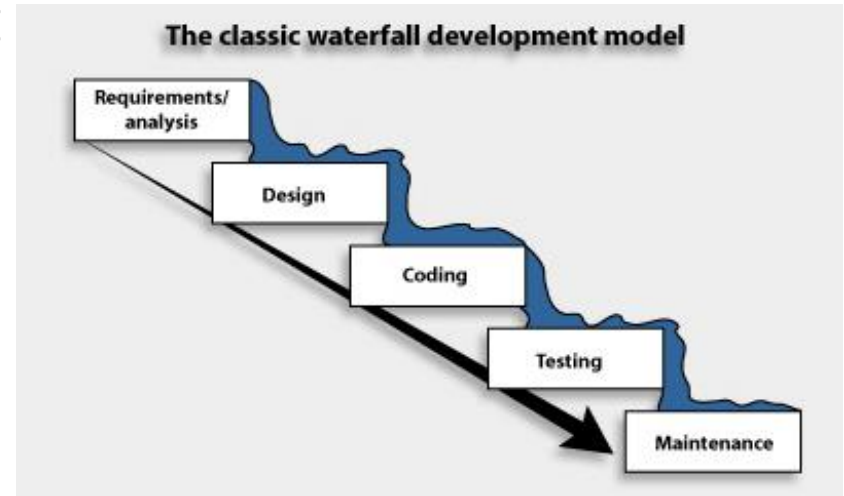


Arbejdsmetoder ifm. udvikling af IT-systemer

Vandfaldsmodellen

- Et projekt opdeles i faser:
 - Kravspecifikation/Analyse
 - Design
 - Kodning
 - Test
 - Igangsætning



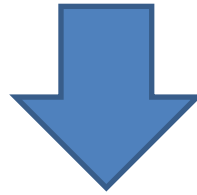
- En fase startes når den forrige er færdig
- Stepwise refinement (fra 0% implementeret til 100% implementeret)
- Bruges inden for mange industrier

Opgave

- Find et eksempel på en branche hvor vandfaldsmodellen helt naturligt bruges (ikke IT branchen)
- Hvad svarer faserne til i denne branche?
 - Kravspecifikation/Analyse
 - Design
 - Kodning
 - Test
 - Igangsætning

Fase 1: Kravspecifikation/Analyse

- Analyse af hvilke behov det kommende IT-system skal dække



- Liste af krav = kravspecifikation
- Krav: *Hvad* skal udvikles og *ikke hvordan*
- Hvert krav i en kravspecifikation skal:
 - *være præcist formuleret*, så der ikke er uklarhed om, hvad kravet går ud på
 - *være testbart*, så det kan undersøges, om et slutprodukt faktisk opfylder kravet
 - *give mening* i forhold til produktet

Eksempel - sundhedsapp

- Krav:

App'en skal på baggrund af indtastet vægt og højde kunne udregne BMI og give en tilhørende klassifikation.

Kan det gøres mere præcist?

Husk!

Et krav skal:

- være præcist formuleret (så der ikke er uklarhed om, hvad kravet går ud på)
- være testbart (så det kan undersøges, om et slutprodukt faktisk opfylder kravet)
- give mening i forhold til produktet

Opgave

- Find et eksempel på et it-system, hvor det har taget så lang tid at udvikle det, at kravspecifikationen har ændret sig undervejs.
- Find et eksempel på et it-system der, da det kom, var så nyskabende, at der kan have været problemer med at lave en klar kravspecifikation.

Opgave

- Lav en krav-specifikation til en webshop (opskriv 3-5 krav)

Husk!

Et krav skal:

- **være præcist formuleret** (så der ikke er uklarhed om, hvad kravet går ud på)
- **være testbart** (så det kan undersøges, om et slutprodukt faktisk opfylder kravet)
- **give mening i forhold til produktet**

Fase 2: Design

- Interaktionsdesign:
 - Gestaltlovene
 - Don Normans designprincipper
 - Skitser
 - Prototyper (LOFI, HIFI)
- Datamodellering:
 - Databasedesign: ER-diagrammer, normalisering

Fase 3: Kodning

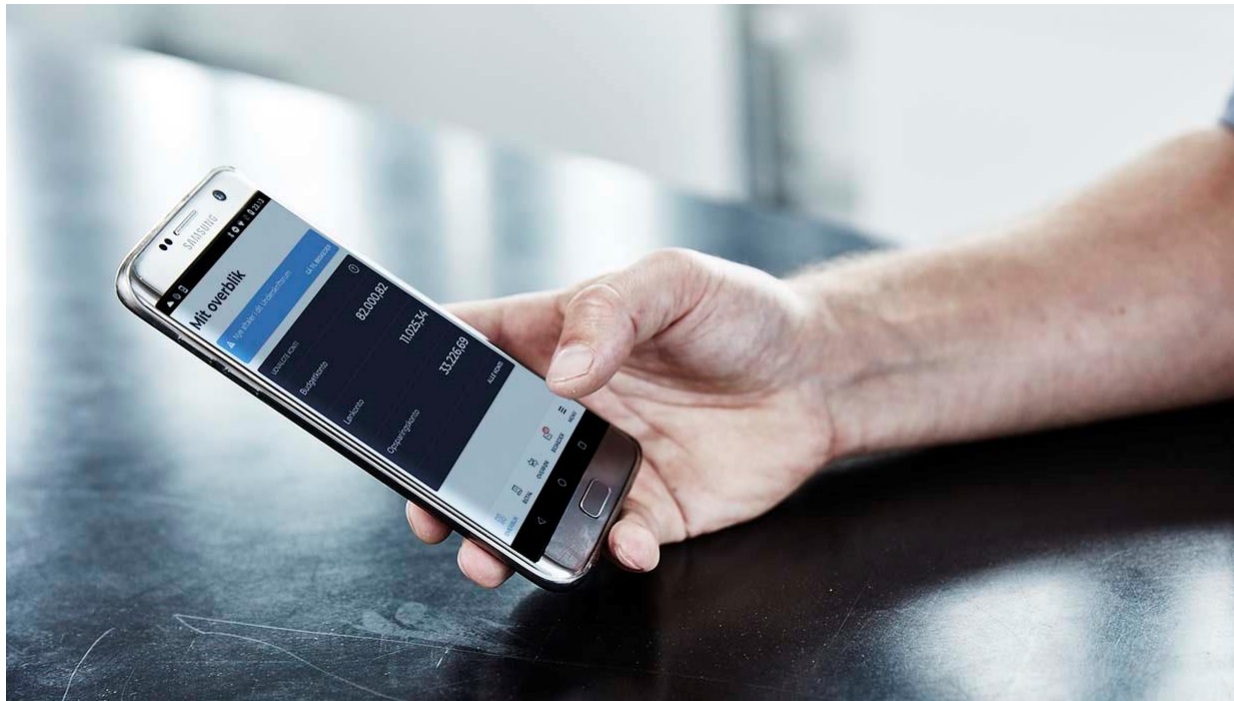
```
) else if (document.selection) {  
    input.focus();  
    var sel = document.selection;  
    var selen = document.selection;  
    sel.moveStart('character', 0);  
}
```

Fase 4: Test

- Usability test (testplan og tænke-højt-test)
- Ekspertgennemgang (det kommer vi tilbage til)
- Spørgeskemaer

Fase 5: Igangsætning

- IT systemet stilles til rådighed for brugerne

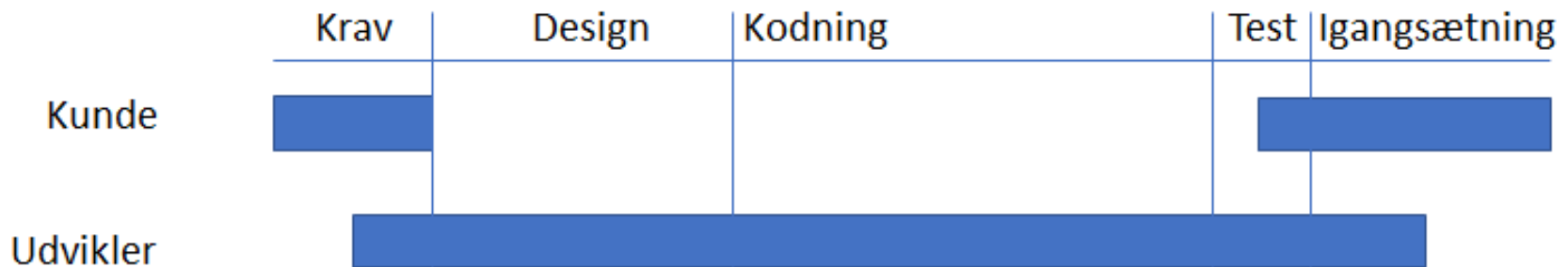


Opgave

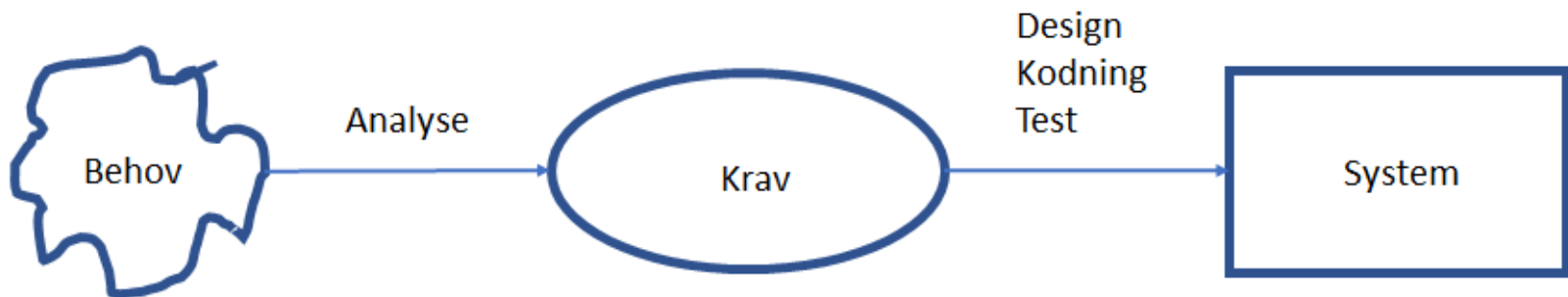


Ser I nogle
udfordringer ved
vandfaldsmodellen?

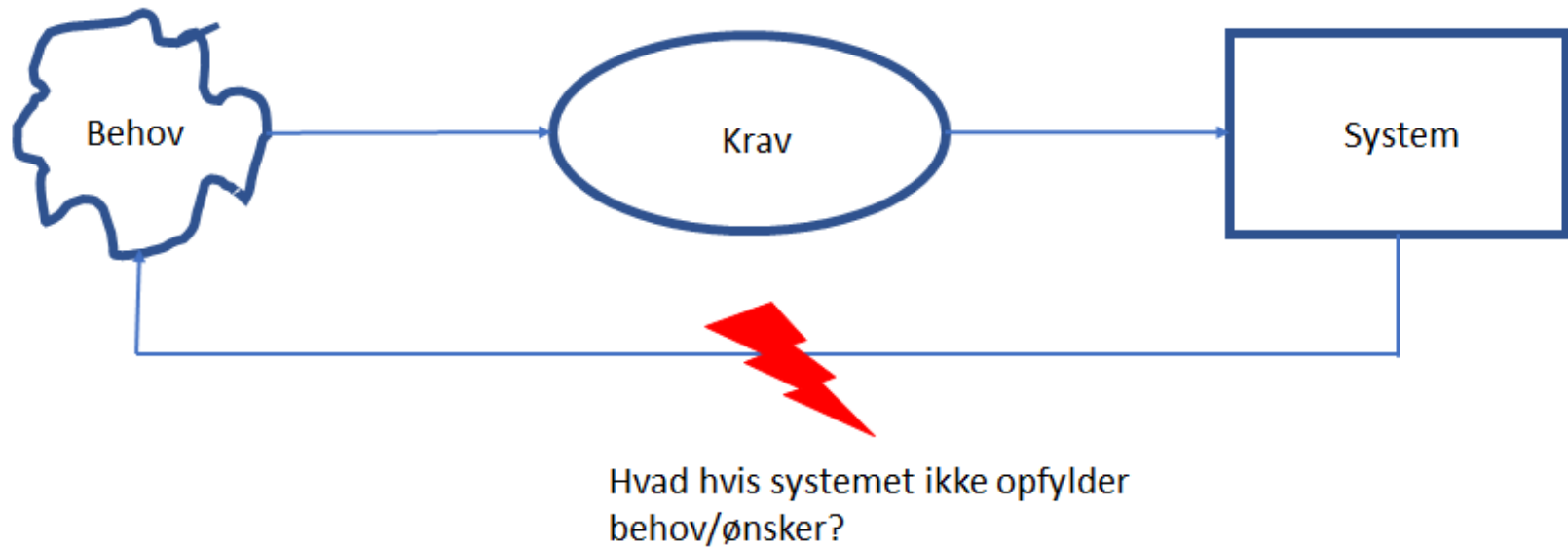
Kundens rolle



Krav som bindeled



Krav som bindeled



Problemer med kravspecificering

- Emperi viser, at fejl i kravspecifikationen er den hyppigste årsag til fejl

"Det gælder for alle projekterne, jeg har undersøgt, at man ikke vidste, hvordan man skulle stille krav til systemet. Det man gør galt er, at kunden med en lang liste beskriver, hvad man synes, systemet skal gøre. Derved giver man ikke leverandøren nogen frihed. Den løsning man får, kan gøre alle de ting, kunden har bedt om, men giver ikke en smidig arbejdsgang", udtaler Søren Lauesen i en pressemeddelelse.

Se: <https://www.magisterbladet.dk/aktuelt/2017/maj/professor-samme-fejl-er-skyld-i-offentlige-it-skandaler>

Opgør med vandfald og krav

The Agile Manifesto

Individuals and interactions	over	Processes and Tools
Working Product	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.

www.agilemanifesto.org

Skrevet i 2001 af 17 softwareudviklere

Opgør med vandfald og krav

The Agile Manifesto

Individuals and

Agil = bevægelig og adræt

**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.*

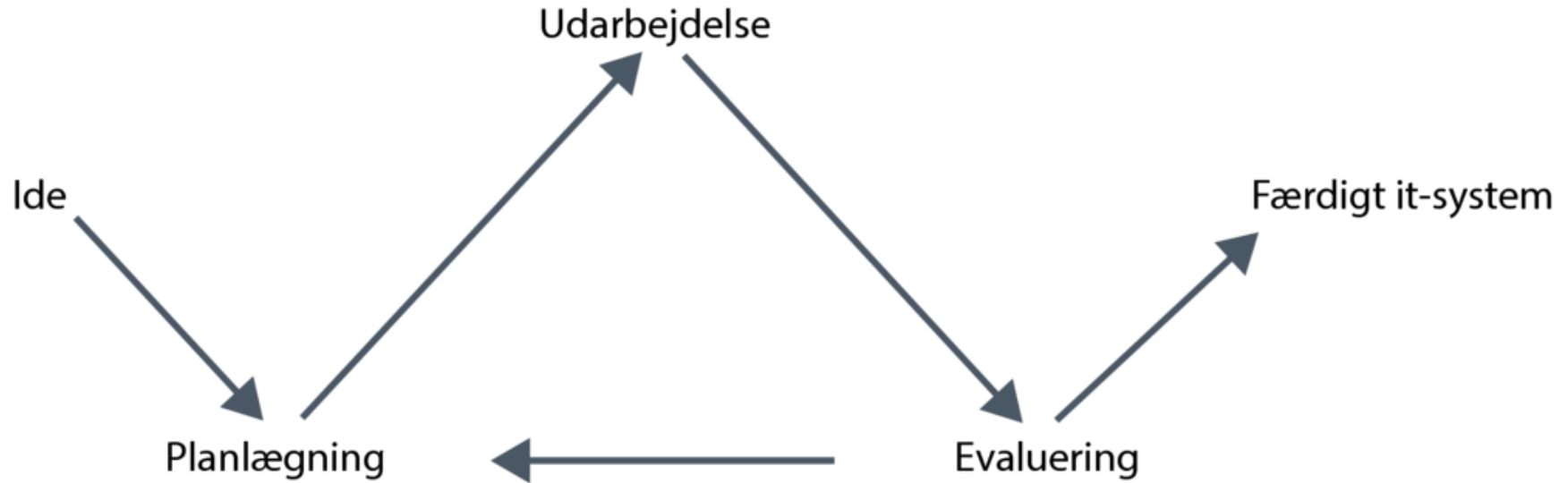
www.agilemanifesto.org

Skrevet i 2001 af 17 softwareudviklere

Agile principper

- 1 Vores højeste prioritet er at stille kunden tilfreds gennem **TIDLIGE OG LØBENDE AFLEVERINGER** af værdifuld software
- 2 Vi imødekommer **ÆNDRINGER I KRAV**, også selvom de fremkommer sent i udviklingen. De agile processer sikrer at ændringer er til kundens fordel
- 3 Løbende levering af velfungerende software, **JO HYPPIGERE DES BEDRE**
- 4 Forretningsrepræsentanter og udviklere skal **SAMARBEJDE DAGLIGT** gennem hele projektet
- 5 Opbyg projekterne omkring **MOTIVEREDE PERSONER**. Giv dem de rette omgivelser og den nødvendige støtte, og stol på at de kan klare opgaven
- 6 Den mest effektive kommunikationsform er samtale **ANSIGT TIL ANSIGT**
- 7 **FUNGERENDE SOFTWARE** er den primære måde at måle fremdrift på
- 8 Agile processer fremmer **BÆREDYGTIG UDVIKLING**, systemejere, udviklere og brugere bør til enhver tid kunne opretholde et fast udviklingstempo
- 9 Fokus på **TEKNISK EKSPERTISE OG GODT DESIGN** fremmer den agile udvikling
- 10 **ENKELTHED** defineret som kunsten at maksimere mængden af ikke udført arbejde er essentiel
- 11 De bedste arkitekturer, krav og designs stammer fra **SELVORGANISERENDE TEAMS**
- 12 Med jævne mellemrum **REFLEKTERER GRUPPEN** over, hvordan den kan blive mere effektiv, og tilpasser sin adfærd derefter

Den iterative arbejdsproces



- Hvordan kan vandfaldsmodellen repræsenteres med denne figur?

Gå en tur op forbi kontoret
og tilbage igen!

Tal om: Hvad ønsker I jer til
jul 😊

SCRUM

- En agil projektstyringsmodel
 - 3 roller
 - 3 værktøjer
 - 3 mødeformer

SCRUM – 3 roller

- Product owner:
 - overordnet ansvar for at IT-systemet bliver lavet
 - ansvar for product backlog (værktøj)
- Scrum master:
 - har ansvaret for at arbejdsmetoden SCRUM anvendes korrekt
- Scrum team:
 - dem der laver arbejdet (product owner og scrum master kan godt være en del af dette team)

SCRUM – 3 værktøjer

- Product backlog
 - en liste over alt det man ved skal være en del af IT-systemet
 - ofte beskrevet vha. user-stories (se næste slide)
- Sprint backlog
 - en delmængde af product backlog der skal arbejdes med i det næste sprint
- Burndown chart
 - en oversigt som viser hvor lang tid, man forventer at skulle bruge for at løse de resterende opgaver i et sprint

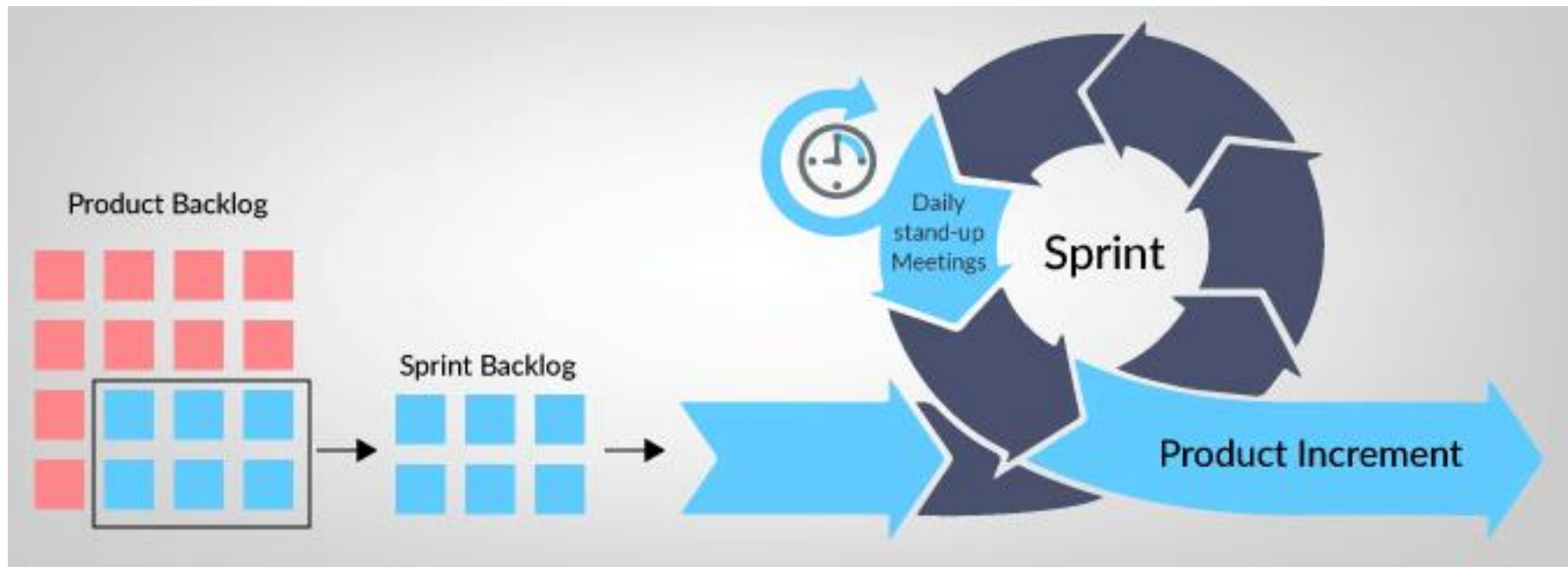
User stories (beskriver et behov)

“Som en <brugertype>
<skal/kan/vil/må> jeg
<handling>, så jeg
<formål>”

AULA:

Som forældre skal jeg kunne skrive en besked til en lærer, så jeg kan kontakte ham/hende, hvis mit barn skal til tandlæge

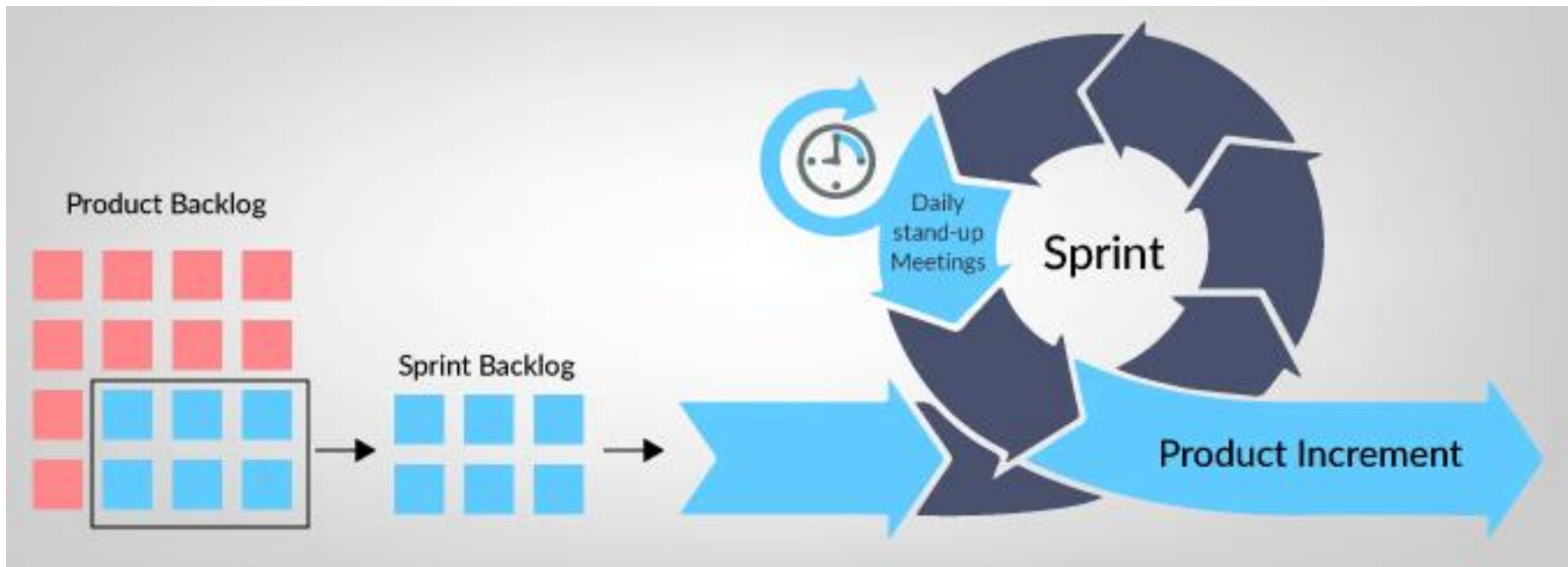
SCRUM – sprint (typisk 1-3 uger)



SCRUM – 3 mødeformer

- Sprint planning
 - inden et sprint skal sprintet planlægges: hvad skal sprint backlog bestå af?
- Daily scrum
 - det daglige møde: hvad har vi nået siden sidst? hvad skal vi nå i dag?
- Sprint review
 - efter hvert sprint: hvordan gik sprintet? nåede vi det vi skulle? er der noget som skal ændres (ændringer til product backlog)?

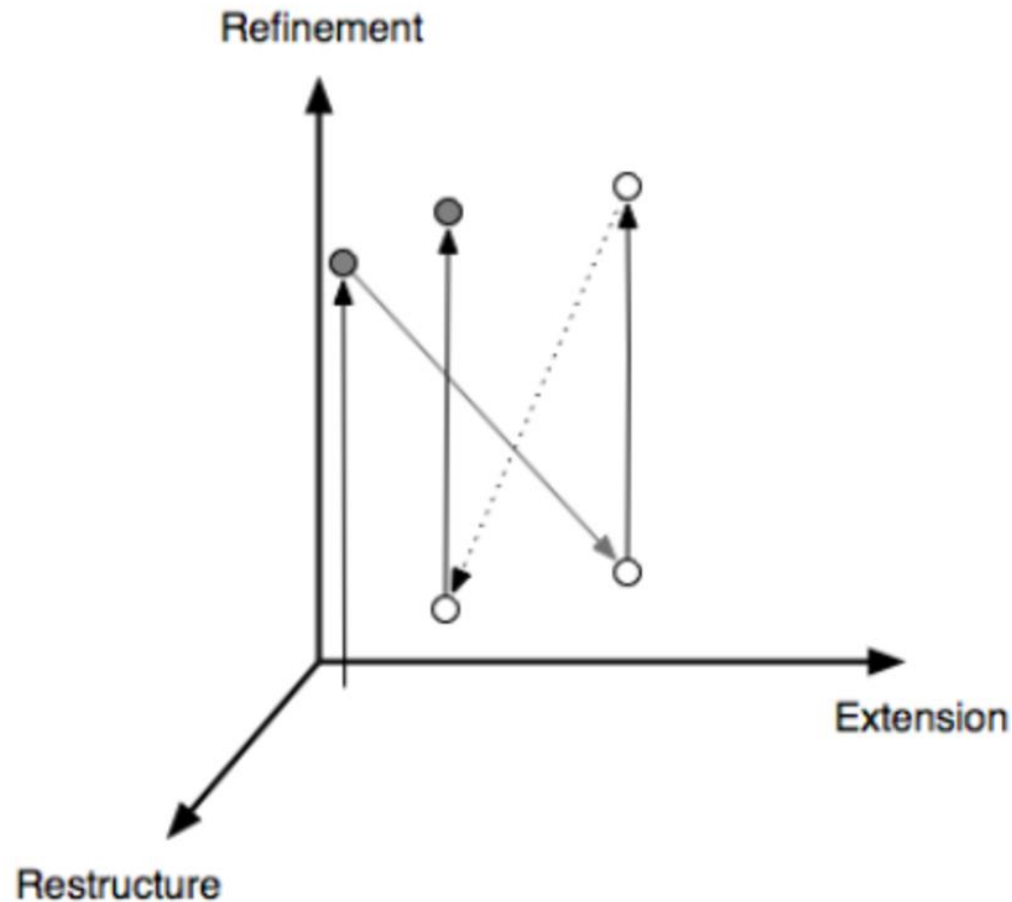
SCRUM



SCRUM – på 7 minutter

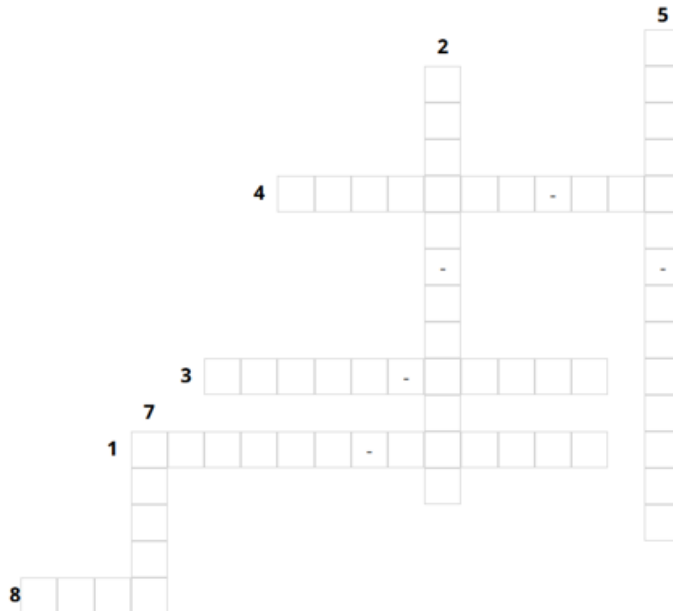
[SCRUM på 7 minutter](#)

Hvordan hænger SCRUM sammen med stepwise improvement?



Opgave

SCRUM krydsord (fra "Informatik", Systime)



1	Evaluering af sprint
2	Den, der er ansvarlig for, at scrum-metoden anvendes
3	Løbende statusopførelse
4	Har det overordnede ansvar for projektet
5	Beskrivelse af delopgave til sprint
6	En iteration
7	Projektledeelse
8	Arbejdsgruppe

Opgave

- Din mor er blevet træt af at rydde op efter dig, så nu skal du flytte i din egen lejlighed!
 - Hvilke delopgaver skal product backlog bestå af?
 - Hvor mange sprints vil du lave?
 - Hvilke delopgaver skal de forskellige sprint backlogs bestå af?