Agile Software Development

L05 – Scrum

Ezequiel Scott
ezequiel.scott@ut.ee
Agenda

• Recap
• ASD with Scrum
• Scaling Scrum
Recap

• Software development processes, agile terminology
• Current state of Agile worldwide
• Test-driven Development (TDD)
• Handling requirements in ASD
  • Writing user stories
• Organizing User Stories
• Refactoring and Code smells
Agile Software Development

Subway map to agile practices
https://www.agilealliance.org/agile101/subway-map-to-agile-practices/
Agenda

• Recap
• **ASD with Scrum**
• Scaling Scrum
Agile Software Development with Scrum

• An article published in Harvard Business Review (1986) is the inspiration for the Scrum framework.

• Ken Schwaber and Jeff Sutherland co-present Scrum at the OOPSLA Conference

• https://www.scrumguides.org/scrum-guide.html
What is Scrum?

• Scrum is a **process framework** used to manage **product development** and other knowledge work

• Scrum is **empirical** → it provides a means for teams to:
  • set a hypothesis of how they think something works,
  • try it out,
  • reflect on the experience, and
  • make the appropriate adjustments

• Scrum is structured in a way that allows teams to **incorporate practices from other frameworks**
Subway map to agile practices
https://www.agilealliance.org/agile101/subway-map-to-agile-practices/
When is Scrum applicable?

- **Cross functional team** is working in a product development setting

- There is a **non trivial amount of work** that lends itself to being split into more than one 2–4 week iteration

- The essence of Scrum is a **small team** of people (3 to 9 team members)

- The individual team is **highly flexible and adaptive**

- Scrum has been used to develop software, hardware, embedded software, networks of interacting function, autonomous vehicles, schools, government, marketing, managing the operation of organizations and **almost everything** we use in our daily lives, as individuals and societies.

https://www.scrumguides.org/scrum-guide.html
Scrum roles

- **Product Owner**: Owns “what” is desired and “why” it’s desired
- **ScrumMaster**: Keeper of Scrum Process, facilitator
- **Scrum Delivery Team**: Owns “how” and “how quickly” work is delivered

Direct communication encouraged.
Scrum phases

• **Pregame:**
  
  • **Planning**: Definition of a **new release** based on currently known backlog, along with an **estimate** of its schedule and cost  
  New system → conceptualization and analysis  
  Existing system → limited analysis  
  
  • **Architecture**: Design how the backlog items will be implemented. This phase includes **system architecture** modification and **high-level design**

• **Game:**
  
  • **Development Sprints**: Development of **new release functionality**, with **constant respect** to the variables of time, requirements, quality, cost, and competition  
  
  • Interaction with these variables defines the end of this phase

• **Postgame:**
  
  • **Closure**: Preparation for release, including final documentation, pre-release staged testing, and release
The Scrum framework

- Building of Product Backlog
- Configuration of development environment
- Distribution of workstations

Feedback
Self-reflection
Celebration
Improvements

Closing the Sprint

Organizing and Preparing User Stories

Controlling and Monitoring Sprint Work
- Product Increment
- Product Integration

Daily Scrum 24 HS

Planning the Sprint Backlog
- Sprint Backlog
- Planning Poker

Delivering the Product
**User Stories**

*Role*

As a user, I can cancel a reservation.

*Must take less than 1 second*

<table>
<thead>
<tr>
<th>Feature</th>
<th>Acceptance conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>□ Verify that a premium member can cancel the same day without a fee.</td>
<td></td>
</tr>
<tr>
<td>□ Verify that a non-premium member is charged 10% for a same-day cancellation.</td>
<td></td>
</tr>
<tr>
<td>□ Verify that an email confirmation is sent.</td>
<td></td>
</tr>
</tbody>
</table>

Details behind the user story come out during conversations with the Product Owner.
Backlogs

Product Backlog

S
User login

S
SSL enable

M
Reset lost password

S
Account lockout after three attempts

M
LDAP integration

L
Register a new login

M
Edit registration

XL
Admin reporting

XL
User-managed wishlists

Selected Product Increment

Sprint Backlog

User login

SSL enable

Reset lost password

Account lockout after three attempts

MTAT.03.295 | Agile Software Development | © Ezequiel Scott 2020
The Scrum framework
Agile Estimating and Planning

Size → Calculation → Duration

300 kilograms

Velocity = 20

300 ÷ 20 = 15 iterations
Story points

• How long a User Story will take to develop (effort)

• Influenced by:
  • Complexity
  • Risk
  • Uncertainty
How to determine story points?

• One popular alternative is Planning Poker

• Scrum poker or planning poker is a consensus based, gamified technique to estimate the complexity and effort of a software feature.

• All the team members discuss using cards

• The dialogue improves accuracy (Hoest and Wohlin 1998)

• Different scales can be used
  • Fibonacci (e.g. 1, 2, 3, 5, 8, 13, 20, ... )
  • T-shirt sizes (XS, S, M, L, XL)
Planning Poker

Online version: https://scrum poker.online/
The allocation of User Stories to iterations is done according to **story points**, **velocity** and **sprint goals**.
Daily Scrum

(using Kanban boards → ScrumBan?)
Daily Scrum

• Ensure that all members *self-assign* at least one task

• Produce high quality code

• Document as necessary

• Keep in mind the **Sprint Goal**

• The team works to achieve a **product increment**
Monitoring Progress Toward Goals

• At any point in time, the total work remaining to reach a goal can be summed.

• Various projective practices upon trending have been used to forecast progress, like burn-downs, burn-ups, or cumulative flows. These have proven useful. However, these do not replace the importance of empiricism.
The Burndown chart

Can be used at any point in time:
- To show daily progress
- To show project progress

Figure 11.4 Burndown chart for the project in Table 11.2.

Figure 11.5 A daily burndown chart.

Burndown Chart (cont.)

![Burndown Chart](image)

- **Behind Schedule**
- **Ahead of Schedule**

- **TODO**
- **DONE**

**Sprints**

**Story Points**
Daily meetings

- On **each day of a sprint**, the team holds a daily scrum meeting
- They are strictly **time-boxed** to 15 minutes
- All team members are required to attend scrum meetings

Daily meeting

- What did you do yesterday?
- What will you do today?
- Are there any impediments in your way?

https://www.agilealliance.org/daily-scrum-is-it-a-waste-of-time/
Daily meetings

- On each day of a sprint, the team holds a daily scrum meeting.
- They are strictly **time-boxed** to 15 minutes.
- All team members are required to attend scrum meetings.
- **Some improvements →**

(Downey and Sutherland, 2013)

The Scrum framework

- Product Increment
- Product Integration
- Feedback
- Self-reflection
- Celebration
- Improvements

Closing the Sprint

Organizing and Preparing User Stories

Planning the Sprint Backlog

Controlling and Monitoring Sprint Work

Daily Scrum 24 HS

- Sprint Backlog
- Planning Poker

Product Owner

As an User
I want

As an User
I want

As an User
I want

As an User
I want
Sprint Review

• Prepare a demo to validate the customer’s User Stories
• Ensure that the product works under acceptable conditions
• Receive and leverage the feedback on the increment
Sprint Retrospective

• Each team holds a private meeting
• Reflect on what happened along the sprint
• Each team member is asked to identify specific things that the team should:
  ✓ Start doing (AN IMPROVEMENT)
  ✓ Stop doing (THE BAD)
  ✓ Continue doing (THE GOOD)
• The result is at least one action item included on the following Sprint’s Sprint Backlog
Agenda

• Recap
• ASD with Scrum
• Scaling Scrum
Scaling Scrum

• Why scale?
  • We need to go faster?
  • Do we have a lot of people?
• We scale **to get more done in a given timeframe**
• It doesn’t mean to add more people necessarily

• Different approaches:
  • Scrum of Scrums
  • NEXUS
  • …
Scrum of Scrums
What’s next?

• Next lecture (next week):
  • Exam Warm up
  • Agile at Scale

• Practice (this week):
  • Intro to Phoenix

• HW1 deadline soon!
  • Submission deadline: **Friday 09.10.20 at 23:59**
  • Submit via Moodle