Agile & User Centred Development

Joakim Lööv
Lead Usability Designer

Agile dev.: An iterative and incremental (evolutionary) approach to software development, which is performed in a highly collaborative manner, by self-organizing teams within an effective governance framework, with "just enough" ceremony, that produces high quality software, in a cost effective and timely manner, which meets the changing needs of its stakeholders.

Contents

1. Generic foundation
2. Traditional processes
3. The agile movement
4. Scrum walkthrough
5. UCSD & Scrum
Matrix organization

<table>
<thead>
<tr>
<th></th>
<th>Line mng 1</th>
<th>Line mng 2</th>
<th>Line mng 3</th>
<th>Line mng 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project mng 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project mng 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project mng 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Förut - RUP**

![Diagram of Phases and Iterations]
Lean

- Waste: The ideal is to find out what a customer wants, and then make or develop exactly what they want, virtually immediately. Whatever gets in the way of satisfying a customer need is waste.
- Amplify learning: The best approach to improving a software development environment is to amplify learning.
- Design as late as possible: Delaying decisions is valuable because better decisions can be made when they are based on fact, not speculation.
- Build integrity in: Software with integrity has a coherent architecture, scores high on usability and fitness for purpose, and is maintainable, adaptable, and extensible.
- See the whole: When individuals or organizations are measured on their specialized contribution rather than overall performance, suboptimization is likely to result.

Agile manifest

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan
Process walkthrough

Idea / Vision

Product Backlog

Sprint Planning
4 x 4 h

Sprint Backlog
30 d

28 x 24 h

Sprint Retrospective
3 h

Sprint Review
1 x 4 h

Daily Scrum
15 min

Burndown chart

Common ground – Areas of conflict

Behaviors

Attitudes

Values
Common ground – Areas of conflict

<table>
<thead>
<tr>
<th>Values</th>
<th>Agile</th>
<th>User centred</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highest priority is to satisfy the customer</td>
<td>User, usage, and organizational value is what matters</td>
<td></td>
</tr>
<tr>
<td>Customer collaboration</td>
<td>Active user involvement</td>
<td></td>
</tr>
<tr>
<td>Adaptive to change</td>
<td>Iterative design</td>
<td></td>
</tr>
</tbody>
</table>

Common ground – Areas of conflict

<table>
<thead>
<tr>
<th>Attitudes</th>
<th>Agile</th>
<th>User centred</th>
</tr>
</thead>
<tbody>
<tr>
<td>Focus on product (rather than project)</td>
<td>Focus on user (rather than technology)</td>
<td></td>
</tr>
<tr>
<td>Broad skills</td>
<td>Specialists</td>
<td></td>
</tr>
<tr>
<td>Fast, and good enough</td>
<td>Accurate</td>
<td></td>
</tr>
<tr>
<td>Knowledge and learning</td>
<td>Understanding</td>
<td></td>
</tr>
<tr>
<td>Cross-functional</td>
<td>Interdisciplinary</td>
<td></td>
</tr>
<tr>
<td>System development</td>
<td>Organization development</td>
<td></td>
</tr>
<tr>
<td>Simplicity in solution</td>
<td>Simplicity in usage</td>
<td></td>
</tr>
</tbody>
</table>
Common ground – Areas of conflict

<table>
<thead>
<tr>
<th>Behaviours</th>
<th>Agile</th>
<th>User centred</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Incremental system design</td>
<td>Evolutionary usability/interaction design, prior to implementation</td>
</tr>
<tr>
<td></td>
<td>Sprint, scrum, sprint planning, demo, retrospect, refactoring, …</td>
<td>User/task analysis, prototyping, usability evaluation, …</td>
</tr>
<tr>
<td></td>
<td>User stories, product backlog, sprint backlog, burn down, …</td>
<td>Personas, scenarios, wireframes, storyboards, evaluation highlights, …</td>
</tr>
</tbody>
</table>

Key success factors

- Product owner with UCD skills
- Create a conceptual design
- Iterate within sprints
Key success factors

Product owner
- UCD skills
- Req. management
- ...

Team
- Programing
- UCD skills
- Test
- ...

Stakeholders
- Users
- ...

ScrumMaster

Process walkthrough

Idea / Vision

Product Backlog

Sprint Planning
4 + 4 h

Sprint Review
1 + 4 h

Sprint Retrospective
3 h

Sprint Backlog

28 x
24 h

Daily Scrum
15 min

Burndown chart

30 d

1 + 4 h
Questions

Extras

- researching work to be done 2 timeboxes from now (T + 2)
- validating design prototypes to be built 1 timebox from now (T + 1)
- being available to collaborate with development to support work done in the current development timebox (T)
- working with customers to validate working software built in the previous timebox (T – 1)
Extras

Planning

System Architecture/High Level Design

Develop
Adjust
Sprint
Wrap
Review
Sprint Review

Closure