

INF1050: Systemutvikling

05. februar 2013

Prosjektledelse, prosjektplanlegging, teamarbeid



**UNIVERSITETET
I OSLO**

Universitetslektor Yngve Lindsjørn

Kort om meg

- Yngve Lindsjørn (ynglin@ifi.uio.no)
 - Universitetslektor ved Ifi (fra 2010)
 - Master Informatikk, Ifi, UiO, 1987
 - Forsker ved Norsk Regnesentral (IT forskning) 1987-97
 - Drev eget programvareselskap, Kompetanseweb, i 8 år
 - Lang erfaring fra næringslivet – med blant annet prosjektledelse for store IKT prosjekter

Temaer i dagens forelesning

- Prosjektstyring/Prosjektledelse (Project Management)
- Risikohåndtering
- Teamarbeid
- Prosjektplanlegging
- Ledelse av smidige prosjekter – bruk av SCRUM

Utvikling av programvare - Prosjektstyring

- Suksesskriterier
 - Leverer programvaren til rett tid
 - Holde kostnadene innen budsjett
 - Leverer programvare som møter kundens forventninger
 - Opprettholde et velfungerende utviklingsteam

Prosjektstyring – Hva er spesielt med utvikling av programvare?

- Produktet er uhåndgripelig
- Programvare-prosjekter er ofte “engangs-”prosjekter
- Programvareprosesser er variable og organisasjons-spesifikke

6 viktige temaer innen prosjektstyring/prosjektledelse

- | | |
|-------------------|--------------------------------|
| • Forretningsplan | Hvorfor? |
| • Organisasjon | Hvem? |
| • Kvalitet | Hva? |
| • Plan | Hvordan? Hvor mye? Når? |
| • Risiko | Hva hvis? |
| • Endring | Hvilke endringer gir systemet? |

Risikohåndtering

- En risiko er en sannsynlighet for at uønskede omstendigheter skjer
 - **Prosjekt-risikoer** vil ha effekt på tidsplanen og/eller ressurser
 - **Produkt-risikoer** vil ha effekt på kvaliteten eller av programvaren som utvikles
 - **Forretnings (Business)-risikoer** vil ha effekt på organisasjonen som utvikler eller eier programvaren

Ulike typer av risiko – fra Sommerville s.599

Risiko type	Mulige risikoer
Teknologi	Databasen som brukes i systemet klarer ikke å prosessere så mange transaksjoner per sekund som forventet
Mennesker	Umulig å rekruttere mennesker med den kompetanse som kreves Nøkkelpersonell ikke tilgjengelig i kritiske faser
Organisasjon	Organisasjonen blir restrukturert slik at ulik ledelse er ansvarlig for prosjektet
Verktøy	Ulike programvareverktøy lar seg ikke integrere
Krav	Endringer av krav krever omfattende "redesign"
Estimering	Underestimering (i tid) av programvareutviklingen Tiden det tar å rette feil er underestimert

Risikoanalyse

- Vurder sannsynlighet og mulig konsekvens for hver risiko
- Sannsynlighet kan være svært lav, lav, moderat, høy eller svært høy
- Konsekvensen kan være katastrofal, alvorlig, mindre alvorlig eller ubetydelig

Risikotyper og eksempler fra Sommerville s. 600

Risiko	Sannsynlighet	Konsekvens
Det er umulig å rekruttere medarbeidere med kompetansen som er nødvendig	Høy	Katastrofal
Nøkkelpersonell er syke eller fraværende i kritiske faser av prosjektet	Moderat	Alvorlig
Det er foreslått endringer i kravspesifikasjonen som vil kreve store endringer i design av systemet	Moderat	Alvorlig
Organisasjonen restruktureres slik at ulike ledelse har ansvar for prosjektet	Høy	Alvorlig
Databasesystemet kan ikke prosessere antall transaksjoner per sekund som forventet	Moderat	Alvorlig

Ledelse av mennesker

- Mennesker er organisasjonens største ressurs
- Oppgavene til en leder er hovedsakelig orientert mot andre mennesker
- Mangelfull eller dårlig ledelse er ofte en av de viktigste årsakene til at et prosjekt feiler

Motivasjon

- Er ikke folk motiverte, er de lite interesserte i hva de gjør
- Motivasjon er komplekst, men ulike typer motivasjon er basert på
 - Basis behov (mat, søvn, etc.)
 - Personlige behov (respekt, selvtillit , etc.)
 - Sosiale behov (å bli akseptert i en gruppe, etc.)

Personlighetstyper

- Oppgaveorientert (Task-oriented)
 - Motivasjonen for å gjøre arbeidet er oppgavene i seg selv
- Selvorientert (Self-oriented)
 - Arbeidet er et middel for å oppnå individuelle mål – bli rik, få posisjon, reise, etc.
- Samspillorientert (Interaction-oriented)
 - Motivasjonen er først og fremst å samarbeide og ha det bra med medarbeiderne. Man går på jobb fordi man liker å gå på jobb

Hva er et team?

- “et lite antall folk med komplementære ferdigheter som har et felles formål, felles mål for ytelse, en felles tilnærming, og som holder hverandre gjensidig ansvarlige”
- Katzenbach, J.R., Smith, D.K.: The Discipline of Teams. Harvard Business Review 71, 111–120 (1993)

Hva er spesielt med Software-team?

- Hyppige endringer gjør det vanskelig å planlegge
- Komplekse sosiale og tekniske system
- Få etablerte teorier om systemutvikling

Teamarbeid

- Et team som fungerer er samlet og har en god teamfølelse. Teamets mål viktigere enn egne mål
- Kommunikasjon er en nøkkelfaktor for å lykkes
- Fleksibilitet i teamsammensetning er ofte begrenset av hvem som er tilgjengelig

Effektivitet i team

- Teammedlemmer
 - Trenger en blanding av folk fordi systemutvikling involverer ulike aktiviteter som programmering, testing, programvarearkitektur og dokumentasjon
- Teamorganisering
 - Et team bør organiseres slik at alle teammedlemmene kan bidra best mulig og at oppgaver blir utført som forventet
- Kommunikasjon
 - God kommunikasjon mellom teammedlemmene, og mellom teamet og andre interessenter (stakeholders), er helt essensielt



DID
NOT
WORK
WELL
IN
GROUPS

SIEMONE

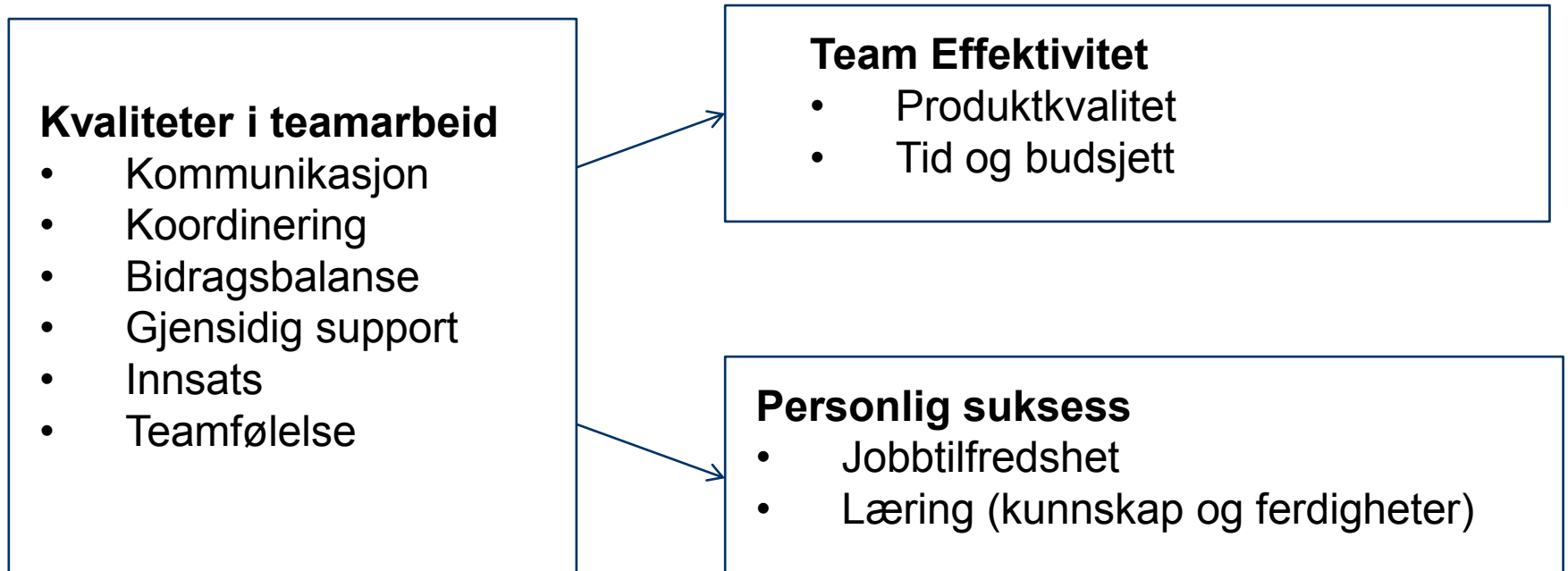


Forskningsprosjekt innen teamarbeid

- Hva er effektivt teamarbeid i “smidige” prosjekter?
 - Måle effektiviteten innad i team og på tvers av team
 - Distribuerte team
- Noen dimensjoner ved teamarbeid som vi undersøker
 - Selvstyre i team
 - Kommunikasjon i team
 - Teamfølelse
 - Læring i team
 - Gjensidig support og oppgavefordeling

Stor spørreundersøkelse – 71 smidige team

Modell



Eksempler – spørsmål kommunikasjon

Spørsmål Teamarbeid - kommunikasjon	Sterkt uenig	Uenig	Verken enig eller uenig	Enig	Sterkt enig
Det er hyppig kommunikasjon innad i teamet					
Teammedlemmene kommuniserer hyppig i spontane møter, telefon etc.					
Teammedlemmene kommuniserer for det meste direkte og personlig med hverandre					
Mye av kommunikasjonen i teamet går gjennom sentrale personer (teamleder eller visse teammedlemmer)					
Relevante ideer og informasjon om teamarbeidet deles åpent blant alle teammedlemmene					
I noen situasjoner blir viktig, relevant informasjon holdt tilbake fra andre teammedlemmer					
I teamet er det konflikter om åpenheten rundt informasjonsflyten					
Teammedlemmene synes fullt ut at de får informasjon fra andre teammedlemmer i tide					
Teammedlemmene synes fullt ut at de får nøyaktig nok informasjon fra de andre teammedlemmene					

Eksempler – spørsmål effektivitet

Spørsmål teamarbeid	Sterkt uenig	Uenig	Verken enig eller uenig	Enig	Sterkt enig
Kunden er fornøyd med kvaliteten på resultatet av teamarbeidet					
Teamet er fornøyd med resultatet av arbeidet sitt					
Produktet som utvikles i teamet, krever lite omarbeiding					
Produktet er stabilt i bruk					
Produktet er robust i bruk					
Bedriften er fornøyd med progresjonen i teamarbeidet					
I det store og hele jobber teamet på en kostnadseffektiv måte					
I det store og hele jobber teamet på en tidseffektiv måte					
Teamet holder seg innenfor tidsskjema					
Teamet holder seg innenfor budsjett					

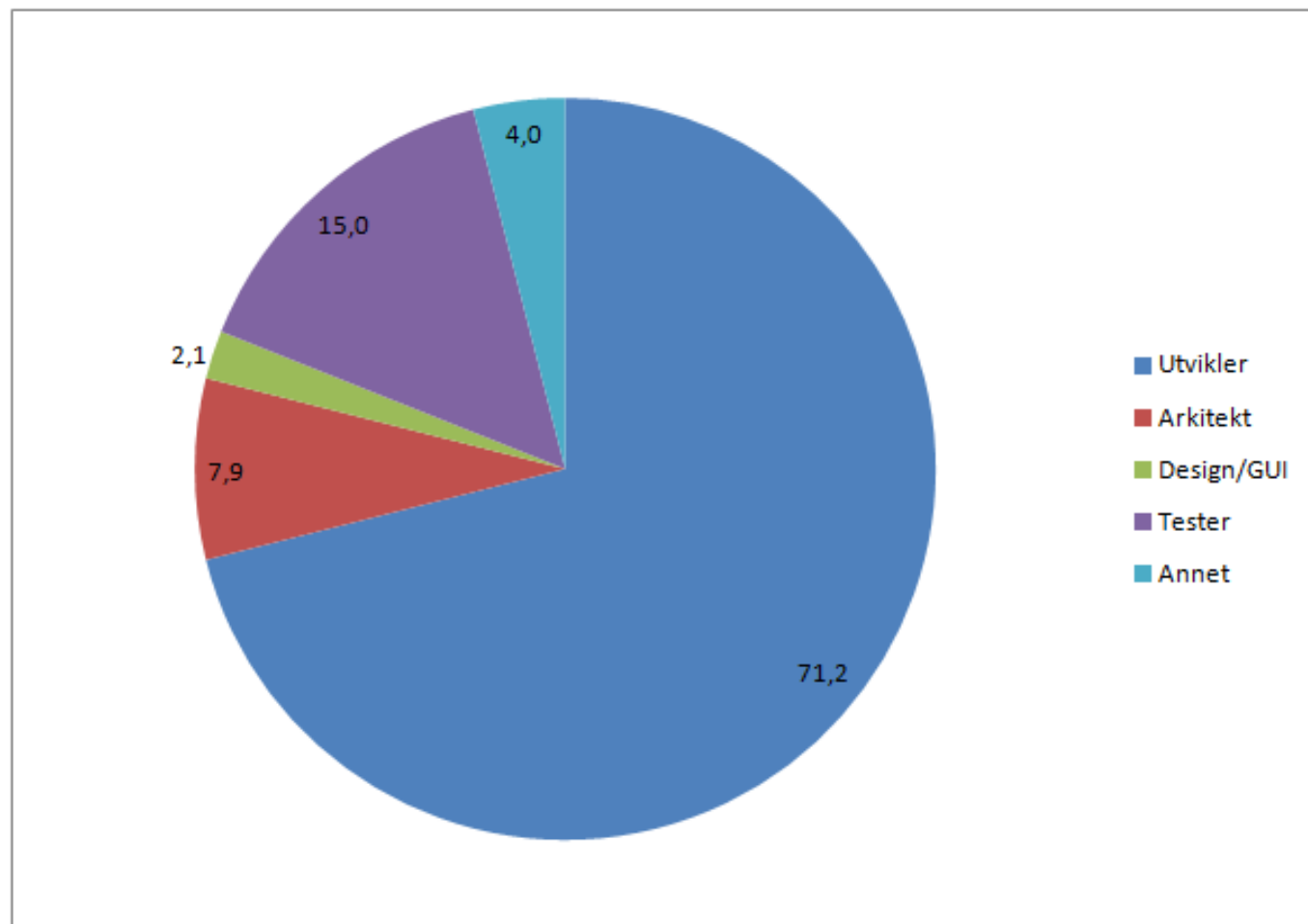
Eksempler – spørsmål personlig suksess

Spørsmål teamarbeid – personlig suksess	Sterkt uenig	Uenig	Verken enig eller uenig	Enig	Sterkt enig
Så langt kan teamet være fornøyd med seg selv					
Teammedlemmene drar nytte av team-samarbeidet					
Teammedlemmene vil gjerne gjøre lignende samarbeid igjen					
Vi tilegner oss viktig kunnskap gjennom teamarbeidet					
Vi anser teamarbeidet som faglig vellykket					
Teamet trekker viktige lærdommer av teamarbeidet					
Teamarbeid fremmer personlig vekst					
Teamarbeid fremmer profesjonell vekst					

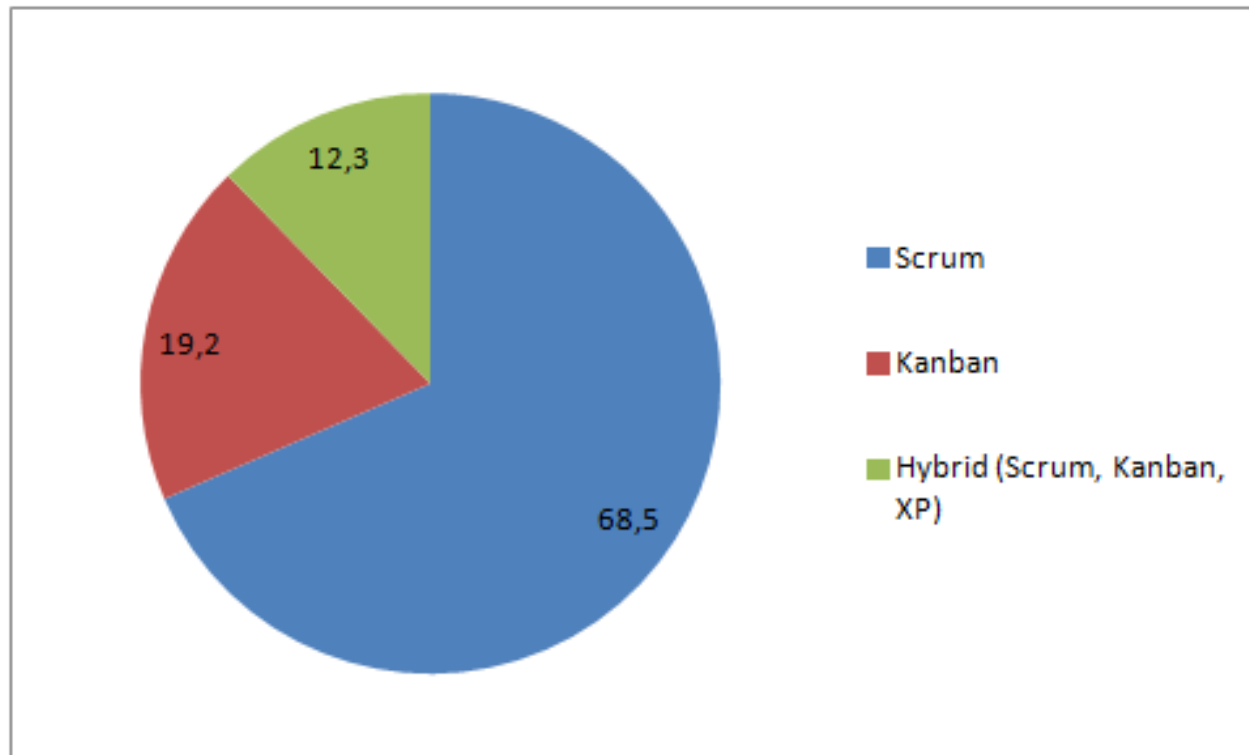
Personer fra 71 team

Rolle	Sum	Gjennomsnitt	Min	Max
Team leder	76	1,1	1	3
Produkteier	78	1,1	1	4
Team medlem	323	4,5	1	10
Total	477	6,7	3	13

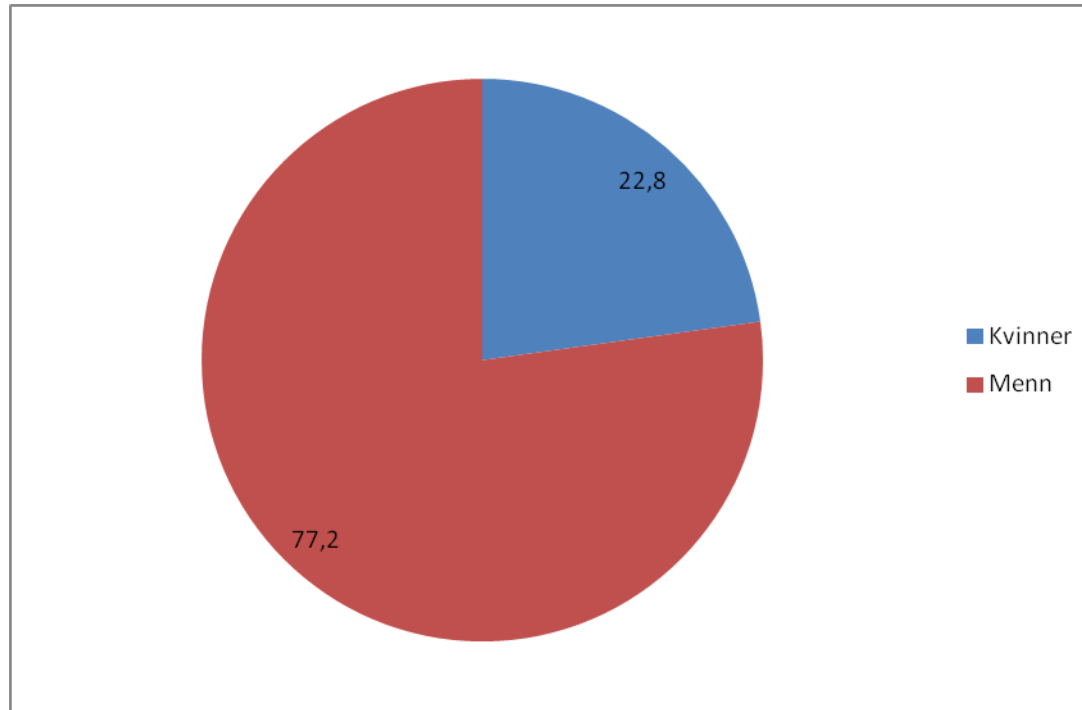
Teammedlemmer – primærrolle i team



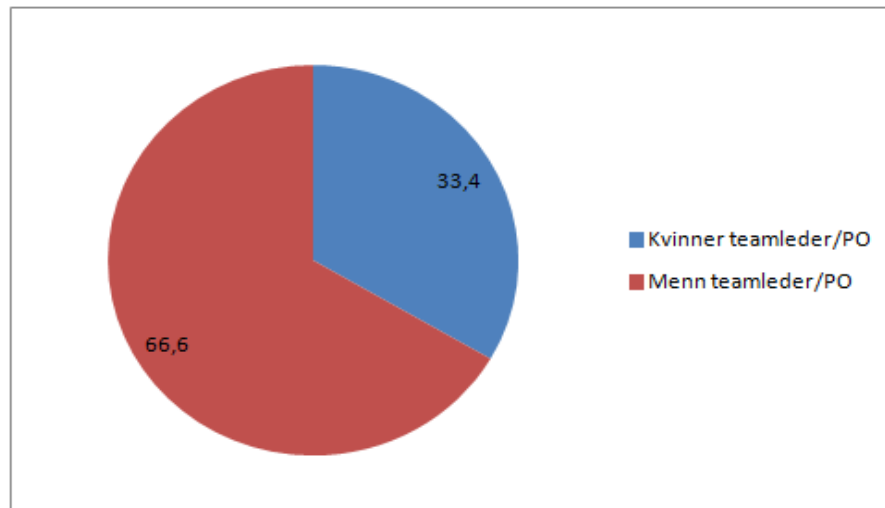
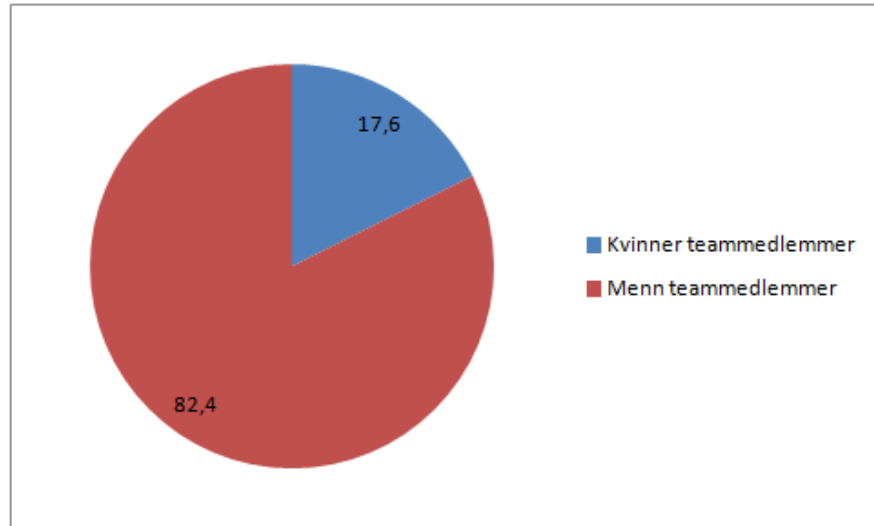
Type "smidig" team



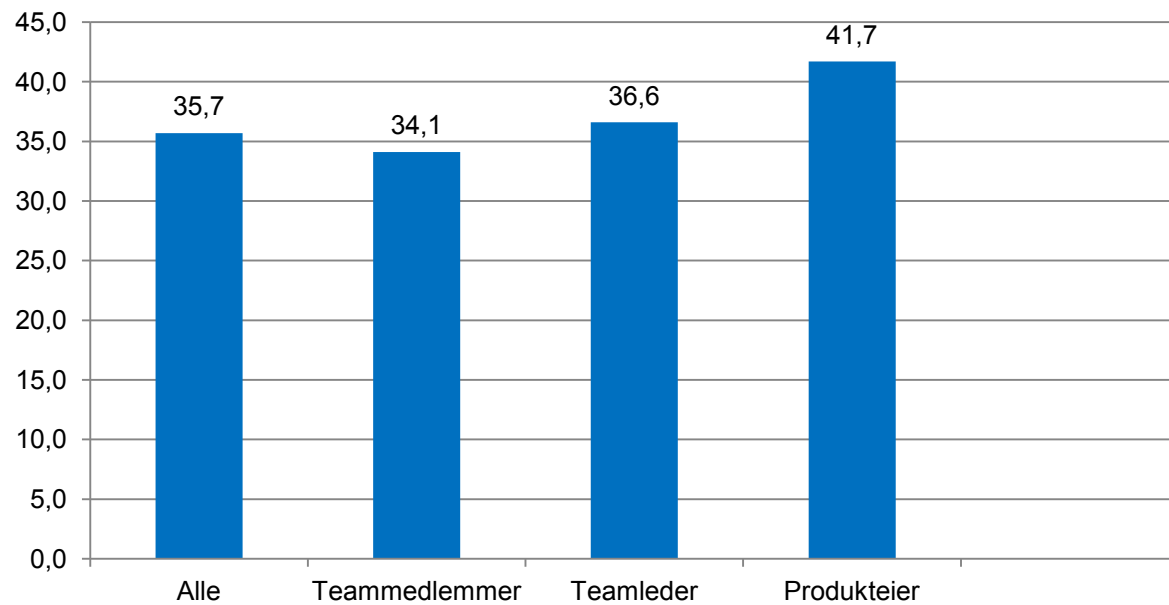
Kvinner/menn i teamene



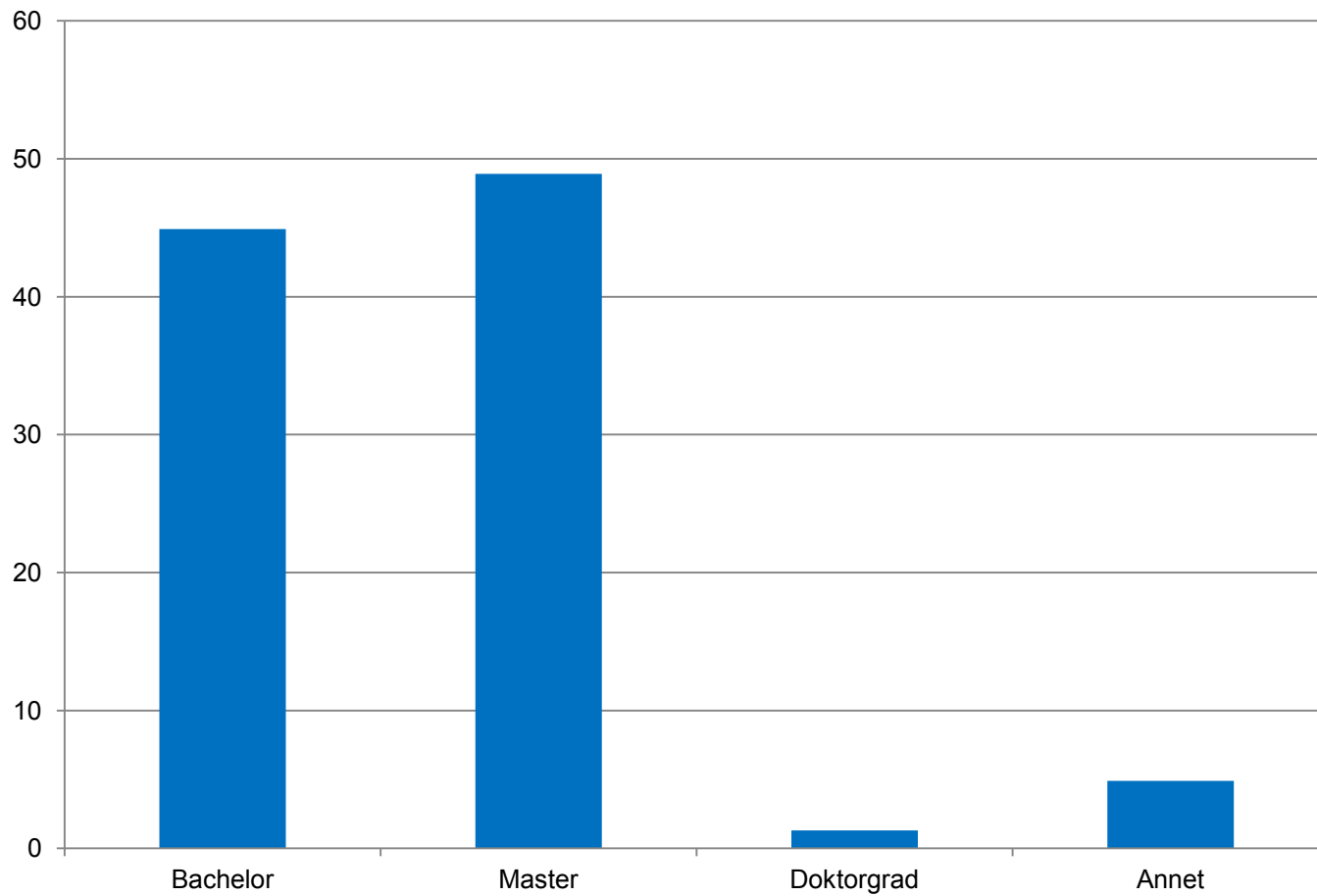
Kvinner/menn i teamene – teammedlemmer vs teamleder og produkteier



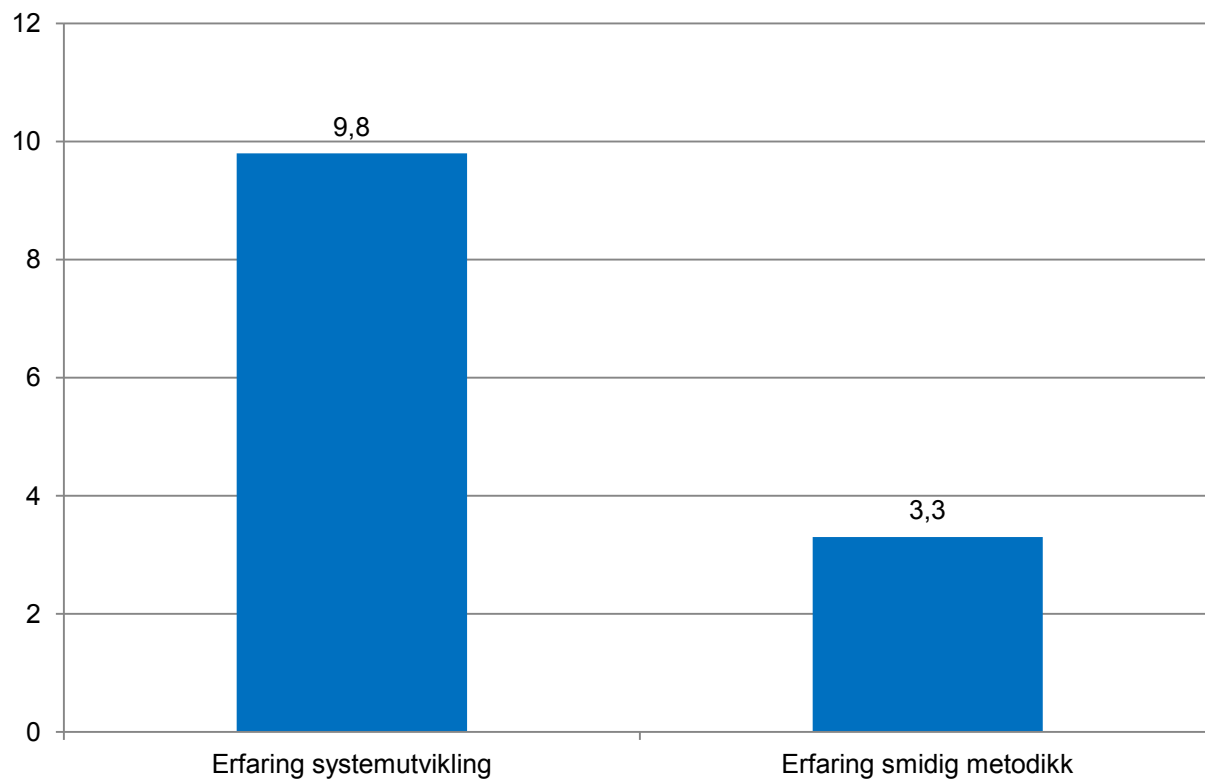
Alder



Utdanning



Erfering med systemutvikling – antall år



Uformelle team

- Teamet opptrer som en helhet og kommer til enighet i forhold til avgjørelser som gjelder systemet som utvikles
- Teamleder er en fasilitator til teamet og andre deler av prosjektet, men har ikke ansvar for å fordele spesifikke arbeidsoppgaver
- Arbeidet blir diskutert i teamet som helhet og oppgaver blir fordelt i forhold til evne, erfaring og tilgjengelighet

Kommunikasjon i team

- Teamstørrelse
 - Jo større gruppen er, dess vanskeligere med kommunikasjon
- Teamstruktur
 - Kommunikasjon går bedre i uformelle team enn i hierarkisk strukturerte team
- Teamsammensetning
 - Ulike personlighetstyper og kjønn fungerer best
- Arbeidsomgivelser
 - God organisering av arbeidsplassen kan bidra til økt kommunikasjon

Prosjektplanlegging

- Prosjektplanlegging handler om
 - å dele arbeidet i mindre deler og aktiviteter
 - tilordne delaktiviteter til teammedlemmene i prosjektet
 - forutse problemer som kan dukke opp
 - forbrede løsninger på problemene som kan dukke opp
- Prosjektplan utarbeides i starten av prosjektet og blir brukt til å kommunisere hvordan arbeidet blir utført
 - til prosjekttemaet
 - til kundene og andre interessenter

Faktorer som påvirker pris - fra Sommerville s. 622

Faktor	Beskrivelse
Muligheter i markedet	Et selskap i etableringsfasen kan gi en lav pris for å komme inn i markedet.
Usikkerhet i kostnadsestimeringen	Usikkerhet i estimering av kostnad kan føre til økt pris; legger til for "uforutsette" hendelser.
Kontraktsmessige forhold	En kunde kan gi leverandøren eierskap til kildekoden som dermed kan benyttes til andre prosjekter, mot at de betaler mindre for produktet.
Hyppig endringer i kravspesifikasjonen	For å vinne kontrakten er det vanlig å prise lavt hvis man "vet" at det vil komme endringer i kravspesifikasjonen. Så kan endringer i kravspesifikasjonen prises høyt etterpå.
Finansiell styrke	Utviklere i økonomiske vanskeligheter kan gi lav pris for å vinne kontrakten. Bedre med "break even" enn ingenting.

Plandrevet utvikling

- Utviklingsprosessen planlegges i detalj
 - Basert på tradisjonelle prosjektstyringsteknikker og har vært den mest vanlige måten å styre store programvareprosjekter på
- Argumenter for en plandrevet tilnærming er at tidlig planlegging sørger for at organisatoriske forhold, som tilgjengelige ressurser blir tatt hensyn til, og at potensielle problemer og avhengigheter blir oppdaget før prosjektet starter i stedet for midt i prosjektet
- Hovedargumentet mot plandrevet utvikling er at mange tidlige avgjørelser likevel må endres på grunn av endringer i omgivelsene der programvaren utvikles og skal brukes

Prosjektplan

- I et plandrevet utviklingsprosjekt vil en prosjektplan blant annet si noe om tilgjengelige ressurser i prosjektet, arbeidsfordelingen blant ressursene og en tidsplan for når arbeidet skal gjøres.
- Planen inneholder gjerne følgende:
 - Innledning
 - Prosjektorganisering
 - Risikoanalyse
 - Maskinvare- og programvarekrav (standard programvare som skal brukes)
 - Arbeidsfordeling
 - Tidsplan for prosjektet
 - Rapporterings- og overvåkingssystemer

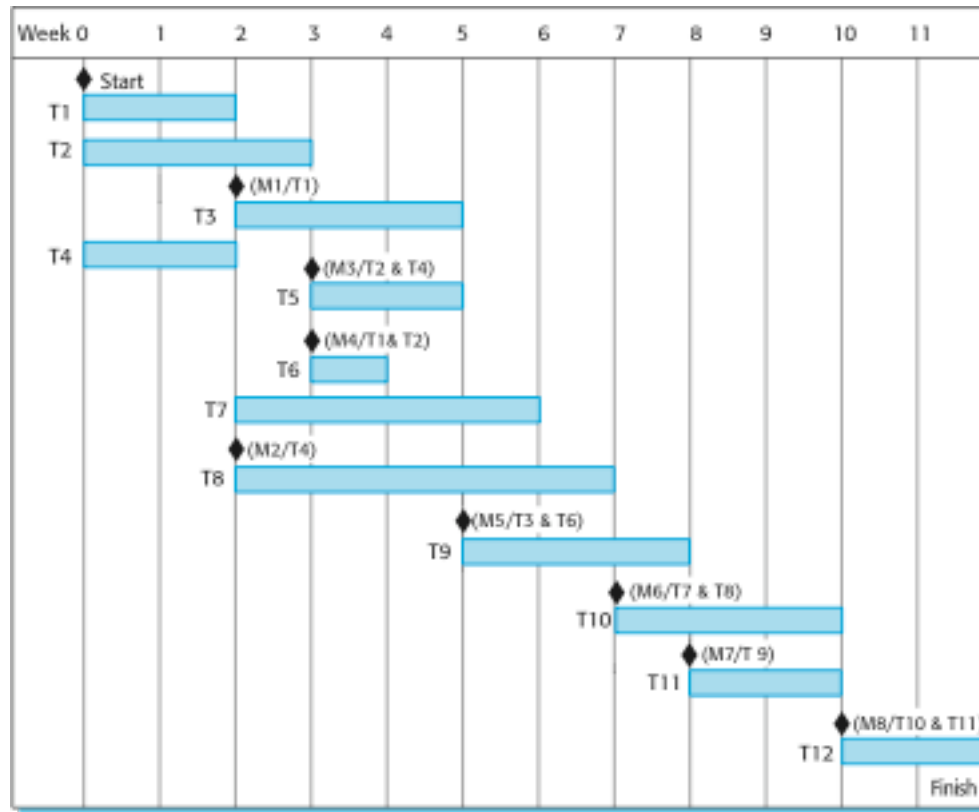
Milepæler og leveranser

- Milepæler er punkter i timeplanen (schedule) som måles mot fremskritt (progress) i prosjektet. En milepæl kan for eksempel være at systemet er klar til test
- Leveranser er produkter som leveres kunden (eller andre interessenter), for eksempel kravdokumentet for systemet

Oppgaver, varighet og avhengighet – eksempel fra Sommerville

Oppgave	Arbeid (person-dager)	Varighet(dager)	Avhengighet
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)

Aktivitetetsdiagram (stolpediagram – bar chart)



Smidig planlegging

- Smidige metoder i programvareutvikling er en iterativ tilnærming der programvaren blir utviklet og levert til kundene som "tillegg" ("increments")
- Til forskjell for plandrevet tilnærming, er funksjonaliteten til tilleggene (increments) ikke planlagt på forhånd men avgjøres under utviklingen
 - Hva som tas med i en iterasjon avhenger av fremskritt (progress) i prosjektet og kundens prioriteringer
- Kundens prioriteringer og krav endrer seg. Derfor kan det være fornuftig å ha en fleksibel plan som kan ta høyde for disse endringene

Planleggingsfaser i smidig utvikling

- "Release" planlegging, med et tidsperspektiv på flere måneder og som avgjør hva som skal være med i en "release" av systemet
- Iterasjonsplanlegging, med en kortere tidshorisont, og som fokuserer på å planlegge neste "increment" av systemet. Dette er typisk 2-4 uker for hver iterasjon

Undersøkelse av 71 smidige team

- Release-intervall – 4,4 måneder
- Iterasjonsintervall – 2,8 uker

Historiebasert ("Story-based") planlegging

- System spesifikasjonen i XP (og SCRUM) er basert på brukerhistorier (user stories) som reflekterer egenskapene i systemet
 - "Som student, ønsker jeg å melde meg opp i kurs"
- Teamet diskuterer historier og rangerer dem i forhold til tiden de tror det tar
- Historier som skal være med i en "iterasjon" velges, der antall historier reflekterer tiden det tar å levere en iterasjon (typisk 2-4 uker)

Les mer om user stories og forskjellen til use cases (sentralt i UML) på http://en.wikipedia.org/wiki/User_story

Smidig utvikling – 4 overgripende utsagn

- **Personer og samspill** fremfor prosesser og verktøy
- **Programvare som virker** fremfor omfattende dokumentasjon
- **Samarbeid med kunden** fremfor kontraktsforhandlinger
- **Å reagere på endringer** fremfor å følge en plan

Scrum – tre faser

- **Planleggingsfasen:** overordnede mål for prosjektet etableres og programvarearkitekturen designes
- **Gjennomføringsfasen:** en serie med *sprint-iterasjoner*, der hver iterasjon leverer et inkrement av systemet
- **Avslutningsfasen:** nødvendig dokumentasjon som hjelp-funksjoner og brukermanualer fullføres, og man oppsummerer hva man har lært i prosjektet

Teamarbeid i Scrum

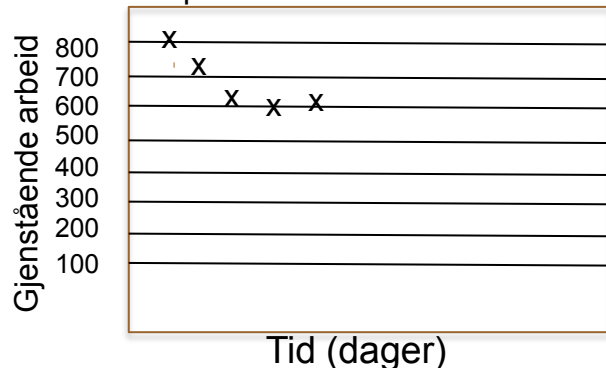
- ‘Scrum master’ er en tilrettelegger som
 - arrangerer daglige møter
 - holder orden på (back)loggen av arbeid som skal gjøres i hver iterasjon
 - noterer beslutninger
 - måler progresjon i forhold til loggen
- Hele teamet deltar i korte, daglige møter der alle
 - deler informasjon
 - beskriver progresjon siden forrige møte
 - diskuterer problemer som har oppstått
 - planlegger neste dag
- Alle i teamet vet hva som skjer. Oppstår det problemer, kan man endre planer for å kunne håndtere dem

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?

Sprint X Burn Down Chart



Scrum Master

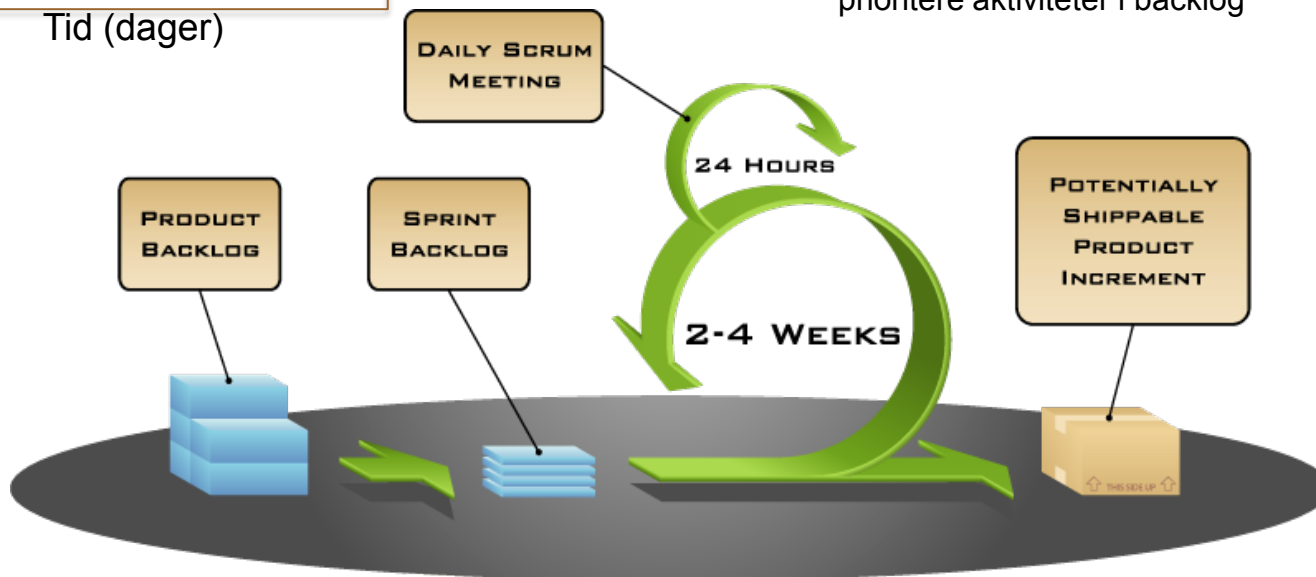
- Sørger for at Scrum prosessen blir fulgt
- Leder "daily stand up"
- Hindrer støy slik at utviklerne kan fokusere på oppgavene
- Teamets "coach og beste venn"
- Koordinerer SPRINT planlegging, og estimering av Sprint backlog

Scrum Team

- Typisk 5-9 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





Nexte - 1. system

Analyse & Design

Umbau

Test - intern

Abnahme - extern

1 p