

IN 2000

30 januar 2024

Teamarbeid og smidig metodikk. Lean og Scrum. Prosjektarbeid



**UNIVERSITETET
I OSLO**

Yngve Lindsjørn

ynglin@ifi.uio.no

Fasilitering av møte

- Teamet skal ha et møte, og i gjennomgangen av oppgaver som skulle vært gjort siden siste møte ser man at en oppgave ikke har blitt gjort, og den har hengt igjen også fra tidligere.
- Poenget her er **fasilitering** - fange opp situasjonen og gjøre det beste ut av det for å komme videre.

Skuespill

Oppsummering av skuespill

- Poeng: Fange opp situasjonen og gjøre det beste ut av det
- Fasilitator prøver å løse problemet med oppgaven som ikke har blitt gjort, i fellesskap, saklig og respektfullt.
- Kjefter eller bebreider ikke, men skjuler heller ikke at dette er andre gang oppgaven er forsømt.
- Fasilitator prøver å få hjelp av andre til å løse oppgaven
 - uttrykk for tvil om at personen vil klare det alene,
 - håndsrekning (hjelp og sparring).
- Da det blir for mye og personen lover å gjøre det, endrer fasilitator hjelpen til en oppringning, en midtveis sjekk, slik at teamet på neste møte ikke igjen skal sitte med oppgaven uløst.
- Merk også at den litt krasse talen fra fasilitator ikke er rettet mot personen som har forsømt oppgaven, men mot vedkommende som prøver å slippe unna. Alle er selvsagt klar over hvor den primære forsømmelsen ligger!

Fasilitering av arbeidet er viktig

- **Eksempel – skrive en rapport sammen**
 - Hva skal kapitlet handle om, og hvordan strukturere underkapitler.
 - Viktig at teamet diskuterer hvordan en oppgave skal løses sammen.
 - Vanskelig å skrive oppgave sammen. Noen vil at det skal være bedre, og noen er flinkere til å skrive enn andre.
 - Må ofte gi slipp på det perfekte.
 - Fordeling av oppgaver er en av de største utfordringene.
 - Diskuter opplegget.
 - hva er målet etc. før fordeling.

Rapport

- Start å skrive tidlig (ikke vent til dere er ferdige med å utvikle appen) – 40-60 sider.
- Struktur
 - Presentasjon av teamet og case
 - Kort sammendrag
 - Brukerdokumentasjon (noen skjermbilder ...)
 - Kravspesifikasjon (backlog- user stories,...)
 - Modellering (use case diagram, klassediagram, sekvensdiagram...)
 - Produktdokumentasjon (mer teknisk; arkitektur, teknologier....)
 - Testdokumentasjon (enhetstesting, integrasjonstesting,...)
 - Prosessdokumentasjon
 - Endringer, sprinter (sprint backlog – fra user stories til tasks)
 - Refleksjon (hvordan har teamet fungert, hva kunne vært gjort annerledes.....)
 - Referanser, kilder
 - Eventuelle vedlegg (f.eks. intervjuguide, spørreskjema, eksempel på kildekode...)

Fasilitator

- Må ha en form for styring og struktur
 - Hvis ikke - lett for at de som dominerer tar styringen
- Fasilitator
 - Styrer formen – ikke innholdet
 - Struktur, rekkefølge
 - Ikke prosjektleder, ingen ansvar for hvem som skal utføre hvilke oppgaver
 - Fasilitator har kun ansvar for å sikre at samtalen og arbeidet i teamet organiseres eller formes best mulig.
 - Hva teammedlemmene presterer innenfor fasilitators rammer, er *deres* ansvar.

Ryktespredning

«Blir plutselig oppmerksom på at det går rykter om meg og en annen, at vi ikke stiller opp og ikke lager noe eller bidrar i teamarbeidet. Og det ryktet gjør at det det ingen som vil ha med oss i nye team»

Teammedlemmer som ikke bidrar

- Hva skal vi gjøre når X aldri stiller opp på møtene? X er sjelden med i undervisningen og det virker som det er helt andre ting enn prosjektarbeidet som blir prioritert.
 - Skal vi ha med navnet på prosjektrapporten?

Manglende deltagelse i teamarbeidet

Generelt to årsaker:

1. Personlige årsaker:

- Vanskeligheter i privatlivet eller sykdom
- En feilaktig forestilling om at kurset kan gjennomføres med minimal innsats
- Manglende motivasjon

2. Teamdynamikken. Kan bli kjørt ut på et sidespor. Deltar i starten, men opplever at det blir vanskelig å bidra, pga. teamets måte å jobbe på.

«Jeg deltok i starten, men hver gang vi produserte noe, ble alt det jeg hadde gjort skrevet om. Jeg følte meg dum og verdiløs og lot etter hvert være å møte opp»

Punkter i møte

- **Lage dagsorden** (forberedelse)
- **Start gjerne 5, 10 eller 15 minutter over hel**, så folk får tid til å lande, få en kopp kaffe, sagt hei etc. Snakker som regel mest med dem man kjenner best. Fasilitator oppmerksom på at for eksempel en sitter alene.
- **Start til avtalt tid**. Viktig å sende et signal om at det ikke er OK å komme for sent. Alle kan komme for sent av og til, men ikke hver gang! Ikke gjør stort nummer av det, men la det likevel få konsekvenser!
- **Oppfølging av avtaler fra forrige møte**. Kan være følsomt, da mange kanskje ikke har fulgt opp. Fasilitator spiller en viktig rolle her. Søker løsning er viktig, igjen smidig. Viktig ikke å henge ut den som ikke har gjort det den skal, men likevel det blir lagt merke til!
- **Fasilitering av skriving**. Styre prosessen for å få form og struktur på oppgaveskrivingen, ikke om det skal brukes 2 eller 6 timer på problemstilling. Må sørge for felles forståelse, få folk til å gi feedback til hverandre...
- **Fordeling av oppgaver**. Fasilitator skal ikke bestemme hvem som skal gjøre hva (er ikke BOSS), men kan sikre at beslutninger tas og at noen utfører dem slik at teamet får gjort et de skal. (for eks. det med syk hund – må da sørge for å finne en løsning).
- **Konklusjon og avslutning av møtet**. Kan være vanskelig for en selvstyrt gruppe å avslutte, ofte noen som har mer å si. Fasilitator må ta en vurdering. Slutt i god tid (noen tripper for å komme av gårde, noen diskuterer videre) slik at man har tid til å gjøre nye avtaler. Viktig at F også ser om alle punktene nås på dagsorden for eks en halvtime før slutt for å se hvilke punkter som kan utsettes til neste gang.
- **Evaluering av møtene – bruk retrospektiv**. Kan også ha retrospektiv av selve møtene. Hva fungerer, hva kan gjøres bedre. Gjerne tre ting (en på hva går bra, to på hva kan gjøres bedre). a) fasiliteringen, b) alle i teamet, c) et bestemt teammedlem.

Teamkontrakt

- Kan være en god ide å lage en teamavtale eller teamkontrakt
- Kan være en del av oblig 3 og kan for eksempel inneholde noe om:
 - Tilstedeværelse
 - Tidsbruk
 - Forventninger til den enkeltes bidrag
 - Hva skjer ved avvik eller uenigheter

Jobbe sammen i team

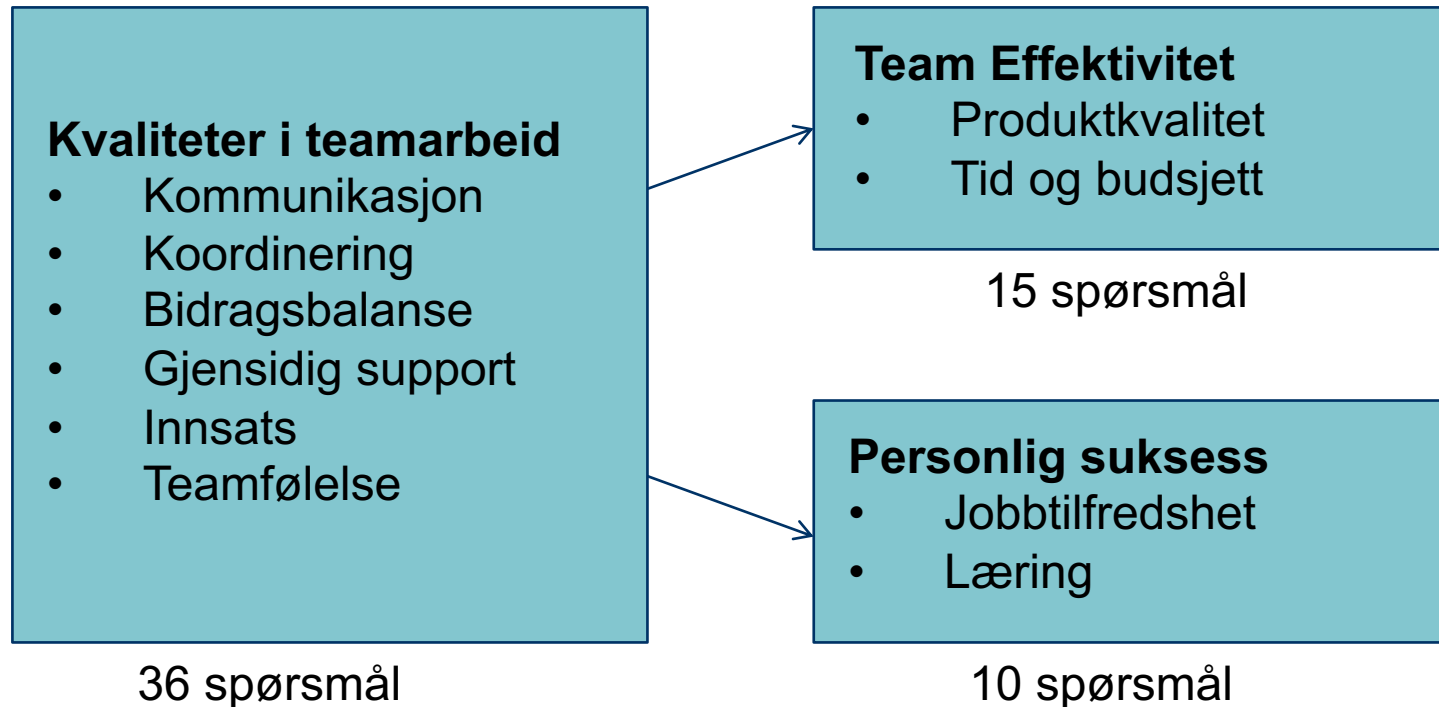
- Arbeid er koblet
- Holder hverandre gjensidig ansvarlige
- Selvstyrte team
- Teamets mål er viktigere enn eget mål

Nils Arne Eggen

– fra boka Godfoten

- «Det viktigste i teorien er å gjøre andre bedre. En Rosenborgspiller skal alltid gå på banen med den hensikten at vedkommende skal gjøre andre på laget bedre».
- «Du er ikke dyktig hvis du ikke kan bruke dyktigheten din til å gjøre andre dyktige».
- «Det er laget og kollektivet som er viktig, ikke den enkelte spiller eller trener. Hele samhandlingsprosessen handler om å utvikle enkeltindividet, gjerne eneren, slik at vedkommende gjør kollektivet enda bedre. En fotballspiller er ikke et produkt av seg selv, men et produkt av laget».

Teameffektivitets-modell i studie



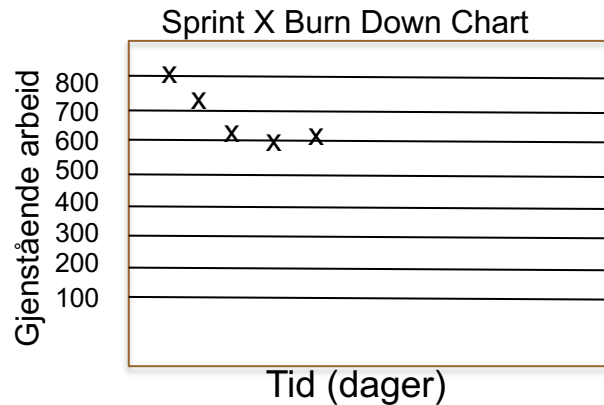
Lindsjørn et. al, 2016: Teamwork quality and project success in software development: A survey of agile development teams

Lindsjørn, Almås and Stray: A Case Study of Teamwork and Project Success in a Comprehensive Capstone Course

SCRUM - prosess og roller

Daily Standup – 3 spørsmål

- Hva har du gjort siden i går?
- Hva planegger du å gjøre til i morgen?
- Hvilke eventuelle hindringer har du?



Scrum Master (fasilitator)

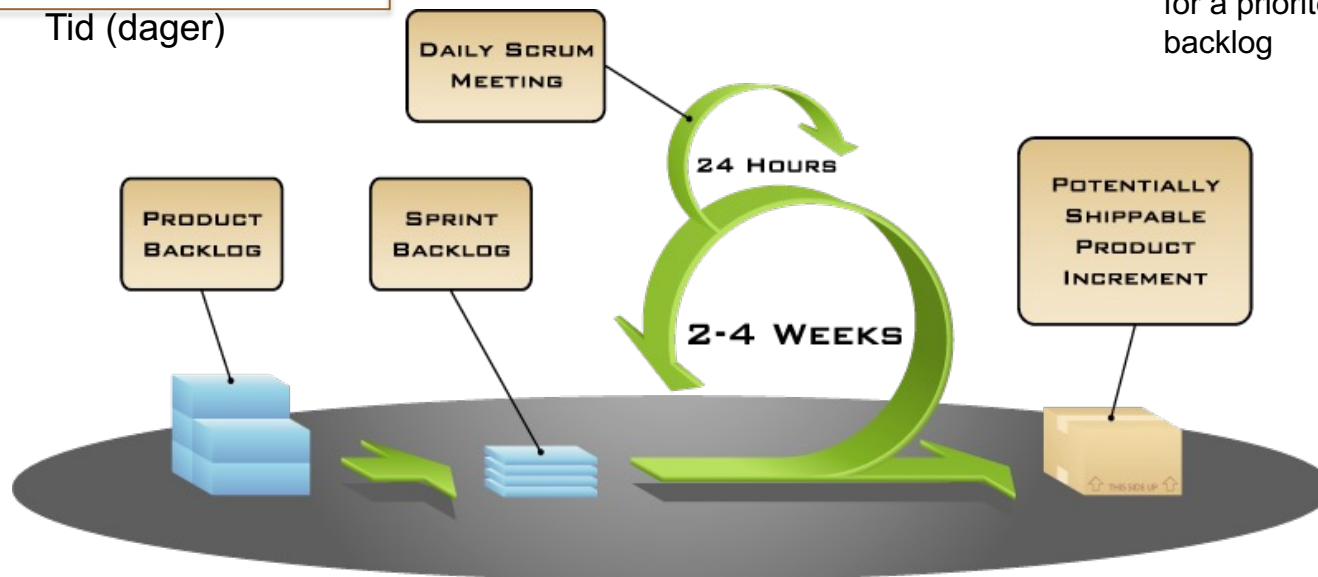
- Sørger for at Scrum prosesser blir fulgt
- Leder "daily stand up"
- Hindrer støy slik at utviklerne kan fokusere på oppgavene
- Teamets "coach og beste venn"
- Koordinerer SPRINT planlegging, og estimering av Sprint backlog

Scrum Team

- Typisk 5-8 personer
- Utviklere, testere, arkitekt, ...
 - Jobber fulltid
- Selvstyrt
- Utvikler systemet

Product Owner

- Representerer kunden
- Setter opp mål for hver SPRINT
- Ansvar for product backlog, og for å prioritere aktiviteter i backlog



Product Backlog (produktkø)



	Nye estimater i Sprint					
Prioritet	Brukerhistorie	Estimat (Initielt)	1	2	3	4
1	Som en bruker ønsker jeg å få en oversikt over luftkvalitet i Oslo en bestemt dag	8				
....					
	Restrukturer koden for å ta i bruk pattern MVVM (Model View View Model)	13				

Product Backlog (produktkø)



- Ulike “items”
 - Kunde spesifikke (Få en oversikt over luftkvaliteten i byer)
 - Forbedringsmål (Restrukturere kode)
- “Items” i produktkøen kan uttrykkes som
 - User stories
 - Use Cases
 - Andre måter å spesifisere krav på som passer

Sprint planlegging

En måte å lage sprint backlog



		Nye estimater etter dag					
Brukerhistorie	Oppgave i Sprint	Estimat (Initielt)	Team medlem	1	2	3	4
Som en bruker ønsker jeg å få en oversikt over luftkvalitet i Oslo en bestemt dag	Finne riktig API og tolke data	2	Syver				
	Lage datastruktur - klasser	5	Anne				
	Vise kart med fargekode	8					
	Gi tekstlig beskrivelse i kartet	5					
Restrukturer koden for å ta i bruk pattern «Model View Presenter» eller MVVM							

En ide: se på mob programming – se for eksempel https://en.wikipedia.org/wiki/Mob_programming

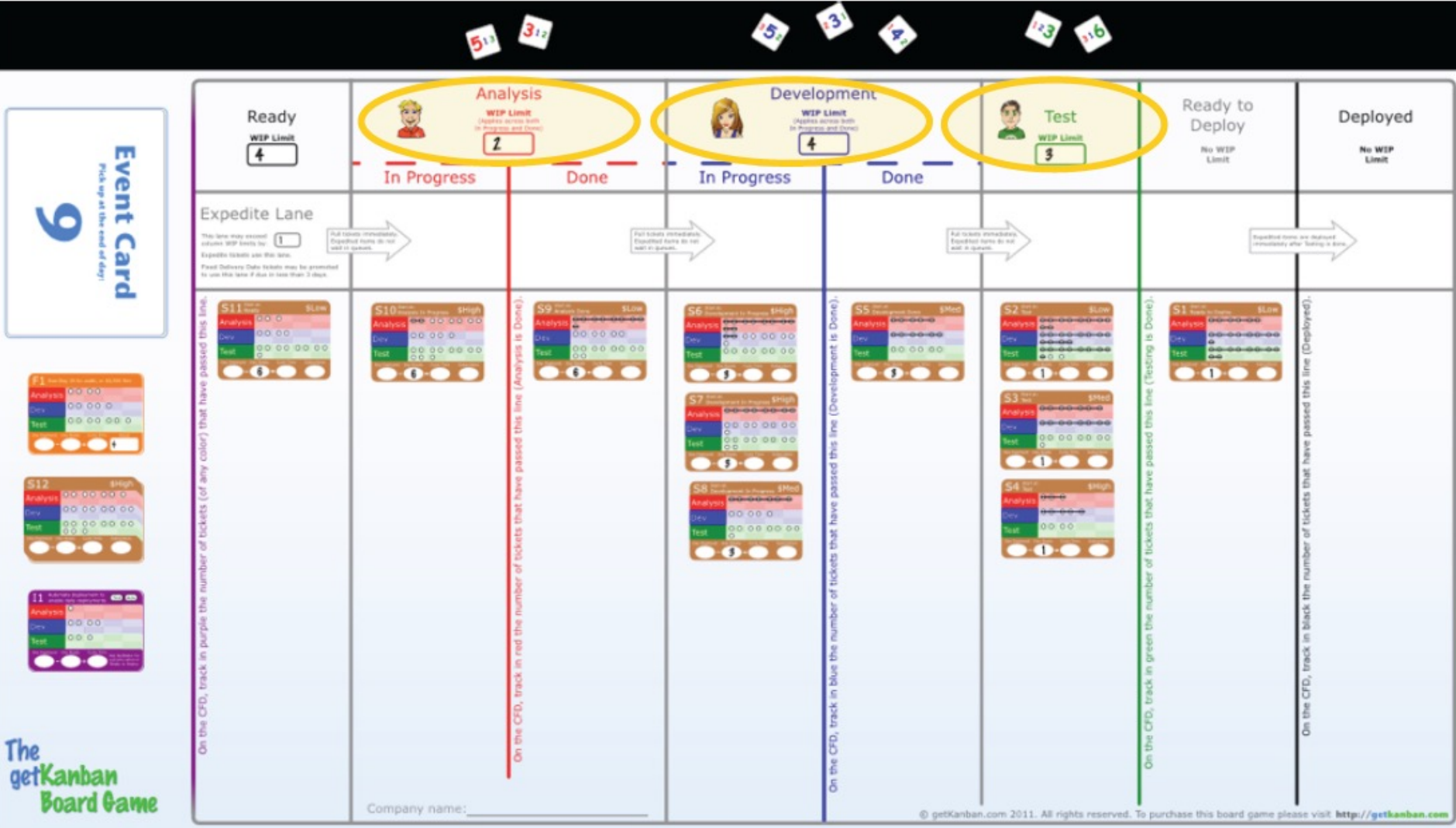
Lean –viktige prinsipper

- JIT (Just in Time)
- Tilfredsstillte kunden
- Flyt
- Visualisering
- Unngå “Waste”
- Støtte opp om endringer

Lean prinsipp: Just-In-Time (JIT)

- Ikke lag noe før det er etterspurt.
- Ikke ta endelige avgjørelser for tidlig.
 - Mange detaljer kan utsettes etter at fundamentale aspekter er implementert.
 - Fokuser kun på detaljer som er nødvendig for å implementere oppgavene som er i prosess.
- Unngå waste:
 - Kan vise seg at enkelte oppgaver ikke er nødvendige å implementere i det hele tatt.
 - Ved å utsette oppgaver blir de ofte bedre implementert enn om de blir tatt tidlig.
 - Reduser flaskehalsler.

Visualisering



Waste

- Alt som krever ressurser
 - tid
 - arbeids innsats
 - utstyr og lager
 - penger
- Og som ikke gir verdi for kunden

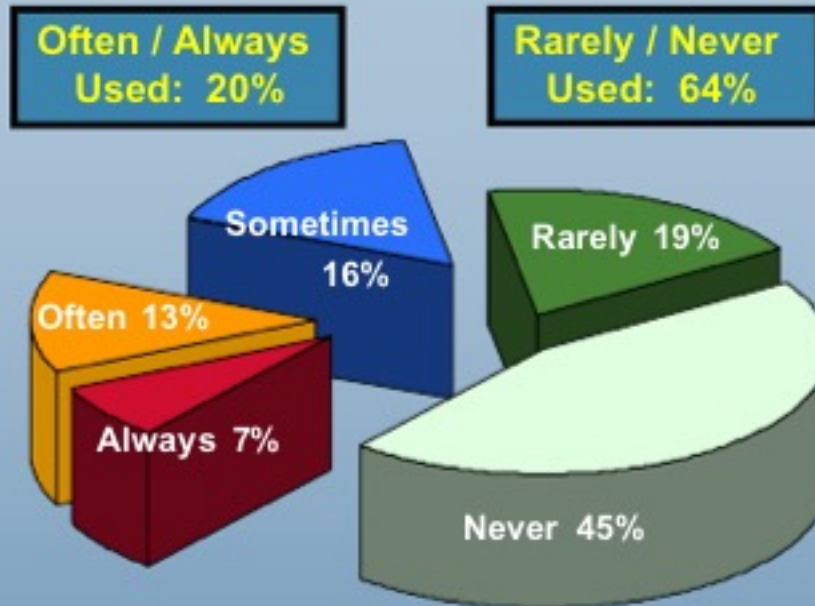
Noen viktige grunner til “waste” i programvareutvikling

- Kompleksitet
- For stort skille mellom de som tar avgjørelser og de som gjør jobben
- Teknisk gjeld

Waste 1: Extra Features

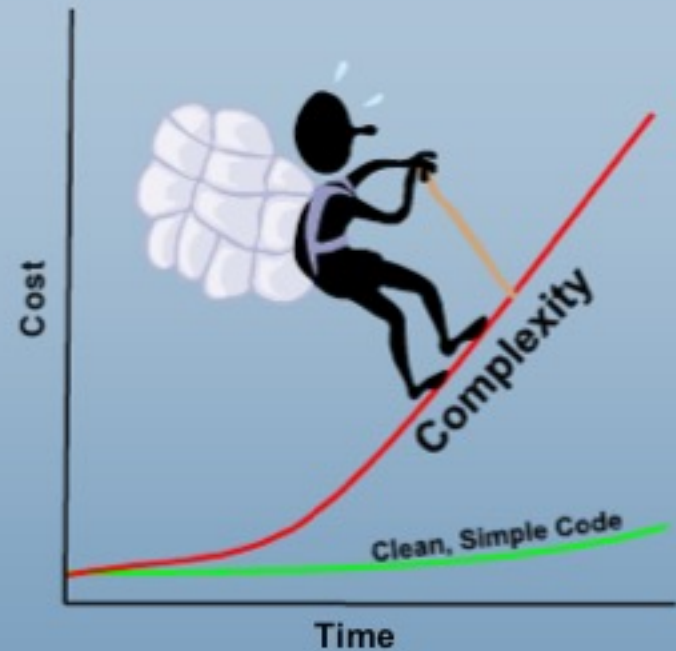


Features / Functions Used in a Typical System



Standish Group Study Reported at XP2002 by Jim Johnson, Chairman

Cost of Complexity



The Biggest opportunity for increasing Software Development Productivity: Write Less Code!

Grunner til waste: Teknisk gjeld

- En metafor som beskriver
 - “shortcuts” som tas i utvikling og vedlikehold for å oppnå kortsiktige fordeler
 - Men som i det lange løp vil føre til økte kostnader på vedlikehold og videre utvikling.



Mer om teknisk gjeld

- = "Alt som gjør det vanskelig å endre/modifisere kode"
- Siden kode alltid må vedlikeholdes/endres, må vi betale tilbake gjelden.
- Skriv "clean code", konsis, enkel med få avhengigheter
- Ny funksjonalitet leder til større kodebase og økt kompleksitet; refaktorering reduserer gjelden.

Team effektivitet

- Finnes mange modeller
- Hva er effektivt?
- Hvordan få alle teammedlemmene motivert og få ut potensiale?
- Kommunikasjon, koordinering av oppgaver, god teamfølelse, tillit til hverandre, teamledelse,.....

Kvaliteter i teamarbeid – team effektivitetsmodell

Kvaliteter i teamarbeid

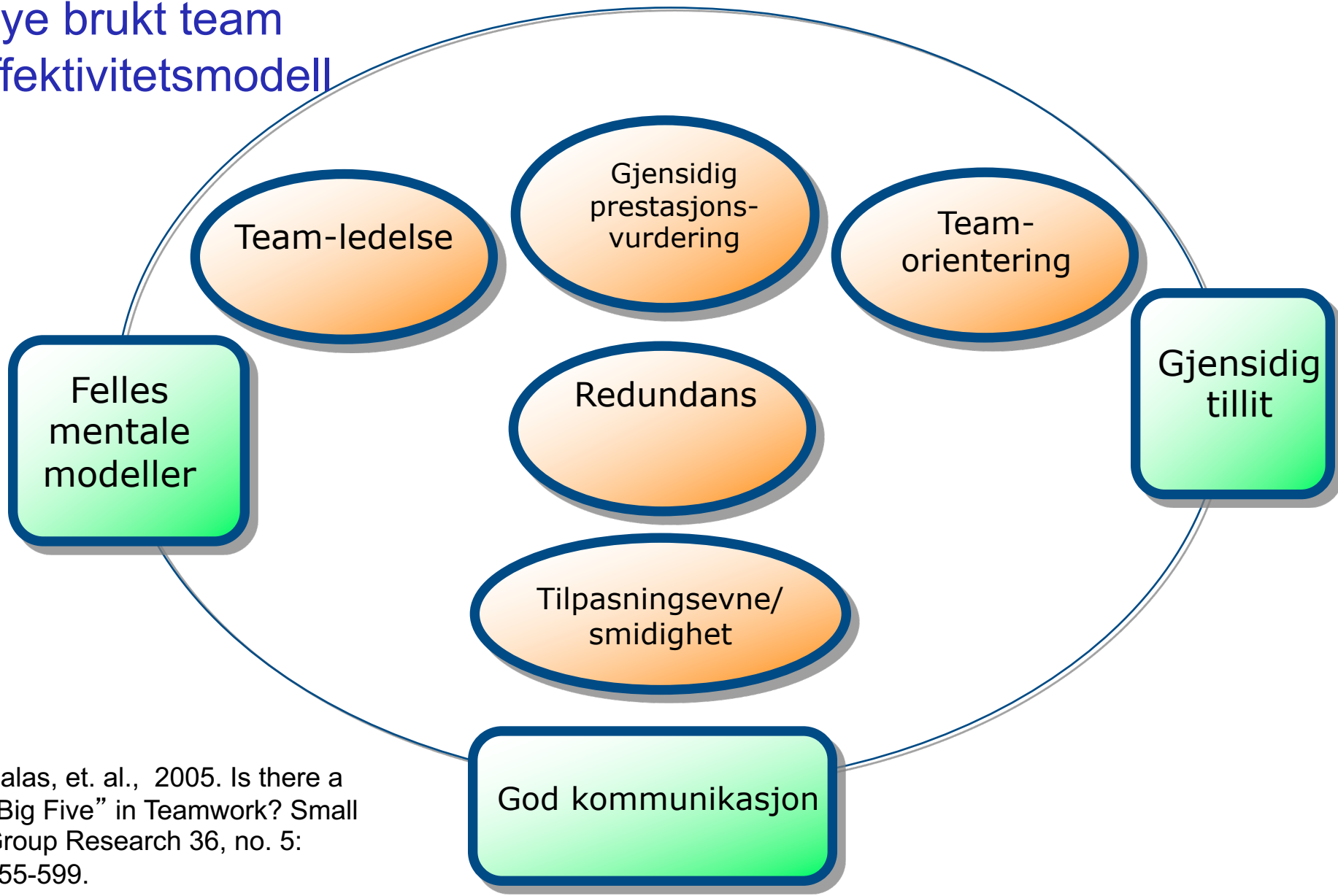
- Kommunikasjon
- Koordinering
- Bidragsbalanse
- Gjensidig support
- Innsats
- Teamfølelse

The five keys to a successful Google team

- **Psychological safety:** Can we take risks on this team without feeling insecure or embarrassed?
 - Teammedlemmene tør å føle seg sårbare for hverandre
- **Dependability:** Can we count on each other to do high quality work on time?
 - Teammedlemmene får ting gjort med god kvalitet (Googles krav)
- **Structure & clarity:** Are goals, roles, and execution plans on our team clear?
 - Teammedlemmene har klare roller, planer og mål
- **Meaning of work:** Are we working on something that is personally important for each of us?
 - Arbeidet er personlig viktig for teammedlemmene
- **Impact of work:** Do we fundamentally believe that the work we're doing matters?
 - Teammedlemmene føler at arbeidet betyr noe og skaper endring

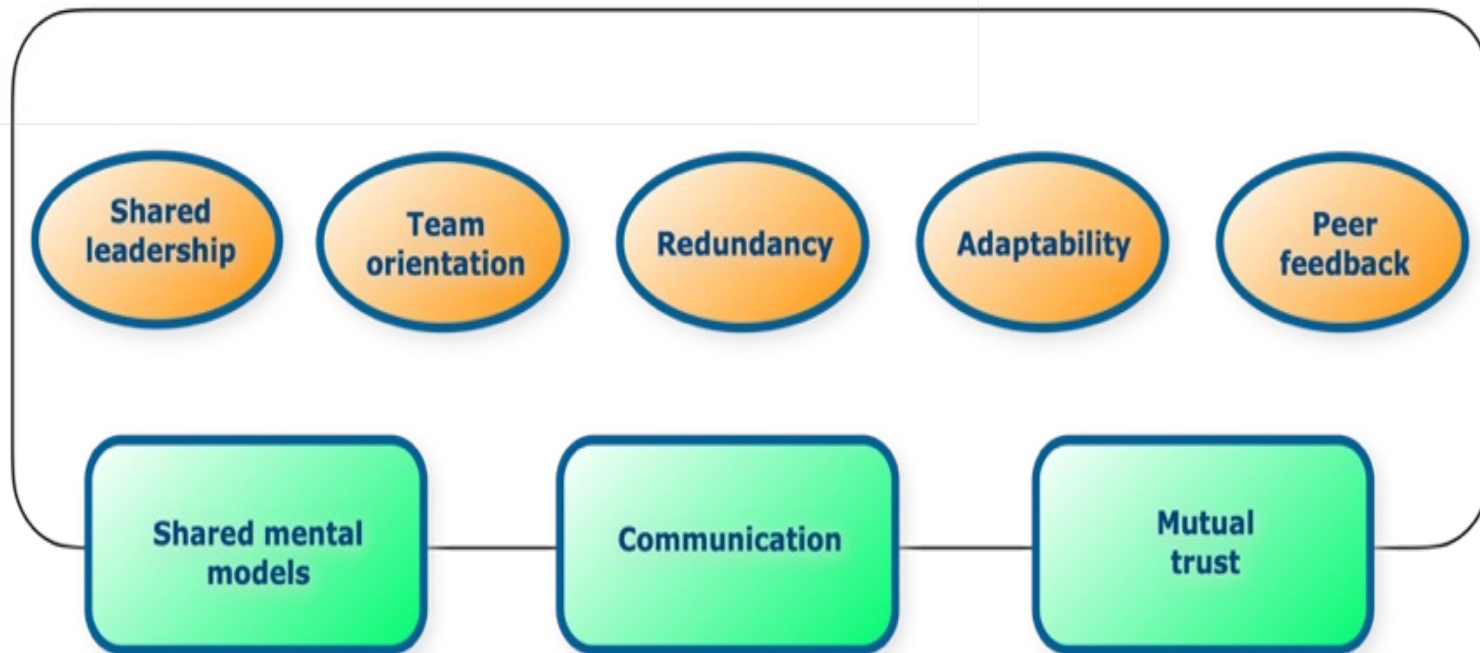
Les mer på <https://rework.withgoogle.com/blog/five-keys-to-a-successful-google-team/>

Mye brukt team effektivitetsmodell



Salas, et. al., 2005. Is there a “Big Five” in Teamwork? Small Group Research 36, no. 5: 555-599.

Revidert «Big five» model – Agile Teamwork Effective Model«ATEM»



Strode, D., Dingsøy, T., Lindsjörn, Y.: *A teamwork effectiveness model for agile software development*, [*Empirical Software Engineering* volume 27](#), Article number: 56, March 2022.

Coordination mechanisms with behavioural markers in ATEM model

Teamwork coordination mechanism	Behavioural markers
Shared mental models	<ul style="list-style-type: none">• Anticipating and predicting each other's needs.• Common understanding of work process.• Common understanding of tasks.• Common understanding of individual skills.
Communication	<ul style="list-style-type: none">• The team follows up on progress of tasks.• Visualising project information.• Facilitate informal communication.
Mutual trust	<ul style="list-style-type: none">• Information sharing.• Willingness to admit mistakes and accept feedback.• Supportive environment.

Teamwork components with behavioural markers, ATEM model

Teamwork component	Behavioural markers
Shared leadership	<ul style="list-style-type: none">• The agile team facilitates team problem solving.• The agile team synchronises and combines individual team member contributions using agile practices combined with automated tools.• The agile team seek and evaluate information that affects team functioning.• Agile values and methodologies determine team member roles.• Agile values and methodologies determine the frequency and type of preparatory meetings and feedback sessions.• A servant leader facilitates a boundary-spanning function.
Team orientation	<ul style="list-style-type: none">• Taking into account alternative solutions provided by teammates and appraising that input to determine what is most correct.• Increased task involvement, information sharing, strategizing, and participatory goal setting.• The team "sticks together and remains united".
Redundancy	<ul style="list-style-type: none">• Recognition by potential backup providers that there is a workload distribution problem in their team.• Shifting of work responsibilities to underutilized team members.• Completion of the whole task or parts of tasks by other team members.
Adaptability	<ul style="list-style-type: none">• Identify cues that a change has occurred, assign meaning to that change, and develop a new plan to deal with the changes.• Identify opportunities for improvement and innovation for habitual or routine practices.• Remain vigilant to changes in the internal and external environment of the team.
Peer feedback	<ul style="list-style-type: none">• Identifying mistakes and lapses in other team members' actions.• Regular feedback regarding team member actions to facilitate self-correction.

Koordinasjonsmekanismer med atferdsmarkører i ATEM-modell

Koordinerings- mekanismer for teamarbeid	Atferdsmarkører
Delte mentale modeller	<ul style="list-style-type: none">• Forutse og forutsi hverandres behov.• Felles forståelse av arbeidsprosess.• Felles forståelse av oppgaver.• Felles forståelse av individuelle ferdigheter.
Kommunikasjon	<ul style="list-style-type: none">• Teamet følger opp fremdrift av oppgaver.• Visualisere prosjektinformasjon.• Tilrettelegge for uformell kommunikasjon.n.
Gjensidig tillit	<ul style="list-style-type: none">• Informasjonsdeling.• Vilje til å innrømme feil og ta imot tilbakemeldinger.• Støttende miljø.

Teamarbeidskomponenter med atferdsmarkører, ATEM-modell

Teamarbeidskomponent	Atferdsmarkører
Delt ledelse	<ul style="list-style-type: none">• Det smidige teamet legger til rette for teamproblemløsning.• Det smidige teamet synkroniserer og kombinerer bidrag fra individuelle teammedlemmer ved å bruke smidige praksiser kombinert med automatiserte verktøy.• Det smidige teamet søker og evaluerer informasjon som påvirker teamfunksjonen.• Smidige verdier og metoder bestemmer rollene til teammedlemmene.• Smidige verdier og metoder bestemmer hyppigheten og typen av forberedende møter og tilbakemeldinger.
Teamorientering	<ul style="list-style-type: none">• Ta hensyn til alternative løsninger levert av andre teammedlemmer og vurdere disse innspillene for å finne ut hva som er mest riktig.• Økt oppgaveinvolvering, informasjonsdeling, strategisering og deltakende målsetting.• Teamet "holder sammen og forblir samlet".
Redundans	<ul style="list-style-type: none">• Erkjennelse av at det er et problem med fordeling av arbeidsbelastning i teamet deres.• Flytting av arbeidsansvar til underutnyttede teammedlemmer.• Gjennomføring av hele oppgaven eller deler av oppgaver av andre teammedlemmer.
Tilpasnings- evne	<ul style="list-style-type: none">• Identifiser tegn på at en endring har skjedd, tildel endringen mening og lag en plan for å håndtere endringene.• Identifiser muligheter for forbedring og innovasjon for vanlige eller rutinemessige praksiser.• Vær oppmerksom på endringer i teamets interne og eksterne miljø.
Peer feedback	<ul style="list-style-type: none">• Identifiser feil og bortfall i andre teammedlemmers handlinger.• Regelmessige tilbakemeldinger angående teammedlemshandlinger for å lette selvkorrigeringsprosessen.