

## Det agile manifesto og SCRUM

*Vi finner bedre måter å utvikle programvare på ved å gjøre det selv og ved å hjelpe andre med det. Gjennom dette arbeidet har vi lært oss og versette dette.*

Verdiene i det agile manifesto brukes i Scrum:

- Personer og samspill, fremfor prosesser og verktøy
- Programvare som virker, fremfor omfattende dokumentasjon
- Kundesamarbeid framfor kontraktforhandlinger
- Reagere på endringer, fremfor å følge en plan

.....  
**What is scrum?**

- an agile process
- team-based
- a process that controls chaos
- improve communication
- detect and cause removal of anything that gets in the way of developing products scaleable
- a way for everyone to feel good

### Scrum - three pillars

- Transparency - feedback loop
- Inspection - kontroll opp mot sprintmål
- Adaption - justeringer underveis

Three ancilliary roles: stakeholder, managers  
Product backlog - sprint backlog  
- sprint recursion with daily scrum meeting - potentially shippable product

### Three primary roles of SCRUM

- Product owner: voice of the customer, managing the product backlog, responsible for profits, adjust priority, decide release date
- Scrum master: ensure that the team is productive and functional, ensure the process is followed, remove barriers
- Scrum team: around 7 members, cross functional, selects the spring goal, has the right to do anything, demos work results to product owner

### Three artifacts

- Product backlog: the master list of all functionality, dynamic, (user stories)
- Sprint backlog: list of tasks that the scrum team is committing to complete in the current sprint, based on product backlog
- Burndown chart: a chart showing remaining work in the sprint backlog, updated every day during the scrum meeting, it is successful if the sprint backlog is zero at the end of the sprint

### **User stories**

Estimated in points, how many points a team can handle in one sprint. As a role i want to achieve something

### **Product backlog**

- functional requirements: what the system is supposed to do
- non-functional requirements: how the system is supposed to be (written as user story in the *product backlog*)
- features
- technical tasks

### **The systems quality attributes (user stories)**

- accuracy
- reliability
- robustness

The time between serious errors, availability, accuracy in calculations, behaviours when errors occur

- performance
- efficiency

Number of simultaneous users, resource use, response duration, expenses

- maintenance

Troubleshooting, changeability, testability, portability

### **Four ceremonies**

- Sprint planning meeting: everyone present, definition of done, sprint goal
- Daily scrum meeting: 15 mins, what have you done, what have you planned, blocks?

- Sprint review: everyone present, demo, inspect and adapt, the success will be assessed during this meeting against the sprint goal
- Sprint retrospective: evaluation of the sprint. if we could redo, we would do these things the same way, and what would we do differently. Concrete ideas.