel

* estimates not used and verified
  * upper management was sometimes off
  * UI was originally expected that deadline will be given and achieved

AO:

- clear next direction
Scrum framework

Roles
• Product owner
• ScrumMaster
• Team

Ceremonies
• Sprint planning
• Sprint review
• Sprint retrospective
• Daily scrum meeting

Artifacts
• Product backlog
• Sprint backlog
• Burndown charts
Product backlog

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the product owner
- Reprioritized at the start of each sprint
# A sample product backlog

<table>
<thead>
<tr>
<th>Backlog</th>
<th>Issue</th>
<th>Description</th>
<th>Priority</th>
<th>Status</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SWEE-12891</td>
<td>SDK - SIS Request - DAC API</td>
<td>2 Versions</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-9189</td>
<td>[Server / Web] Check for record changes before action (validation) - part 2</td>
<td>Rel-4.0</td>
<td></td>
<td>2w 1d</td>
</tr>
<tr>
<td></td>
<td>SWEE-12885</td>
<td>SDK - SIS Request - Business Service API</td>
<td>2 Versions</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-11243</td>
<td>[MWB - Web] - Use server function for counting of deposit price for some article</td>
<td>Rel-4.0</td>
<td></td>
<td>1w</td>
</tr>
<tr>
<td></td>
<td>SWEE-11052</td>
<td>Web: Result set restrictions in overview functions</td>
<td>Rel-4.0</td>
<td></td>
<td>2w</td>
</tr>
<tr>
<td></td>
<td>SWEE-12825</td>
<td>[MM] Screen Refresh - Internal Storage Amount - Inventory Master data export</td>
<td>Rel-4.0</td>
<td></td>
<td>2w 4d</td>
</tr>
<tr>
<td></td>
<td>SWEE-10606</td>
<td>Portal - Grid: Check save of positions into count list + small label improvement</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-12766</td>
<td>[Server - Report Items without stock] items must be loaded based on ITEM ID</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-10816</td>
<td>Server - configurable prefixes for price calculation</td>
<td>Rel-4.0</td>
<td></td>
<td>4d 4h</td>
</tr>
<tr>
<td></td>
<td>SWEE-2104</td>
<td>Support Main Currency in all Processes and Store Device Control</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-6085</td>
<td>Web: Configurable UI - layout 2/2 - Concept</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-11732</td>
<td>Support in Web - Product Web</td>
<td>Rel-4.0</td>
<td></td>
<td>1w</td>
</tr>
<tr>
<td></td>
<td>SWEE-7228</td>
<td>International Master Data on one system - Control Points</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-5539</td>
<td>[Portal] Validation Property (NullQuantityAllowed) hangs up save. 0 is defined as valid</td>
<td>Rel-4.0</td>
<td></td>
<td>1w</td>
</tr>
<tr>
<td></td>
<td>SWEE-12760</td>
<td>[MWB] Support Sybase Database (ASE)</td>
<td>Rel-4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWEE-10744</td>
<td>[SIS] Properties in Structure - Item Standard view should be displayed in standard way</td>
<td>Rel-4.0</td>
<td></td>
<td>1d</td>
</tr>
<tr>
<td></td>
<td>SWEE-10746</td>
<td>[SIS] Properties in Structure - Item Standard view should be displayed in standard way</td>
<td>Rel-4.0</td>
<td></td>
<td>1d 4h</td>
</tr>
<tr>
<td></td>
<td>SWEE-10606</td>
<td>SDK - API JavaDoc description improvement</td>
<td>Rel-4.0</td>
<td></td>
<td>7w 1d</td>
</tr>
<tr>
<td></td>
<td>SWEE-11772</td>
<td>[Server/Web/Mobile] Support of Storage Locations - new function Stock Transfer</td>
<td>Rel-4.0</td>
<td></td>
<td>4w</td>
</tr>
<tr>
<td></td>
<td>SWEE-10778</td>
<td>Inventory - Additive registration by scan in quantity field</td>
<td>Rel-4.0</td>
<td></td>
<td>1w</td>
</tr>
<tr>
<td></td>
<td>SWEE-6192</td>
<td>GUI Portal Tables improvement - Details Viewer (Item + supplier)</td>
<td>Rel-4.0</td>
<td></td>
<td>2w</td>
</tr>
</tbody>
</table>
The sprint goal

- A short statement of what the work will be focused on during the sprint

**Database Application**
Make the application run on SQL Server in addition to Oracle.

**Life Sciences**
Support features necessary for population genetics studies.

**Financial services**
Support more technical indicators than company ABC with real-time, streaming data.
The Sprint Backlog

- **Plan of the work** to be finished during sprint (result of sprint planning)
Managing the sprint backlog (1)

• Individuals sign up for work of their own choosing

• Estimated work remaining is updated daily
Managing the sprint backlog (2)

• Any team member can add, delete or change the sprint backlog
  • Work for the sprint emerges

• If work is unclear
  • Define a sprint backlog item with a larger amount of time and break it down later
  • Update work remaining as more becomes known
Tracking progress: Burndown

• Simple visual information on work remaining vs time available
• Information radiator
A sprint burndown chart

- Work remaining vs available time – simple visual presentation ("information radiator")
SUMMARY
**Scrum Development Process**

1. **Product Backlog**
   - List of requirements and issues
   - Owned by Product Owner
   - Anybody can add items
   - Only Product Owner prioritizes

2. **Sprint Planning**
   - Declare Sprint Goal
   - Select highest priority items from Product Backlog that supports the Sprint Goal
   - Team turns selected items into Sprint Backlog and estimates them

3. **Sprint Backlog**
   - List of tasks
   - Owned by Scrum Team
   - Only Scrum Team modifies it

4. **Backlog Estimation**
   - Product Owner describes new items
   - All discuss
   - Team estimates
   - Product Owner updates Product Backlog

5. **Daily Scrum**
   - Same time every day
   - What did you do yesterday?
   - What will you do today?
   - What’s in your way?
   - Team updates Sprint Backlog
   - Scrum Master updates Impediment List & Sprint Burndown Chart
   - Sprint Burndown Chart: How much work left until done

6. **Sprint Review**
   - Sprint Demonstration
   - Team demonstrates Increment
   - All discuss
   - Announce next Sprint Planning Meeting
   - Sprint Retrospect
   - Team discusses the sprint to answer the following:
     - What should we keep doing?
     - What should we stop doing?
     - What should we start doing?

7. **Increment**
   - Version of the product
   - Potentially shipable (tested, documented etc.)

**Roles**

- **Product Owner**
  - Sets priorities and manages the product backlog

- **Scrum Master**
  - Manages the process and removes impediments

- **Scrum Team**
  - Develops the product and self-organizes

FURTHER TOPICS
Distributed Teams

• Not ideal
• But possible

• Backlog in online tools
• Meetings via voice/video conference
• Communicate a lot (include “small talk”)
• Document more
Scalability

- Typical individual team is 7 ± 2 people
  - Scalability comes from teams of teams
- Factors in scaling
  - Type of application
  - Team size
  - Team dispersion
  - Project duration
- Scrum has been used on multiple 500+ person projects
Scaling through the Scrum of scrums
Names and Books

- Jeff Sutherland
- Ken Schwaber
- Mike Cohn
- Craig Larman
- Henrik Kniberg
Thank You

Presentation by:
Mike Cohn <mike@mountaingoatsoftware.com>
Premek Brada <brada@kiv.zcu.cz>