

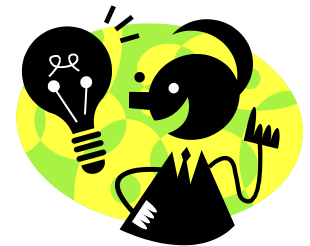
CS-C2130 / CS-C2140 / CS-E4910

Software Project 1 / 2 / 3

Lecture 2: Scrum Basics
Casper Lassenius & Jari Vanhanen

Agenda

- Scrum basics and applying Scrum on this course
- Additional requirements for the course projects
- Next Steps on the course





Aalto University
School of Science
and Technology

Scrum Basics

13.9.2023

Goals of This Lecture

- Teach you
 - The basics of the Scrum process
 - Roles
 - Process steps
 - Terminology
- After this lecture
 - You know the basics of Scrum and how it can/should be applied in the course project
 - You are able to participate in the Scrum Simulation in the developer role
- This lecture is based on
 - Scrum Primer (and Scrum Guide 2020)
 - [CS-C2130 Project Manual](#)



Scrum

- Iterative and incremental agile software development framework for **managing** product development

Does not cover design, implementation, or concrete testing practices

- Process framework
 - not a process, technique, or definitive method
 - every team must decide the specific tactics for using Scrum

Introduction to Scrum (7 min)

- <https://youtu.be/9TycLR0TqFA>

Why Process?

- A process defines how an organization, in your case, a team works together to achieve its goals
- Why do you think having an understood/agreed upon process is or is not beneficial?

Scrum Roles

Product Owner (PO)

- Responsible for maximizing return on investment, thus has the final authority
 - Identifies product features
 - Prioritizes the features
 - Interacts regularly with the developers
 - e.g. reviews the Sprint results
 - May delegate some work to the developers, but remains accountable
 - One person
 - Product Owner \approx Product Manager \approx Customer
-



The Developers

- Develop the product and provide ideas to the Product Owner about how to make the product great
- 7 ± 2 people
- Cross-functional team
 - together they have all expertise necessary to deliver a potentially shippable product each sprint
- Are self-managing: high degree of autonomy and accountability
- Every developer is just a developer, no other roles



Scrum Master

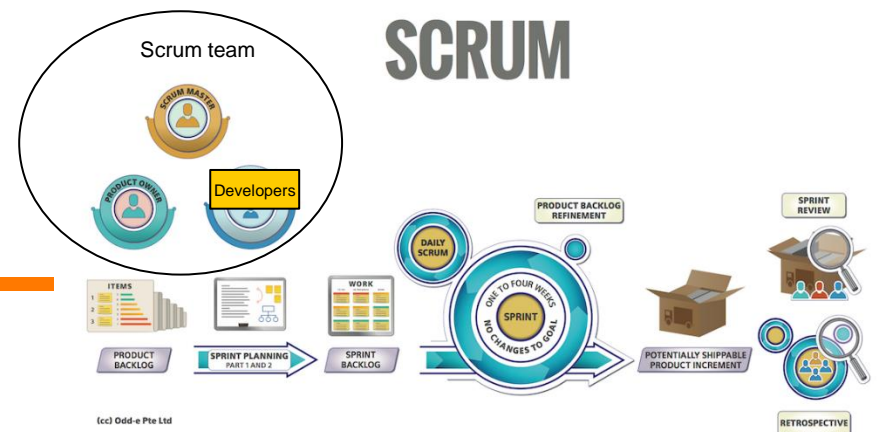


- Helps the Scrum team learn and apply Scrum to achieve business value
- Is NOT the manager of the team members, NOR a project manager OR team lead
- Is a coach and teacher, especially Scrum principles and practices
- Serves the team, e.g. helps to remove impediments, protects from outside interference, helps to adopt good work practices

Scrum Process

Sprint

- Time-boxed development cycles of 1-4 weeks
- Never extended: ends exactly when planned, contents give flexibility
- The output of every sprint is: “Potentially Shippable Product Increment”, which means that item chosen for that sprint are “Done” (according to the Definition of Done)
 - System is integrated
 - Fully tested
 - End-user documented
 - Potentially shippable



Sprints (CS-C2130)

- At least six Sprints
 - $225\text{h} / 6 = 37.5\text{h}$ (per student)
- Plan in the beginning of the project
 - start and end dates of all sprints
 - effort allocation per person per Sprint
- First Sprint (“Sprint 0”) and last Sprint differ from the normal Sprints
 - contain some tasks defined by the course

Product Backlog

- Is a prioritized list of customer-centric features
- “Everything that could be done by the Scrum team ever in order of priority”
- Includes “items”, e.g. new customer features, major engineering improvement goals, research work, (known defects)
 - User stories, epics
- Includes effort estimates
 - e.g. as story points
- Is detailed appropriately
- Is regularly refined (“grooming”) = splitting, estimating, re-estimating items



User Stories and Epics [1]

- User story
 - Basic format: “As a [type of user] I [want/can/am able to/need to/etc.] so that [some reason].”
 - Can be in other formats, as long as the above aspects are covered
 - Can be implemented in **one** Sprint
 - Works well for functional requirements, less well for quality attributes

CS-C2130: Describe SW features as user stories

- Epic
 - Basically a “big user story”, i.e. cannot be implemented in a single sprint
 - Usually broad in scope, short on details, and will commonly need to be split into multiple, smaller stories before the team can work on them

Product Vision (CS-C2130)

1. Why?

- explain why the product is being built (the business view)

2. What?

- product goal, i.e. the desired state of the product in the end of the course project
- include also critical quality attributes that are difficult to include in the Definition of Done

3. For Whom?

- characterize the end users

Created based on the project proposal and further discussions with the PO

Sprint Planning: Topics 1 & 2

- Participants: Product Owner, Developers, Scrum Master
- Understand
 - WHY this Sprint is valuable
 - Sprint goal
 - WHAT can be done
 - items from the Product Backlog
- Discussion
 - PO explains
 - Developers ask questions
 - Joint decision on what can realistically be included



**“WHY &
WHAT?”**



Sprint Planning: Topic 3

- Participants: Developers, Scrum Master (PO reachable for questions)
- Focus on HOW to implement the selected items
- May contain:
 - Overall design
 - Splitting product backlog items into tasks – building sprint backlog!
 - Estimating items/tasks
 - Renegotiating scope














“HOW?”



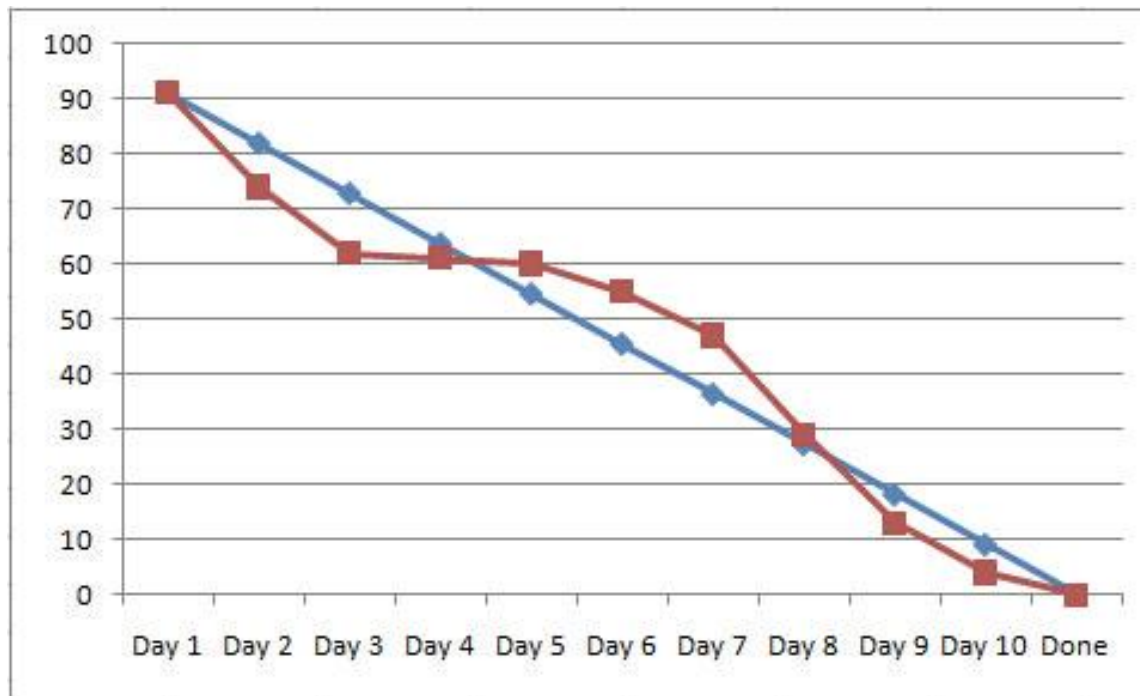
Sprint Backlog

- Sprint goal
- Sprint backlog items
 - Some items from the product backlog, and the necessary tasks
 - Attributes of the tasks
 - name/description
 - effort estimate as hours or story points
 - at least if the tasks are large (>1 workday)
- Product and Sprint backlogs should be in a (real) backlog management tool
 - Jira, Trello, ...
 - a well-organized Miro board is allowed too

Scrum Board

Story	Not Started	In Progress	Done
			
			
			
			
			

Sprint burn-down



Sprint burn-down

Daily Scrum Meeting

CS-C2130

- At least once per week

- Participants: Developers, Scrum Master (Product Owner optional)
- Update and coordination – not a status reporting to anybody else
- Max 15 min
- Each member report to the other team members, e.g.:
 - What have I accomplished since the last meeting?
 - What will I do before the next meeting?
 - What obstacles are in the way?
- If discussion needed: follow-up meetings agreed and held afterwards

15 min
3 Questions



Definition of Done (DoD)

- Everyone must understand what “done” means
 - e.g. for tasks, user stories, sprints
- Sometimes people say it is “done-done” to mean it meets the criteria for DoD
- Scrum Team must define their own DoD (and follow it!)
 - ... and improve it when needed
- Often at several levels
 - Task, User story, Sprint
- Typically things like
 - Code is implemented, commented, integrated
 - Automated unit tests have been written, and pass when executed

- unit testing
- functional system testing
- coding standard

More Quality Assurance (CS-C2130)

- Quality attributes (non-functional requirements)
 - e.g. usability, security, performance, compatibility
 - identify the most relevant ones (at least one, but not too many)
 - consider them appropriately in DoD / Product vision / technical design
- Peer testing
 - by some other team on the course
 - outsiders can often spot different bugs and improvement ideas than the developers themselves
 - you must plan how to best utilize the other team (when & what)
 - at least 8 man-hours per team using Session-based exploratory testing

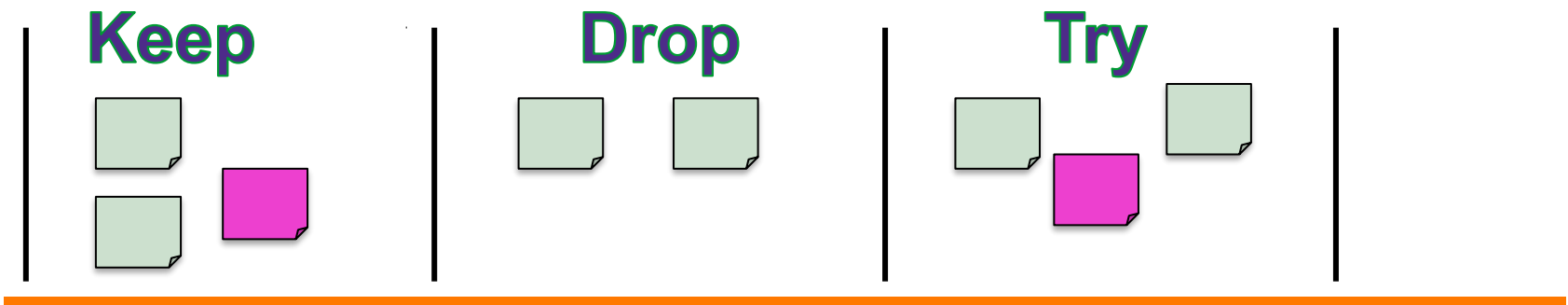
Sprint Review

- Participants: Developers, PO, Scrum Master, other stakeholders invited by the PO
- Inspection of the increment and adaption of the product backlog, if needed
 - Hands-on inspection of the real software running live
 - In-depth conversation
 - What was accomplished in the Sprint?
 - What has changed in the environment?
 - What to do next?



Sprint Retrospective

- Inspecting how the last Sprint went with regards to individuals, interactions, processes, tools, and DoD
- Participants: Developers, Scrum Master, PO (optional)
 - Developers discuss **what's working** and **what's not working** and agree on **changes to try**
 - Usually, the Scrum Master facilitates
 - Different techniques, try different ones!



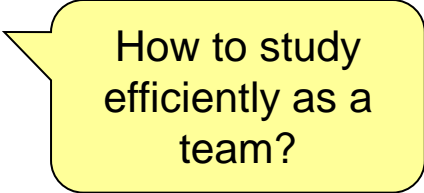
Being Efficient: Doing a Sprint Change (CS-C2130)

- In one sitting
 - Sprint Review
 - Sprint Retrospective
 - Sprint Planning

- Requires access to Product Owner

Sprint 0 (CS-C2130)

- Sprint goal
 - “Set up the project so that everything is ready for starting sw development work from the first day of the following Sprint.”
- Main tasks
 - product vision and initial Product Backlog
 - prototyping, selecting and **studying technologies**
 - deciding work methods and tools, e.g.
 - communication channels, team work sessions
 - practicalities of the Scrum events
 - backlog management, time tracking, version control
- Results presented to the PO and **to the Coach**



How to study efficiently as a team?

Last Sprint (CS-C2130)

- Focuses on finalizing the product for the final delivery to the PO
- Some tasks
 - bug fixing and finalization (no more new features)
 - acceptance testing by the Client
 - handover to the Client (both the system and any necessary knowledge)
 - preparing an excellent software demo and a project poster

What If Something in Scrum Does Not Work for Us? (CS-C2130)

- Try it (for real) first
- If you really need to change it
 - Make a motivated proposal to your coach
 - Try the changed version

Want to Know More about Scrum?

- Google
 - Scrum Guide
 - Scrum Primer

Read the CS-C2130 Project Manual

1. It summarizes briefly the requirements set in the Scrum Guide.
2. It describes the modified/additional requirements set by the course.

In order to understand why and how to follow the Project Manual, you must read [Scrum Guide](#) / [Scrum Primer](#)

Tips for Working as a Remote Team

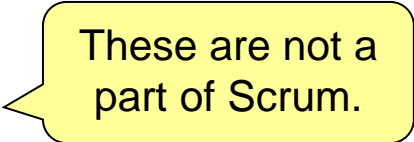
- Plan joint working sessions – just like you would if you were collocated
- Use e.g. Zoom and keep the session open for the whole day / length of your coworking time
- Always use video when you are communicating in a meeting
- Have (e.g. Slack) channel for the team, always on

End of Scrum Basics

Additional requirements for the course projects set by the course

Additional requirements for the course projects (CS-C2130)

- Process Overview Document
- Technical Overview Document
- Time Tracking
- Project Reviews



These are not a part of Scrum.

Document – Process Overview (CS-C2130)

- Document briefly the *currently* used work practices and tools **so that all stakeholders can understand how the team works**
- Minimum content
 - project schedule and effort distribution
 - Sprint dates and allocated effort per person
 - other main events (Project Reviews, team work sessions)
 - recurring events of the Sprints (how and when)
 - Sprint Planning, Daily Scrums, team work sessions, Sprint Review, Sprint Retros
 - other main practices and tools
 - backlogs, time tracking, communication etc.
 - version control, testing etc.

See the template

Producing a document is not the main purpose.
The most important thing is to adopt good work practices that can be realistically used.

Document – Technical Overview (CS-C2130)

- Very project specific
- General goals
 - **Helping the Scrum Team during the project**
 - e.g., in communicating about the design or in dividing responsibilities
 - **Meeting the Client's needs after the project**
 - e.g., helping some new developers fix bugs or develop new features
- Minimum content
 - Document briefly the **most important architectural design decisions**
 - Document **one or more relevant views** of your architecture design
 - Google e.g. 4+1 architectural view model.

Time Tracking (CS-C2130)

- Total effort spent per student per each Sprint
 - includes everything related to the project
 - must be visible also to the Coach
 - must be updated at least weekly
 - impossible to remember what you did last week
 - if someone falls behind or works extra in one Sprint, update the remaining hours in coming Sprints accordingly
- Some backlog management tools support time tracking
- A simple spreadsheet can work too
 - if you are not interested in task level tracking

	M1	M2	M3	M4	M5	M6	M7	SUM
Sprint 0	40	35	35	20	20	35	35	Xxx
Sprint 1	20	40	40	55	55	40	40	Xxx
...
Sprint N	20	40	40	25	25	40	40	xx
Total	100	225	225	225	225	225	225	xxxx

(realized hours and remaining hours)

See the course's
Google Sheet example

Project Review (CS-C2130)

- November, February and April
- Participants
 - student team, PO, coach, teacher, and possibly some other people
- Team presents data on the project status
 - status of Sprint Goals and selected Product Backlog items
 - main findings from Sprint retros
 - software quality
 - effort usage per person
- Team presents the results (mainly a software demo)
 - plan and rehearse
- After each project review, PO and coach evaluate the project

See the Progress /
Final report template (slides)

Project Review: Summary of the required materials

- Product vision (Template available)
- Product Backlog
- Sprint Backlog of the current Sprint
- Process overview (Template)
- Definition of Done
- Technical overview
- Progress report / Final report slides (Template)
- Test session charter(s) for peer testing (Template)

- Learning diaries (submitted individually by each student to MyCourses)

Send a link to the materials to the teacher, coach and PO

- 24 hours before each project review
- the link will be published in MyCourses

Next Steps

- **14.-15.9. 8:00 - 16** CSM Training for SMs only
 - **Mo 18.9. 13:00** The remaining students assigned to the teams
 - If you are not in any team, **fill the “Preferred Teams”-column by Su 17.9.**
 - Scrum Masters should fill the *Team’s favorite topics* column before that
 - **We 20.9. 16:15 - 18** session for SMs only
 - Scrum Master's role in the course project and in the Scrum Simulation
 - **ASAP**, register to a Scrum Simulation session
 - 5 teams are needed already to the first session on Mo 25.9. 16-20.

 - **6.-9.10. Send Team "CV" to 3-5 Clients**
 - Choose also one topic that you will “certainly” get
 - Keep the *Team’s favorite topics* column on the Team Info sheet up-to-date
 - **We 11.10. 16:15 - ~18:15** Meetings with the Clients
 - Reserve 10-minute meetings on the GoogleSheet
-