

Smidig utviklingsmiljø: Organisatorisk læring og kommunikasjon

Case studier av retrospektivmøter

Aravinthan Yogarajah

Master i informatikk

Innlevert: juni 2016

Hovedveileder: Eric Monteiro, IDI

Norges teknisk-naturvitenskapelige universitet
Institutt for datateknikk og informasjonsvitenskap

Sammendrag

Smidig utviklingsmetodikk har blitt bredt brukt blant programvareselskapene i både samlokalisert og distribuert team. I slike situasjoner forekommer det at prosjektene ikke leverer produktet til tide, der manglende læring og kommunikasjonsutfordringer kan være grunnen. Derfor undersøkes utfordringene systemutviklings-team har i forhold til organisatorisk læring og kommunikasjon i både samlokalisert og distribuert team. Det er gjennomført ti observasjoner basert på tre case studier.

Resultatene mine viser at systemutviklingsteam gjennomfører retrospektivmøter og tar læring av dette, men læringen er begrenset. Det er både identifisert enkelt og dobbeltløkkelæringer i retrospektivmøtene, men det er primært enkeltløkkelæringer som forekommer og dette henger sammen med eksisterende teori. Den største hindringen for å danne dobbeltløkkelæring har vært gjennomføring av forbedringstiltakene. Manglende gjennomføringsevne skyldes at det fokuseres på mange forbedringstiltak, manglende oppfølging og tilrettelegging av forbedringstiltakene, manglende konkrete forbedringstiltak og involvering av eksterne aktører. Når forbedringstiltakene ikke gjennomføres fører det til at de samme utfordringene diskuteres gjentatte ganger. En konsekvens av dette har ført til misnøye for gjennomføring av retrospektivmøtene. Jeg har utarbeidet en liste med faktorer som har bidratt til å danne dobbeltløkkelæring. Basert på denne listen har jeg foreslått en modell for gjennomføring av retrospektivmøtene. Når det gjelder kommunikasjon viser det seg at mangel på ansikt-til-ansikt kommunikasjon i distribuerte team har ført til vanskeligheter med å forstå informasjonen som blir formidlet korrekt i motsetning til samlokalisert team.

Tilslutt konkluder jeg med at manglende gjennomføring av forbedringstiltakene vil påvirke produktiviteten til teamet. Manglende ansikt-til-ansikt kommunikasjon i distribuert team har ført til redusert forståelse rundt informasjonen som er formidlet mellom distribuerte teammedlemmer og henger sammen de fleste studiene på dette forskningsfeltet.

Abstract

Agile development methodology has been widely used among software companies in both co-located and distributed teams. In such situations it happens that the projects do not deliver the product to the time, where the lack of learning and communication challenges may be the reasons. That's why I will investigate the challenges software development teams have in relation to organizational learning and communications challenges in co-located and distributed teams. It is conducted ten observations based on three case studies.

My results shows that software development teams conducts the retrospective meetings and learns from it, but the learning is limited. There are identified single and double loop learning in retrospective meetings but the primary learning type is single loop learning which supported the existing theory. The main obstacle to from double loop learning has been the implementation of improvement measures. The lack of implementation of improvement measure are becasue the team focus on many improvement measures, lack of follow-up and facilitating the improvement measures, lack of specified improvement measures and the involvement of external actors. When improvement measures are not implemented, it will causes to discuss the same challenges repeatedly. A consequence of this has led to dissatisfaction for conducting the retrospective meeting. The results about communication shows the lack of face-to-face communication in distributed teams has led to difficulty in understanding the information being conveyed as opposed to co-located teams.

Finally, I conclude that the lack of implementation of the improvement measures will affect the productivity to the team due lack of learning. The lack of face-to-face communication in distributed team has reduced the understanding about the information conveyed between distributed team members which support existing main field in research.

Forord

Denne masteroppgaven ble utarbeidet som en avslutning på det to-årig studiet i master i informatikk ved Norges teknisk-naturligvitenskaplige universitet. Masteroppgaven fokuserer på organisatorisk læring og kommunikasjon i smidig utviklingsmiljø med utgangspunkt i retrospektivmøter.

Arbeidet med masteroppgaven er relatert til agile 2.0 prosjektet som er finansiert av forskningsrådet i Norge gjennom bevilgning 236759, og støttes av bedriftene Kantega, Kongsberg Defence & Aerospace, Sopera Steria og Sticos.

Jeg vil takke hovedveileder, Eric Monteiro for veiledningen som jeg har fått for å skrive masteroppgaven. Jeg fikk også veiledning fra Torgeir Dingsøyr spesielt i forhold til retrospektivmøter og organisatorisk læring. Begge har gitt meg tilgang til casene som er brukt i studiet. Takk til Nils Brende Moe og Torgeir Dingsøyr for å ha sett gjennom kravene mine for bruk av verktøy i retrospektivmøter.

Videre ønsker jeg å takke alle bedriftene og samtlige deltakere som deltok i dette studiet. Samtidig ønsker å takke studentgruppa som jeg har møtt ukentlig gjennom studieåret og gitt meg gode råd og tips som har vært nyttig for å gjennomføre masteroppgaven. Deriblant medstudenten, Tina Christin Syversen for å gitt meg tilgang til datamaterialet hun har samlet inn i forhold til et av casene i dette studiet.

Innhold

1	Introduksjon	1
1.1	Motivasjon	1
1.2	Forskningsspørsmål	2
1.3	Framgangsmåte og avgrensning	2
1.4	Oppbygningen av oppgaven	3
2	Systemutviklingens historie: Fossefall til smidig utvikling	4
3	Retrospektiv	10
3.1	Definisjoner på retrospektiv	11
3.2	Gjennomføring av retrospektivmøte	12
3.2.1	Derby og Larsens modell	12
3.2.2	Dingsøyrs vurderingsfaktorer	15
3.3	Empiriske studier og modeller for retrospektivmøter	18
3.4	Modell basert på eksisterende litteratur	20
3.5	Organisatorisk læring	23
3.5.1	Enkeltløkkelæring	24
3.5.2	Dobbeltløkkelæring	25
3.5.3	Bruksteori	26
3.5.4	Model I	26
3.5.5	Modell II	27
4	Distribuert og samlokalisert team	31
4.1	Kommunikasjon	32
4.1.1	Innholdsrik kommunikasjon	33
4.1.2	Formaliteten på kommunikasjon og fysisk avstand	34
4.1.3	Bevissthet og kommunikasjon	36
4.1.4	Historikk og fortolkning	36
4.1.5	Kommunikasjon og koordinering	37
4.1.6	Kommunikasjon i smidig utviklingsmetodikk	37
4.2	Ressursbruk	38

5	Metode	39
5.1	Valg av forskningsmetode	39
5.2	Forskningsdesign	40
5.3	Bakgrunn for casestudiet	41
5.3.1	Bedriftene	41
5.3.2	Casene	41
5.4	Datainnsamling	44
5.4.1	Observasjon	44
5.4.2	Ustrukturerte intervjuer	45
5.4.3	Framdrift og omfanget av datainnsamlingen	46
5.5	Datanalyse	47
5.5.1	Tematisk analyse	47
5.5.2	Dataanalyse for organisatorisk læring	51
5.6	Refleksjon	54
6	Resultater	56
6.1	Gjennomføring av retrospektiv	57
6.1.1	Trinn 1: Forberedelse	57
6.1.2	Trinn 2: Introduksjon	58
6.1.3	Trinn 3: Datainnsamling	60
6.1.4	Trinn 4: Diskusjon og utarbeide forbedringstiltak	65
6.1.5	Trinn 5: Planlegging for gjennomføring av forbedringstiltak	69
6.1.6	Trinn 6: Avslutt retrospektiv	70
6.2	Utbytte av retrospektivmøter	71
6.2.1	Gjennomføring av forbedringstiltak	71
6.2.2	Enkeltløkke-og dobbeltløkkelæring	72
7	Diskusjon	76
7.1	Organisatorisk læring	77
7.1.1	Model I	77
7.1.2	Model II	79
7.1.3	Refleksjon rundt læringsprosessen	84
7.1.4	Bakgrunn for dobbeltløkkelæring	88
7.1.5	Min modell for gjennomføring av retrospektivmøter	90
7.2	Kommunikasjon	94
7.2.1	Innholdsrik kommunikasjon	94
7.2.2	Uformell kommunikasjon og fysisk nærhet	95
7.2.3	Oppsummering	99
8	Konklusjon og videre arbeid	100
8.1	Konklusjon	100
8.1.1	Organisatorisk læring	100
8.1.2	Kommunikasjon	101
8.2	Forslag til videre arbeid	102
	Referanseliste	109

Vedlegg

A	Verktøy for distribuert retrospektiv	111
A.1	Fase 1: Resultat for vurdering av verktøy	113
A.1.1	Beskrivelse av kategoriene	114
A.1.2	Argumentasjonen for kategoriene	115
A.2	Fase 2: Resultat for vurdering av verktøy	117
A.3	Nærmere beskrivelse av verktøyet ”Retrium”	119
A.4	Case D2: Bruk av verktøyet Retrium	122
B	Scrum rammeverket	129
B.1	Sprint	129
B.2	Roller	129
B.3	Møter	130
B.4	Artefakter	132
C	Beskrivelse av temaene i retrospektivmøter	133

Tabeller

3.1	Ulike typer for retrospektivmøter	10
3.2	Oppsummering av Derby og Larsens modell	14
3.3	Oppsummering av Dingsøyrs vurderingsfaktorer	17
3.4	Modell basert på eksisterende litteratur	21
3.5	Model I bruksteori [1]	29
3.6	Model II bruksteori [1]	30
4.1	Beskriver hvordan geografisk avstand og tidsforskjeller påvirker team- sammensetningen, basert på informasjon fra [2].	31
4.2	Beskriver og sammenligner formell og uformell kommunikasjon [3].	34
5.1	En sammenligning av casene i studiet	43
5.2	Lyddoptak av retrospektivmøtene	45
5.3	Omfanget av datainnsamlingen	46
5.4	Forkortelse for rollene	47
6.1	Rollene som har deltatt i møtene	57
6.2	Hendelsenforløpet i trinn 2, bortsett fra ved bruk av verktøy	60
6.3	Hendelsesforløpet i trinn 3	65
6.4	Fokusert retrospektivmøter	65
6.5	Diskusjonsstrukturen i retrospektivmøtene i case S	66
6.6	Statusen på forbedringstiltakene ved studie slutt i case S.	72
6.7	Antall enkeltløkke-og dobbeltløkkelæring i case S og D1 i løpet av studiet.	72
7.1	Oppsummering av diskusjonen i forhold til modell I.	79
7.2	Oppsummering av diskusjonen i forhold til modell II.	83
7.3	Bakgrunnen for at forbedringstiltak ikke gjennomføres	86
7.4	En oversikt over hvilke kommunikasjonsutfordringer distribuerte team har i forhold til samlokalisert team.	99
A.1	En beskrivelse av kategoriene som brukes i figur A.1	114
A.2	Vurdering av 7 beste verktøyene i de tre hovedkategoriene	117
A.3	Vurdering av 11 dårligste verktøyene i de tre hovedkategoriene	118
A.4	Vurdering av verktøyet Retrium i forhold til kategoriene på tabell A.1	120

A.5	Vudering av verktøyet Retrium i forhold til kategoriene på tabell A.1	121
C.1	En beskrivelse av temaene som er brukt i figur 6.5	133

Figurer

2.1	Den originale fosessfallmodellen [4]	4
2.2	Hvordan kravspesifikasjonene kan bli oppfattet av ulike roller	6
2.3	Spiralmodellen [5]	7
2.4	Den evolusjonære utviklingen av utviklingsmetodikk i systemutvikling fra 1983 fram til 2003. [6]	9
3.1	Iterasjonsretrospektiv i praksis [7]	12
3.2	Viser hvordan enkeltløkkelæring fungerer slik Argyis og Schön definerer det [1]	24
3.3	Viser hvordan dobbeltløkkelæring fungerer slik Argyis og Schön definerer det [1]	25
4.1	Beskriver hvor innholdsrik kommunikasjonsformen er, der ”cold” viser at innholdet er fattig i motsetning til ”hot” som viser at innholdet er rikt samt hvor effektiv kommunikasjonen er. [8]	33
4.2	Kommunikasjonsmuligheter avhengig av tid og sted [8]	35
5.1	Tidsplanen til datainnsamlingen	46
5.2	Konteksten på en post-it lapp	47
5.3	Etter å skrevet post-it lapper for alle observasjonsnotatene	48
5.4	Etter å ha identifisert 15 forskjellige kategorier	49
5.5	En oversikt over analyseprosessen for temaet koordinering (kommunikasjon)	50
5.6	Analyseprosessen	51
5.7	Analyseprosessen for styringsvariabelen gyldig informasjon i modell II	53
6.1	Hvordan deltakerne sitter i forhold til hverandre	58
6.2	Vurdering av tidligere forbedringstiltak	59
6.3	Gjennomføring av fornøydhetsskurven	59
6.4	Hvordan votering og gruppering av positive lapper vanligvis foregår i case S.	64
6.5	Oversikt over de mest diskuterte temaene i løpet av studiet.	66
6.6	Hvor ofte forbedringstiltak tildeles til et team eller en spesifikk deltaker	70
6.7	Statusen på forbedringstiltakene ved studieslutt.	71
6.8	Antall enkeltløkke- og dobbeltløkkelæring i case S og case D1.	73

7.1	Revideringene fra eksisterende modell til ny modell	90
A.1	Resultatet av fase 1 for vurdering av verktøy	113
A.2	Deltakeren har selv ikke skrevet noe, og innholdet i andres deltakerens lapper er skjult.	122
A.3	Lappene er synlige etter at alle har skrevet ferdig lappene sine.	124
A.4	Gruppering av positive og forbedringslapper	125
A.5	Votering av de grupperte lappene	126
A.6	Utarbeider forbedringstiltak på temaene	127
A.7	Viser oversikt over forbedringstiltakene fra gjennomført retrospektivmøte.	128

Kapittel 1

Introduksjon

I dette kapittelet beskrives introduksjonen til masteroppgaven. Først beskrives motivasjonen, deretter forskningsspørsmålet, framgangsmåten og avsluttes med oppbygningen av masteroppgaven.

1.1 Motivasjon

Smidig utviklingsmetodikk har i løpet av de siste tiårene blitt utbredt blant programvarebedriftene. Denne metodikken brukes til å utvikle komplekse programvareprodukter som gjerne må ferdigstilles innenfor en tidsramme [9]. Likevel er det prosjekter som ikke leveres innenfor tidsrammen. En av årsakene er manglende fokus på læring av tidligere feil som ført til at prosjektene ble forsinket eller ble mislyktes [10, 11]. En annen årsak er kommunikasjonsutfordringene i distribuerte team, der det er vanskeligere å formidle og forstå informasjon fra lengre fysisk avstand i forhold til å være samlokaliserte [12, 13].

Smidig metoder baserer seg på oftere levering og tilbakemelding fra kunden som gjennomføres i korte iterasjoner. I iterasjonen har vi en praksis som kalles ”retrospektiv” som er et møte der deltakerne i et team reflekterer over hendelsene fra siste iterasjonen og som inkluderer læringen av tidligere feil. Ved å gjennomføre retrospektivmøtet skal deltakerne være i stand til å øke produktiviteten sin. I praksis kan dette gjøres ved å utarbeide forbedringstiltak for å løse både personlige og tekniske utfordringer i teamet [7]. Det er utformet og beskrevet flere metoder for hvordan denne praksisen skal gjennomføres og hensikten bak praksisen [7, 14, 15]. Derfor kan det tenkes at bruken av retrospektivpraksisen har økt fra fra 39 % i 2007 til 74 % i 2015 [16, 17] i følge en webbasert uformell undersøkelse. Etter min beste evne har jeg ikke sett studier som adresserer om deltakerne faktisk lærer i retrospektivmøter. Derfor er dette hovedfokuset for gjennomføringen av forskningen.

Det er stadig flere bedrifter som har valgt å bruke distribuerte team som tidligere var samlokaliserte. Den største utfordringen for distribuerte team har vært kommunikasjonsutfordringene i forhold til samlokaliserte team i et smidig utviklingsmiljø [18]. Denne utfordringen har vært grunnlaget for andre utfordringer som samarbeid og koordinering innad i teamet [12]. Det er ikke like enkelt å kunne kommunisere med et teammedlem som oppholder seg på et annet fysisk i forhold til å være på samme fysisk sted, da det er vanskeligere å forstå og formidle informasjon [19]. Disse utfordringene skal vi se nærmere på og hvilke konsekvenser dette kan gi for effektiviteten for arbeidet som gjennomføres.

1.2 Forskningsspørsmål

Det overordnet forskningsspørsmålet omfatter hovedtemaene som er presentert ovenfor, organisatorisk læring og kommunikasjon. Disse temaene undersøkes nærmere i smidig utviklingsmiljø. Derfor har jeg har definert følgende overordnet problemstilling:

Å undersøke utfordringer knyttet til organisatorisk læring og kommunikasjon i smidig utviklingsmiljø med utgangspunkt i retrospektivmøter.

Det er vanskelig å kunne svare på dette spørsmålet, da det er såpass stort og vidt. Derfor har jeg delt problemstillingen i to delspørsmål:

1. *Hvilke utfordringer har systemutviklingsteam i forhold til å lære gjennom retrospektivmøter?*
2. *Hvilke utfordringer har distribuerte team sammenlignet med samlokalisert team når det gjelder kommunikasjon?*

Det presiseres at det primært fokuseres på det første forskningsspørsmålet som handler om organisatorisk læring. Dette betyr at det andre forskningsspørsmålet vil være ett supplement.

1.3 Framgangsmåte og avgrensning

Det ble brukt eksisterende prosjekter for å finne svaret for spørsmålene. Tre case studier ble gjennomført fra to forskjellige bedrifter. Der jeg gjennomførte kvalitativ datainnsamling, nærmere bestemt observasjon av retrospektivmøtene i disse tre casestudiene samt bruk av ustrukturerte intervjuer. Disse tre casene brukes i ulik grad for å kunne svare på forskningsspørsmålene.

Det kan nevnes at to av tre caser brukes aktivt i denne oppgaven. Da den siste casen (D2) kun fungerer som en støtte for å kunne svare på forskningsspørsmålet om utfordringer knyttet til kommunikasjon i distribuerte team. Bakgrunnen for dette er beskrevet i kapittel 5.2.

For å kunne svare på det første forskningsspørsmålet brukes det to caser, der den ene er samlokalisert og den andre er distribuert. Det ble gjennomført analyse for å undersøke i læringseffekten i retrospektivmøter og det skiltes ikke mellom samlokalisert og distribuert team. Resultatet av denne oppgaven er en beskrivelse av faktorer som har påvirket læringen i retrospektivmøter og det er utarbeidet forslag til forbedringer. Jeg håper disse forslagene hjelper flere bedrifter med å lære gjennom retrospektivmøter i større grad i forhold til i dag.

1.4 Oppbygningen av oppgaven

Masteroppgaven består av åtte kapitler. Disse er gjengitt nedenfor.

Første kapittel introduserer teamene og beskriver problemstillingen og motivasjonen bak masteroppgaven.

Andre kapittel gir en kort innføring i systemutviklingshistorie, fra fossefallsmodellen på 1970-tallet til smidig utvikling på begynnelsen av 2000-tallet.

Tredje kapittel er et teorikapittel om retrospektivmøte og organisatorisk læring. I dette kapitlet beskrives to tilnærminger som brukes for å gjennomføre retrospektivmøter, Derby og Larsens modell og Dingsøyrs vurderingsfaktorer. Deretter gis det en innføring i Argyis og Schöns læringsmodeller som brukes for å svare på det første forskningsspørsmålet.

Fjerde kapittel er et teorikapittel om samlokalisert og distribuert team. I dette kapitlet gis det innføring om utfordringer knyttet til kommunikasjon i samlokalisert og distribuert team. Denne teorien brukes for å kunne svare på utfordringer knyttet til kommunikasjon i distribuerte team sammenlignet med samlokalisert team.

Femte kapittel er metodekapittelet som beskriver de vitenskapelige metodene som er anvendt for å besvare forskningsspørsmålene samt en beskrivelse av bedriftene og casene som deltok i dette studiet.

Sjette kapittel er resultatkapittelet som gjengir hvordan de ulike casene gjennomførte retrospektivmøtene samtidig som resultatene fra studiet presenteres.

Syvende kapittel er diskusjonskapitlet som svarer på forskningsspørsmålene. Det første delkapittelet diskuterer det første delspørsmålet om organisatorisk læring, men det andre delkapitlet diskuterer kommunikasjonsutfordringer i distribuerte team sammenlignet med samlokaliserte team.

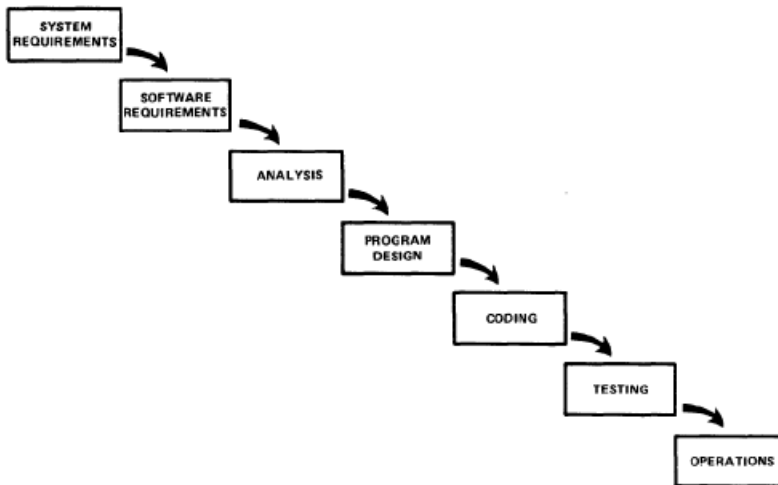
Åttende og siste kapittel er konklusjonen. I dette kapitlet besvares det overordnet forskningsspørsmålet. I tillegg skrives forslag til videre forskning.

Kapittel 2

Systemutviklingens historie: Fossefall til smidig utvikling

Dette kapitlet beskriver utviklingen til systemutviklingsmetodene fra fossefall til smidig utvikling og hva som kjennetegner smidig utviklingspraksis.

Systemutviklingshistorie har hatt en kontinuerlig og evolusjonær utvikling i de siste tiårene. I systemutviklingsbarndom besto utviklingen av to steg, *analyse* og *koding*. Denne måten å utvikle programvaren var godt egnet for små programvare, men var dømt til feile for større programvare, mente Royce i 1970 [4]. Han foreslo flere steg i utviklingsmetoden, og som senere førte til at fossefallsmodellen ble introdusert, se figur 2.1. En lang reise mot smidig utvikling har begynt.



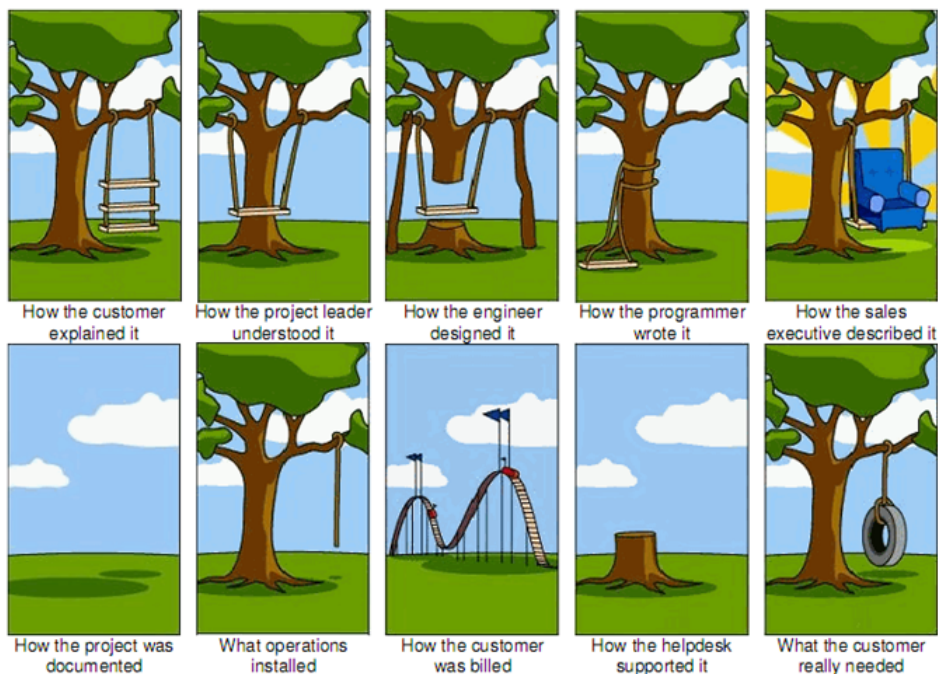
Figur 2.1: Den originale fossefallmodellen [4]

Fossefallmodellen hadde sitt fokus på at kunden visste nøyaktig hva kravspesifikasjonene var tidlig i prosjektet, slik at detaljerte spesifikasjoner ble ferdig dokumentert tidlig i krav og designfasen. Modellen ble etterhvert meget attraktiv, da den ble standard utviklingsmetodikk for offentlige – og industriprosjekter, som gjerne var store [5]. I en rapport fra Bell og Thayer i 1970 kommer det fram at store prosjekter identifiserer problemer med fossefallmodellen spesielt i kravspesifikasjonsfasen og analysefasen. Det var flere personer involverte i disse fasene og hadde ulike meninger som ført til at det tok lengre tid å gjennomføre prosjektet [20]. Petersen et al. identifiserte flere utfordringer knyttet til bruk av fossefallmodellen og konkluderer med at modellen ikke er egnet for store prosjekter. De påpeker at kunden er lite involvert for å gi tilbakemelding til utviklerne, og har dermed ikke mulighet til å avklare misforståelser og eventuelt endre kravspesifikasjonene. I tillegg påpekes at programvarene testes sent i utviklingsprosessen (se figur 2.1). Dette fører til at antall feil er økende og derfor blir valideringsfasen ressurskrevende [21].

I løpet av 1980-tallet var prosjektene blitt mer kompliserte og komplekse. Det ble viktigere at både utviklerne og kundene hadde samme forståelse av kravspesifikasjonene. På denne tiden ble verktøyet *prototyping* introdusert [22]. Prototyping ble velkjent for å øke forståelse rundt kravspesifikasjonene mellom begge parter. En rekke metoder som ble introdusert på denne tid tok i bruk dette verktøyet: *Spiralmodellen* (se figur 2.3) [5], *Rapid Application Development* (RAD)[23] og *Cleanroom Software Development* (CSD) [24]. Det er ikke lyktes å finne empiriske studier, men erfaringsrapportene viser at det har blitt enklere å definere kravspesifikasjoner ved bruk av spiralmodellen [25], RAD [26] og CSD [27], da protypene hjelper både kundene og utviklerne til å øke forståelsen rundt kravspesifikasjonene.

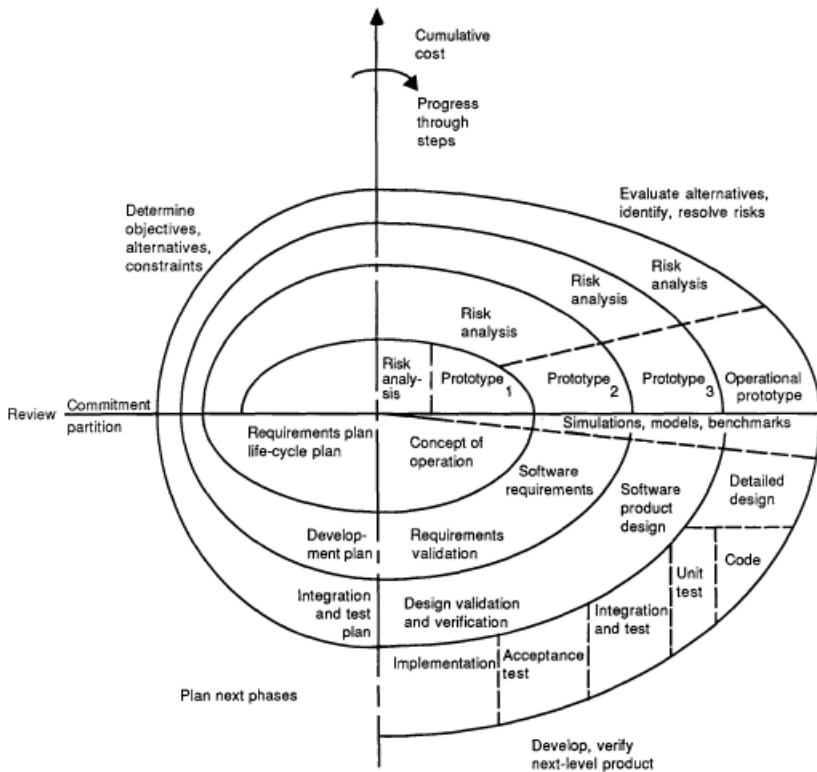
På slutten av 1980-tallet introduserte japanske Takeuchi og Nonaka en ny utviklingsmetode som baserte seg på rugby-tankegangen, som senere dannet grunnlaget for Scrum. Denne metoden baserte seg på seks kjerneverdier, deriblant selvorganiserende team, overlappende utviklingsfaser og læring sto sterkt i fokus. Takeuchi og Nonaka mente at kjerneverdiene tilsammen dannet grunnlaget for fleksibilitet og raskere utviklingstider [28]. Fleksibilitet ble viktigere da kravspesifikasjonene endret seg konstant på bakgrunn av uttrykket ”*kunden vet selv ikke hva de vil ha*”. Dette uttrykket betyr i praksis at kundens kravspesifikasjoner er ikke deres faktiske behov. Derfor endrer kunden kravspesifikasjonene sine til programvaren ettersom de får innsyn i form av prototype og lignende som gir uttrykk for hvordan programvaren har tenkt å bli [29]. Denne situasjonen er illustrert på figur 2.2¹. I tillegg til fleksibilitet var også overlappende faser en av kjerneverdier som sørget for lavere kostnader og utviklingstiden av produktet ble mindre i forhold til tidligere [28]. Takeuchi og Nonaka understreket at de beste teamene er selvorganiserte team som styrte seg selv med begrenset veiledning og involvering fra ledelsen. Derfor ble informasjon og kunnskapsutvikling et viktig faktum. Effektiv læring bidrar til å beskytte og forsterke bedrifts konkurransegrunnlag i et markedet [28]. Kjerneverdiene som er presentert danner grunnlaget for det vi kaller *smidig systemutvikling*.

¹Bildet er nedlastet 30.09.2015 fra: <http://projectcartoon.com/about/>



Figur 2.2: Hvordan kravspesifikasjonene kan bli oppfattet av ulike roller

Utover 1990-tallet økte både kompleksiteten og størrelsen på prosjektene, og for lave risikoprosjekter ble spiralmodellen unyttig. I følge Schwaber synker sannsynligheten for at prosjektet gir positive resultater når kompleksiteten øker på prosjektene ved bruk av spiralmodellen [30]. Det har ikke lyktes å finne empiriske studier som bekrefter påstanden, men en erfaringsrapport fra 1989 viser at det har vært vanskeligheter med å anvende spiralmodellen i praksis [25]. Det er flere som antar at spiralmodellen er en fossefallmodell i iterasjoner, men Boehm understreker at dette er feiltolkning av modellen [31]. I tillegg har det vært store vanskeligheter med å fastsette målsetninger og følge disse opp under prosjektet ved bruk av spiralmodellen [32, 33]. Prototypfasen er den eneste fasen som gir muligheten til å verifisere kravspesifikasjoner og innholdet til produktet, men denne fasen alene er ikke tilstrekkelig da kravspesifikasjonene endres konstant. I tillegg er modellen dokumenterorientert og har ikke fokus på kommunikasjon mellom teammedlemmene. Disse utfordringene åpner dørene for smidig utvikling som fokuserer mer på ansikt-til-ansikt kommunikasjon i forhold til detaljerte dokumentasjon og er mer fleksibel for endringer i kravspesifikasjon samtidig som den gir muligheten til mer kundekontakt underveis i prosjektet [34].



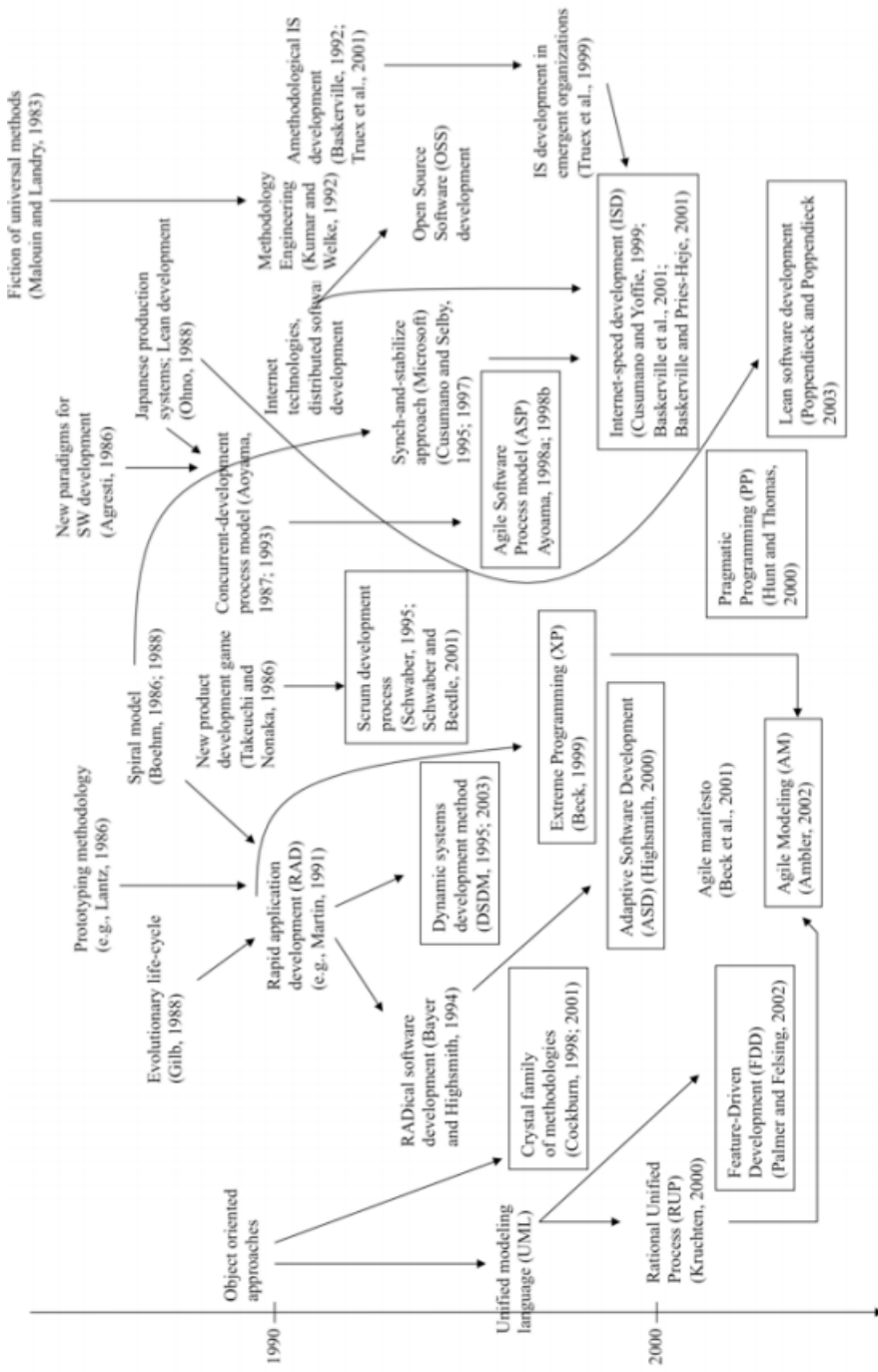
Figur 2.3: Sprialmodellen [5]

Smidig metoder ble introdusert på midten av 1990-tallet og på begynnelsen av 2000-tallet som en reaksjon på tradisjonelle og planbasert metoder, se figur 2.4. Abrahamson et. al beskriver smidig metode som inkrementell utvikling, fleksibel for endring, samarbeidsfokuset, enkel å bruke og kontinuerlig forbedringer [35]. I begynnelsen ble smidig metoder møtt med kritiske holdninger som påpekte at smidige metoder ikke vil konkurrere de tradisjonelle metodene [36], og at det er få empiriske studier som støtter at smidig metoder er effektive og passende for mange situasjoner [6]. Etter hvert som årene gikk meldte flere prosjekter at smidig metoder var godt egnet for deres prosjekt [37, 38, 39, 40]. Det anslås det at omtrent 94 % av alle organisasjoner bruker smidig utvikling i dag, der Scrum-metodikken er den mest utbredte utviklingsmetoden [41]. Det var Nonaka og Takeuchi som først nevnte Scrum som rugby-tilnærming, der de beskriver at de beste teamene er selvstyrte. De karakteriserte selvstyrte team som kryssfunksjonelle, autonome og transcendent [28]. Fornøyde ansatte, lavere sykefravær og færre omplassinger av ansatte i bedriften er noen av årsakene til at bruken av selvstyrte team har blitt utbredt [42]. Sutherland og Schwaber anvendte denne metoden på sine bedrifter på midten av 1990-tallet og gikk senere sammen for å formalisere metoden [43].

Det kan tenkes at en av årsakene til den store utbredelsen til Scrum siden den ble formalisert og fram til i dag, er at den stadig utvikler seg og formaliser disse endringene; *Scrum guide*² ble utarbeidet i både 2011 og 2013 [44, 45]. Ramesh et al. skrev en oversiktsartikkel knyttet til utfordringer i smidig distribuert utvikling [46], og Sutherland kom senere med en mulig løsning på utfordringer i forhold til distribuert utvikling med utgangspunkt i Scrum [47]. En annen årsak kan være forbindet med selve metoden, Qumer og Henderson-Sellers undersøkte og sammenlignet smidigheten til seks forskjellige smidige metoder samt to tradisjonelle metoder, og konkluderte med at praksisene i Scrum er mest smidig [34]. Det at metoden i seg selv kan være egnet til formålet og at den stadig utvikler seg og forbedres kan være årsaken til at den er mest utbredt i dag siden Schwaber formelt introduserte den i 1995.

I løpet av 2000-tallet har flere bedrifter fremstilt smidig utvikling som eget godt for deres prosjekter, men denne metoden følges ikke slik den er beskrevet formelt. Paasivaara et al. viser at distribuerte bedrifter har ikke muligheten til å gjennomføre Scrum slik den er beskrevet formelt på grunn av geografiske og kulturelle forskjeller, men metoden følges så langt den lar seg gjøre [39]. I flere prosjekter har blitt vanlig å bruke en tilpasset versjon av Scrum og andre smidige metoder som er tilpasset deres situasjon og opplever gode resultater med dette [48, 49]. Hvorvidt det er korrekt å bruke metodene på denne måten er avhengig av hvem som spør. I et langtidsperspektiv oppstår det gjerne nye behov som ikke dekkes av eksisterende metoder, og derfor er det viktig med nye eller reviderte metoder. I senere tid har det blitt utformet en metode for å dekke behovene for bruk av Scrum i distribuerte team [47].

²Revisjonsbeskrivelse for scrumguide, besøkt 12.12.2015 på følgende nettside:
<http://www.scrumguides.org/revisions.html>



Figur 2.4: Den evolusjonære utviklingen av utviklingsmetodikk i systemutvikling fra 1983 fram til 2003. [6]

Kapittel 3

Retrospektiv

Dette kapitlet handler om retrospektivpraksisen i smidig utviklingsmetodikk. I første delkapittel tas det nærmere titt på ulike definisjoner av retrospektivmøter, før jeg beskriver to tilnæringer for gjennomføring av retrospektivmøte i praksis. I påfølgende delkapittel gjennomgås empiriske studier og modeller for retrospektivmøter. Tilslutt beskrives retrospektivpraksisen i form av organisatorisk læring, nærmere bestemt Argyis og Schöns læringsmodeller.

I forrige kapittel har bakgrunnen for smidig utvikling blitt beskrevet. En av kjerneverdiene som smidig metoder baserer seg på er læring. I smidig metoder kan læringen foregå i retrospektivmøte. Hensikten med retrospektivmøte er å reflektere over hva som har foregått i forrige sprint eller i løpet av en tidsperiode. Denne refleksjonen kan utgjøre lærdom som kan brukes for å forbedre fremtidige praksiser. I løpet av systemutviklingslevetid har det blitt brukt ulike begreper i forhold til denne praksisen [14] og i denne avhandlingen forholder vi oss til begrepet retrospektivmøte. I tabell 3.1 har jeg beskrevet ulike retrospektivmøter som jeg har kjennskap til. Det siste prinsippet bak smidig manifestet understreker viktigheten med å gjennomføre retrospektivmøter i smidige utviklingsmetodikk [50]:

”Men jevne mellomrom reflekterer teamet over hvordan det kan bli mer effektivt og så justerer det adferden sin deretter” - 12.prinsipp bak smidig manifest.

Retrospektiv type	Beskrivelse
Prosjektretrospektiv	Denne gjennomføres etter at et prosjekt er avsluttet og at produktet er tatt i bruk.
Iterasjonsretrospektiv	Gjennomføres etter at en prosjektfase eller en iterasjon er fullført. Eksempelvis etter en sprint i Scrum-metodikken

Tabell 3.1: Ulike typer for retrospektivmøter.

3.1 Definisjoner på retrospektiv

Det finnes flere definisjoner på retrospektiv, men i dette delkapittelet skal vi se nærmere på Derby & Larsen og Dingsøyrs sine definisjoner. Disse definisjoner er vidt forskjellige, der den ene fokuserer mer på følelsesmessige faktorer i retrospektivmøte, mens den andre fokuserer mer på læringsutbytte. Tilslutt beskrives min definisjon på retrospektivmøte.

Derby og Larsen sin definisjon av retrospektivmøte er beskrevet som følger [7]:

”En spesiell møte hvor teamet kan samles etter å ha fullført ett inkrement av arbeid for å kontrollere og tilpasse sine metoder og teamarbeid. Retrospektiv aktiverer slik at teamet som helhet lærer og fungerer som en katalysator for endringer og generere handling. Å prøve lære fra gamle prosjekter innebærer mer enn å bare gå gjennom en sjekklister for prosjekt-revidering og apatiske avslutninger. Og, i motsetning til tradisjonelle retrospektiv eller prosjektgjennomganger, fokuserer retrospektiv ikke bare på utviklingsprosessen, men også på teamet og teametsproblemer. Og teametsproblemer er vel så utfordrende som tekniske problemer – om ikke mer.”

Dingsøyr sin definisjon av retrospektivmøte lyder som følger [14]:

”En kollektiv læringsaktivitet som kan bli organisert for prosjekter enten når de ender en fase eller har avsluttet. Den viktigste motivasjonen er å reflektere over hva som har skjedd i prosjektet for å forberede framtidige praksis – for de enkelt individene som har deltatt i prosjektet og for organisasjonen som en helhet. Det fysiske resultatet av et møte er en retrospektivrapport.”

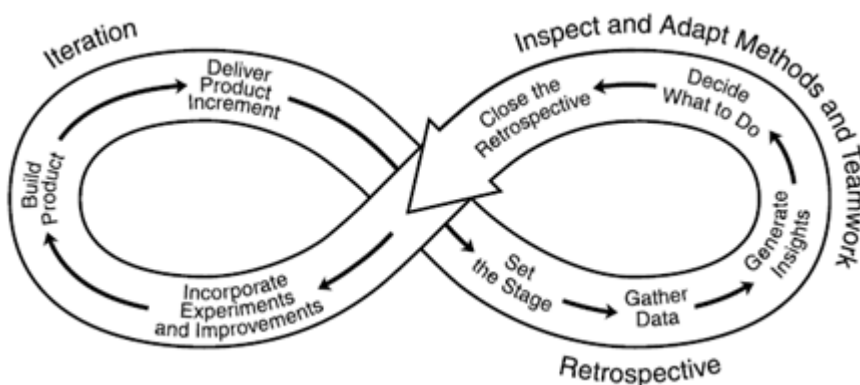
Min definisjon av retrospektivmøte er en sammensetning av definisjonene ovenfor:

”Et møte som er organisert som en læringsaktivitet for prosjekter som enten avslutter en iterasjon eller et prosjekt. Formålet er å kontrollere og tilpasse sine metoder -og teamarbeidet. Dette innebærer å fokusere på å reflektere over hva som har skjedd i prosjektet både når det gjelder tekniske og personlige utfordringer i teamet for å forbedre framtidige praksis. Utbyttet av møtet skal være refleksjon over siste periode, der det utarbeides forbedringstiltak på de viktigste utfordringene og behovene teamet står overfor.”

3.2 Gjennomføring av retrospektivmøte

Det finnes flere måter å gjennomføre retrospektivmøter, der hensikten ligger å ta læring av siste utviklingsfase. I denne oppgaven fokuserer vi på iterasjonsretrospektiv som beskriver hvordan reelle problemer påvirker teamet, hovedsakelig innad i teamet og løsninger som krever lite administrativt arbeid og dermed begrenset involvering fra ledelsen.

3.2.1 Derby og Larsens modell



Figur 3.1: Iterasjonsretrospektiv i praksis [7]

Derby og Larsen har skrevet en bok om gjennomføring av retrospektivmøter basert på deres tiårs erfaring [7]. I denne sammenhengen har de utarbeidet en modell som består av fem trinn for gjennomføring av iterasjonsretrospektiv og beskrevet flere aktiviteter som kan brukes i de ulike trinnene. De fem trinene er oppsummert på slutten av dette delkapitlet, se tabell 3.2. Iterasjonsretrospektiv er en syklus som er illustrert på figur 3.1.

Det første trinnet, *Sett scenen* (Set the stage) er oppvarmingsfase der hver deltaker blir bedt å beskrive sine forventinger til møtet med ett ord. Ved å få alle deltakerne til å delta tidlig i møtet er sannsynligheten større for at de deltar i avgjørelser som teamet tar senere. Dette er nødvendig siden målet med retrospektivmøte er gruppetenking og læring av hverandre. I tillegg er det viktig at denne fasen bygger opp en god atmosfære som gir muligheten til å snakke om emosjonelle temaer og krevende problemstillinger senere. Ved å gjennomføre dette trinnet vil deltakerne være komfortable med å ytre sine synspunkter senere i møtet.

Det andre trinnet, *Datainnsamling* (Gather data) er en innsamlingsfase som er hovedtrinnet i et retrospektivmøte. I dette trinnet skal deltakerne som har vært fraværende fra prosjektet få med seg nødvendig informasjon. Hovedformålet med dette trinnet er at alle i teamet skal ha det samme utgangspunktet for å diskutere

problemsstillingene. Forskjellige deltakere har gjerne ulike synspunkter om ulike problemstillinger med forskjellig utgangspunkt. Ved å samle inn dataene fra deltakerne er det mulig å få det helhetlige bildet. Det begynnes med å spørre om de harde dataene slik som andel fullført oppgaver, burndown chart og lignende, men det er minst like viktig å spørre om emosjonelle spørsmål. Ved å bruke en strukturert måte å snakke om følelser gir dette deltakerne muligheten til å være komfortable med å ta opp problemstillinger som er følelsesladet og som er relatert til teamsamarbeid. Gjennom datainnsamlingen får deltakerne i teamet samme utgangspunkt for å diskutere både faktaer og følelser som fører til bedre tenking og handling for resten av retrospektivmøtet.

Det tredje trinnet, *Gi innsikt* (Generate insights) er en innsiktsfase, der teamet vurderer dataene for å identifisere styrke og problemstillinger fra forrige trinn. I dette trinnet skal teamet undersøke forhold, samhandlinger, mønstre som bidrar til deres suksess, samtidig som de ser seg for risikoer og uønsket handlinger eller utfall. Formålet med trinnet er å se hvordan teamet kan arbeide mer effektivt, noe som er det ultimate formålet med retrospektivmøte. Resultatet av dette trinnet skal være en liste over forbedringstiltak som er utarbeidet av teamet. Dette fører til at deltakerne ser det helhetlige bilde for å løse oppståtte problemstillinger.

Det fjerde trinnet, *Avgjør hva som skal gjøres* (Decide what to do) er en planleggingsfase for forbedringspotensialer. I dette trinnet foregår utvelgelsesprosessen for forbedringstiltak. Her velges ett eller to viktige forbedringstiltak som det utføres eksperimenter for neste iterasjon. For å gjennomføre dette forbedringstiltaket må hvert enkelt deltaker i teamet få tildelt sine arbeidsoppgaver, i stedet for at teamet som helhet får oppgaven. På denne måten hindrer vi at deltakerne tenker "teamet vil gjøre det" og som regel vil ende med at ingen vil ta ansvaret for å gjøre det. Formålet med trinnet er å gi en struktur og veiledning for at teamet skal planlegge eksperimenter og handlinger.

Det femte og siste trinnet, *Avslutt retrospektiv* (Close the retrospective) er avslutningsfasen for retrospektivmøtet. I dette trinnet skal det dokumenteres hva utfallet av møtet ble og hvordan tiltakene fra møtet skal følges opp. I tillegg bør det tenkes på hvordan teamet kan ivareta sin lærdom fra møtet. Helt tilslutt kan det utføres en retrospektiv på selve retrospektivmøtet, der deltakerne forklarer hva som gikk bra og hva som kan gjøres bedre i neste retrospektivmøte.

Trinn	Beskrivelse
Sett scenen	Minne deltakerne på formålet med møtet og oppmuntre de til delta aktivt i møte ved å spørre om deres forventninger til møtet.
Datainnsamling	Foreta datainnsamling slik at deltakerne har felles utgangspunkt for å diskutere utfordringene i innsiktfasen.
Gi innsikt	Finne ut hvordan teamet kan arbeide mer effektivt ved å lage en liste over forbedringstiltak.
Avgjør hva som skal gjøres	Velge ett eller to forbedringstiltak fra listen og planlegge hvordan dette skal gjennomføres.
Avslutt retrospektiv	Gjennomføre refleksjon på retrospektivmøte og læringsutbytte fra møtet.

Tabell 3.2: Oppsummering av Derby og Larsens modell

3.2.2 Dingsøyrs vurderingsfaktorer

Dingsøyr presenterte åtte forskjellige vurderingsfaktorer som bedriftene bør vurdere for gjennomføring av retrospektivmøter [14]. Vurderingsfaktorene er utarbeidet basert på en case studie og tre forskjellige teknikker for gjennomføring av retrospektivmøter. Disse faktorene er oppsummert på slutten av delkapitlet, se tabell 3.3

Den første vurderingsfaktoren handler om god atmosfære for å gjennomføre retrospektivmøte. Dingsøyr påpeker at egenskaper som åpenhet, tålmodighet, lytteevne, høflighet og eksperimentering ved bruk av nye ord og begreper er eksempler på ingrediensene for en god diskusjon i retrospektivmøte. Dette løses ved å ha en dyktig prosessleder som oppmuntrer til en åpen dialog og hindrer kritisering av enkelt individer. Deltakerne i retrospektivmøtet må tro at enhver deltaker har gjort sitt beste i løpet av siste iterasjon for å skape god atmosfære.

Den andre vurderingsfaktoren forklarer hvem som bør være invitert til møtet. Dingsøyr foreslår at alle som har deltatt på prosjektet bør være med på møtet, bortsett fra ledelsen. Dette bidrar til at deltakerne gir sine kunnskap som er relevant for framtidig prosjekt gjennom *sosialisering*. Ved å invitere flest mulig deltakere gir det muligheten til å utvide eksisterende praksis i organisasjonen, spesielt hvis deltakere fra nye prosjekter er invitert. Eksterne interessenter som kunden bør ikke inviteres til møtet, da dette endrer hensikten med møtet, fra intern møte til interessant relasjon.

Den tredje vurderingsfaktoren gir en indikasjon på hvorvidt det lønner seg med hjemmelekse til deltakerne og fasilitatoren. Dette betyr i praksis at deltakerne reflekterer over spørsmål i forkant av møtet. Dingsøyr argumenterer for at hjemmelekse er viktig da læringsprosessen foregår over en tidsperiode og deltakere vil enklere bidra til møtet da de er forberedt. På en annen side viser han til at et møte med høyt antall deltakere vil høyest sannsynlig komme fram til de viktigste problemene uten hjemmelekse, samtidig understrekes det at individuell refleksjon krever tid. Det er nødvendig med hjemmelekse for fasilitator for å kunne stille spørsmål som får deltakerne til å reflektere.

Den fjerde vurderingsfaktoren beskriver hvorvidt fasilitator er nødvendig. Ut ifra de tre teknikkene forklarer Dingsøyr at det er viktig å ha en fasilitator for møtet og denne personen bør være godt trent til å følge opp uttalelser fra deltakere som er uklare. En prosjektleder bør ikke være en fasilitator da det er vanskelig å ta opp sensitive temaer. Fordelen med ekstern fasilitator er at deltakerne må forklare mer gjennomtenkt i forhold til en intern fasilitator.

Den femte vurderingsfaktoren forklarer i hvilken grad diskusjonen skal være åpen eller strukturert. Åpen diskusjonen er en sentral i forhold til å bidra til læring, men denne formen for diskusjon er gjerne tidskrevende og vil fokusere på et begrenset antall problemsstillinger. En strukturert diskusjon bidrar for at hver deltaker får like stor innflytelse på diskusjonen, uavhengig av om deltakerne velger å delta gjennom introspeksjon eller interaktiv idémyldring.

Den sjette vurderingsfaktoren beskriver om ledelsen bør være med i møtet. Dingsøyrl påpeker at ledelsen ikke bør være med i møtet, da intensjonen er å fokusere på læring og ledelsen har som rolle å evaluere ansatte. En prosjektleder bør være med i møtet, da vedkommende har en generell oversikt over prosjektet i forhold til resten av deltakerne. Prosjektlederen vil beskytte alle beslutningene som er tatt i løpet av prosjektet og dermed blir det vanskelig å ha en fri utveksling av idéer om forbedringer. En fasilitator som er klar over denne problemstillingen skal klare å håndtere denne situasjonen.

Den sjuende vurderingsfaktoren forteller om hva utfallet av retrospektivmøte skal være. Det fysiske resultatet av møtet kan enten være en liste av anbefalinger eller en rapport som beskriver prosjektet. En liste med anbefalinger kan brukes for å sikre læring i andre prosjekter, dersom formålet er kunnskapsspredning i bedriften. En rapport kan brukes hvis formålet er forbedring og rapportbeskrivelsen av prosjektet skal inneholde hva som gikk bra og dårlig og årsaken til dette.

Den åttende og siste vurderingsfaktoren beskriver om læringen skal være taus eller eksplisitt kunnskap. Taus kunnskap er kunnskap som ikke lar seg beskrive med ord og uttrykk og det er noe som ligger implisitt i mennesket og påvirker hvordan vi handler [51]. Eksplisitt kunnskap er kunnskap som kan beskrives ved hjelp av ord og uttrykk [51]. Distribuere relevante erfaring fra ett prosjekt ses som en viktig læringsmekanisme. Dingsøyrl forklarer at små bedrifter bør fokusere på å dele taus kunnskap, mens større bedrifter bør bruke ressurser for å fokusere på eksplisitt kunnskap.

Faktor	Beskrivelse
Atmosfære	Ha åpen dialog, hindrer kritisering av enkelt individer og opprettholde god atmosfære.
Hvem skal inviteres i retrospektivmøte	Alle som har blitt bidratt bør være med, men ikke ledelsen og kunden.
Hjemmelekse	Refleksjon krever tid, og dette kan gjøres med hjemmelekse. De fleste problemstillingene bør også kunne identifiseres ved å tillatte deltakere til å reflektere under selve møtet.
Fasilitator	Bør ikke være prosjektleder og må kunne følge opp interessante og uklare utsagn. Det er fordel med ekstern fasilitator da en må forklare mer gjennomtenkt.
Åpen eller strukturert diskusjon	Åpen diskusjon tar opp få utfordringer og er tidskrevende, mens strukturert diskusjon bidrar for at alle deltakerne får lik innflytelse og tar opp flere utfordringer.
Ledelse	Ledelsen bør ikke være med på møtet, mens prosjektlederen kan være med.
Fysisk resultat	En rapport hvis målet er kunnskapsspredning i bedriften eller en liste med anbefalinger hvis målet er forbedringer.
Taus eller eksplisitt kunnskap	Små bedrifter bør fokusere på taus kunnskap, mens større bedrifter bør fokusere på eksplisitt kunnskap.

Tabell 3.3: Oppsummering av Dingsøyrs vurderingsfaktorer

3.3 Empiriske studier og modeller for retrospektivmøter

Modeller for retrospektivmøter

Det er flere studier og modeller på hvordan retrospektivmøte skal gjennomføres for å få mest mulig nytte av det [7, 11, 14, 52, 53]. Allerede før smidig utvikling ble retrospektivpraksisen brukt for å evaluere systemutviklingsprosessen, men på denne tiden omfattet ikke begrepet om utfordringer innad i teamet på lik linje med tekniske utfordringer. Tidemann forslo tre typer retrospektivpraksiser i fossefallsmodellen i bestemte faser og funnene ved studiet viste at retrospektivpraksisen var ett betydelig effektiv organisatorisk læringsverktøy [54]. Denne praksisen har utviklet seg fra å gjennomføre retrospektivmøter på prosjekter til å gjennomføre retrospektivmøter på sprinter i smidig utviklingsmetoder. Derby og Larsen har skrevet en bok om praktisk gjennomføring av retrospektivmøter, og [7] Kua har utvidet denne modellen med andre typer retrospektivmøter [15].

Birk et al. skrev en artikkel om bakgrunnen for hvorfor retrospektivmøte må gjennomføres og foreslo en modell for gjennomføring av praksisen [11]. De viste til at retrospektivpraksisen kan brukes for å lære av tidligere prosjekter og dermed hindre fiasko for fremtidige prosjekter. Modellen består av tre trinn for gjennomføring av retrospektivmøter og disse er forberedelse, datainnsamling og dataanalyse. De anbefaler bruk av ekstern fasilitator, da fasilitatoren har en fordel med å være nøytral og objektiv. Fasilitatoren er den eneste som trenger å forberede seg til møtet. Dette gjøres ved å være klar over hvilke spørsmål som bør stilles til teamet for å få dem til å reflektere, og samtidig bør fasilitatoren hente inn bakgrunnsinformasjon om prosjektet. Denne informasjonen vil hjelpe med å stille konkrete spørsmål som får teamet til å reflektere dypere. Datainnsamlingen kan gjennomføres på tre måter: semi-strukturert intervju, fasilitere gruppediskusjoner og KJ-sesjoner. Deretter bør teamet identifisere viktige temaer (utfordringer) og prioritere dem før analysen og på denne måten sikre at det viktigste teamet analyseres først. I analysetrinnet foreslår de å bruke fiskebeinsdiagram for å finne de underliggende årsakene til utfordringene. Resultatet av møtet skal være en rapport.

Dingsøyr diskuterte tre forskjellige metoder for gjennomføring av retrospektivmøter. Ett av disse var Birk et al. sin metode for gjennomføring av retrospektivmøter. Denne metoden består av to teknikker: KJ - en strukturert idemyldringsteknikk og fiskebeinsdiagram (årsaksanalyse) som brukes for å gruppere og analysere idemyldringen etter KJ-sesjonene. I dette tilfellet ble det brukt til å finne underliggende årsaker til at kravspesifikasjonene endret seg i løpet av det aktuelle prosjektet [14]. Stålhane et al. sammenlignet Birk et al. metode med strukturert intervju, der funnene viste at begge metodene kan brukes til å tilegne seg kunnskap [55]. Bjørnson et al. gjennomført et eksperiment ved bruk av Birk et al. metode for å vise at kvaliteten og effektiviteten øker hvis det brukes "causal maps" i stedet for fiskebeinsdiagram for å analysere suksess og utfordringer [56]. Bakgrunnen for dette er at fiskebeinsdiagrammer har begrenset muligheter for å analysere problemsstil-

linger i forhold til å bruke ”causal maps” som er mer fleksibel. Lehtinen et al. har utviklet en sanntidsbasert og skybasert verktøy for å løse utfordringer som gjorde det vanskelig å gjennomføre retrospektivmøter i geografisk distribuerte team. Verktøyet omfatter å kartlegge behovene for årsaksanalyse-praksisen i distribuerte team [53]. Dingsøy et al. sammenlignet erfaringsrapporter og enkel retrospektiv¹ der funnene viser at de er ikke overlappende. Erfaringsrapporter dekker temaer som er relatert til kunde, mens enkel retrospektiv dekker utfordringer knyttet til kjerneprosesser i programvareutvikling [57].

Hukommelsessvikt

Det er flere utfordringer med å gjennomføre retrospektivmøter og ett av dem er hukommelsessvikt. Kransdoff kritiserer retrospektivpraksisen fordi den holdes etter en iterasjon med arbeid eksempelvis etter en sprint, og da har deltakerne sannsynligvis glemt hva vedkommende hadde på hjertet. Han mener dette kan føre til hukommelsessvikt, og foreslo at dataene samles under prosjektet, eksempelvis med korte intervjuer med hensyn til mer troverdig datamateriell [58]. Det samme observerte Baird et al. der det fremgår at deltakerne ikke husket all informasjon om foregående arbeidsperiode. Dette er på grunn av at hukommelsen til deltakerne avtar med tiden, og det er blir vanskeligere å gjenfortelle informasjonen som nøyaktig som mulig [59].

Læring, motivasjon og tidsbruk

Læring er sentralt i et retrospektivmøte og det kreves både ressurser i form av tid og motivasjon for å gjennomføre retrospektivmøte i et prosjekt. Zedtwit argumenterte for at teamene bør fokusere på å finne de underliggende årsakene til utfordringen da det kan bidra til å forbedre eksisterende praksis [60]. I 2002 påpekte Glass at retrospektivmøte ikke gjennomføres i flere tilfeller på bakgrunn av at teamet er under tidspress og prioriterte ikke retrospektivpraksisen. Glass referer til Brössler som påpeker at manglende prioritering av retrospektiv fører til at teamet antakeligvis gjør den samme feiltakelsen igjen i neste prosjekt [10]. Dingsøy et al. eksemplifiserer utfordringer som vil blitt unngått dersom bedriften hadde brukt ressurser på læring. De utdyper at det er flere bedrifter som bruker lengre tid enn planlagt på testfasen, og denne problematikken kan forebygges ved å lære av hvilke forbedringstiltak som har blitt tatt i bruk av tidligere prosjekter [57]. Det kreves mindre enn 0,1 % av total prosjektetid for å gjennomføre retrospektivmøter [14], og dette gjenspeiler at ressurser bør ikke være årsaken til nedprioritere retrospektiv fra et økonomisk perspektiv. Deltakerne i et retrospektivmøte må være villige til å lære, dersom teamet skal få ett godt utbytte fra retrospektivmøter [7]. Dingsøy og Derby et al. beskriver at en god atmosfære er nødvendig for å skape entusiasme, og derfor må ethvert teammedlem tro at alle andre teammedlemmene har gjort sitt beste i løpet av siste iterasjon slik at dette skaper en god atmosfære [7, 14].

¹Engelsk: Lightweight Postmortem Reviews

Forbedringstiltak

Forbedringstiltakene er utarbeidet for å løse utfordringene og behovene teamet står overfor. Litteraturen viser varierende resultater i forhold til gjennomføringen og nytteverdien ved disse forbedringstiltakene. Myllyaho et al. gjennomførte en litteraturstudie på retrospektivmøter, og funnene viser at retrospektivmøtene er nyttig for å forbedre arbeidsprosessen til teamet [61]. Dette betyr at forbedringstiltakene blir gjennomført og at teamet opplever læringseffekten som nyttig. Dette er en imotsetning til Druyr et al. som viser til at teamene merket ikke forbedringer i praksisen ved å gjennomføre retrospektivmøter. I tillegg viser han til at en av grunnene for manglende gjennomføring av forbedringstiltakene var usikkerheten rundt hvem som hadde ansvaret for gjennomføringen [62]. Dubinsky og Hazzan introduserte retrospektivpraksisen for en storskala prosjekt og funnene viser at teamet identifiserer prosessutfordringer og finner forbedringstiltak på disse utfordringene [63]. Samtidig beskrives behovet for å måle effekten av forbedringstiltakene. På samme måte som Salo viser til at gjennomføring av forbedringstiltakene var nødvendig for å få nytte av retrospektivmøtene [19]. Derby og Larsen beskriver at deltakerne er mer entusiastiske når de selv foreslår forbedringstiltak for utfordringer de står overfor framfor at ledelsen kommer med forbedringstiltak som de må gjennomføre [7].

3.4 Modell basert på eksisterende litteratur

Modellen er basert på eksisterende litteratur, der den har tatt utgangspunktet i Derby og Larsen modell, og utvidet modellen med ett ekstra trinn som ikke dekkes av modellen. Dingsøyrs vurderingsfaktorene kan dekke flere trinn i denne modellen, men er plassert der den hensiktsmessig tilhører. En beskrivelse av trinnene er presentert nedenfor og er oppsummert på tabell 3.4

Det første trinnet er en *forberedelse* for retrospektivmøtet. I denne fasen avgjøres det hvem som skal inviteres til møtet og om disse skal ha hjemmelekse, der de skal reflektere over spørsmål i forkant av møtet og i tillegg velges fasilitatoren. Alle som har bidratt til prosjektet bør inviteres, bortsett fra ledelsen og kunden [11, 14]. Ledelsen bør ikke inviteres da de har intensjon om å evaluere ansatte som hemmer læringsprosessen, mens kunden vil endre fokuset på retrospektivmøtet. Dingsøyr og Birk et. al argumenterer for at ekstern fasilitator bør brukes for å forklare mer gjennomtekt til fasilitatoren og dermed reflektere dypere [11, 14]. Dingsøyr [14], Birk et al. [11] og Kua [15] uttrykker at fasilitatoren har som hjemmelekse å forberede retrospektivmøtet, der det planlegges hvilke aktiviteter som skal gjennomføres og hvilke spørsmål som bør stilles for å få deltakerne til å reflektere.

Trinn	Derby og Larsen [7]	Dingsøyrr [14]	Andre kilder
1. Forberedelse		Hvem bør inviteres, hjemmelekse, ledelsen, fasilitator	Birk et.al [11] Kua [15]
2. Introduksjon	Sett scenen	Atmosfære	Birk et.al [11] Kua [15]
3. Datainnsamling	Datainnsamling	Taus eller eksplisitt kunnskap	Birk et.al [11] Kransdoff [59] Kua [15] Lehtinen [52] Stålhane et. al [64]
4. Diskusjon og utarbeide forbedringstiltak	Gi innsikt	Åpen eller strukturert diskusjon	Birk et.al [11] Kua [15] Stålhane et. al [64]
5. Planlegging for gjennomføring av forbedringstiltak	Avgjør hva som skal gjøres		Birk et.al [11] Druyr et al. [62] Kua [15]
6. Avslutt retrospektiv	Avslutt retrospektiv	Fysisk resultat	Kua [15]

Tabell 3.4: Modell basert på eksisterende litteratur

Det andre trinnet er *introduksjon* der fasilitatoren presenterer agendaen for møtet og spør om deltakernes forventning til møtet. Derby og Larsen [7], Dingsøyrr [14] og Birk et al. [11] hevder at god atmosfære er nødvendig for at teamet skal få nytte av møtet. Dette trinnet bør bygge opp denne atmosfæren slik at kan deltakerne føle seg mer komfortable med ytre sine synspunkter i møtet. Ved å spørre om forventingene til deltakerne i retrospektivmøte bidrar det til å bygge en god atmosfære [7].

Det tredje trinnet er *datainnsamling* fra siste arbeidsperiode. Derby og Larsen utdyper at hensikten med datainnsamlingen er å ha felles utgangspunkt for å diskutere utfordringene i neste trinn [7]. Denne datainnsamlingen kan gjennomføres ved bruk av flere teknikker, eksempelvis ved bruk av post-it lapper [7, 15], strukturert intervju under sprinten [59], bruk av verktøy for å skrive virtuelle lapper [15, 52] og KJ-sesjoner [11, 64]. I tillegg har Derby og Larsen beskrevet flere aktiviteter som kan brukes for gjennomføring av datainnsamlingen [7]. Dingsøyrr anbefaler at små bedrifter bør fokusere på taus kunnskap, mens større bedrifter bør fokusere på eksplisitt kunnskap. Konvertering av taus kunnskap til eksplisitt kunnskap kan gjennomføres ved bruk av flere datainnsamlingsmetoder eksempelvis ved bruk av idémyldring, KJ-sesjoner og andre teknikker der deltakerne formidler kunnskapen sin gjennom disse metodene [14].

Det fjerde trinnet er *diskusjon og utarbeide forbedringstiltak*. Hensikten med dette trinnet er å prioritere utfordringene og deretter diskutere disse i felleskap og utarbeide forbedringstiltak på disse. Birk et al. påpekte at utfordringene burde prioriteres før de ble diskutert, og deretter gjennomføre årsaksanalyse på viktigste utfordringene for å finne de underliggende årsakene til utfordringene [11]. Ved bruk av årsaksanalyse konverteres taus kunnskap til eksplisitt kunnskap. Dingsøy diskuterte strukturen på diskusjonene og anbefalte at deltakerne benyttet seg av strukturert diskusjon i samlokalisert retrospektivmøter [14]. Tilsvarende argumenterte Kua for at enkelte deltakere vil dominere diskusjonen eller vil "skjule seg" ved å ikke delta i distribuert retrospektivmøter (videokonferanse) og derfor bør diskusjonen være strukturert [15]. I diskusjonen bør fasilitatoren kunne følge opp uklare og interessante utsagn fra deltakerne i teamet [11, 14], og ved bruk av ekstern fasilitator vil deltakernes forklaring være mer gjennomtenkt i forhold til å ha intern fasilitator [11, 14, 15]. Derby og Larsen anbefaler at teamene må planlegge hvordan de skal implementere forbedringstiltaket i praksis [7].

Det femte trinnet er *planlegging for gjennomføring av forbedringstiltak*. Derby og Larsen informerer at forbedringstiltakene bør være konkrete og tilrettelagt, da deltakere ikke vil gjøre "ekstra arbeid" for å gjennomføre forbedringstiltak [7]. Kua foreslår at forbedringstiltakene kan synliggjøres ved å sette dem i sprintkøen, da deltakerne daglig forholder seg til den [15]. I tillegg anbefaler både Kua og Derby og Larsen at en bestemt forbedringstiltak tildeles til en bestemt deltaker, selv om teamet som helhet har ansvaret for gjennomføringen av forbedringstiltaket [7, 15]. Drury et al. viser til manglende gjennomføringsevne for forbedringstiltakene, der årsaken er usikkerheten rundt hvem som har ansvaret [62]. Kua foreslo bruk av tidsfrist for gjennomføring av forbedringstiltak, da man setter et søkelys på dette [15].

Det sjette og siste trinnet er *avslutt retrospektiv* som innebærer at deltakerne dokumenterer og reflekterer over utbyttet av retrospektivmøtet. Derby og Larsen, og Kua har utarbeidet aktiviteter for å få deltakerne til å reflektere over nytteverdien av retrospektivmøtet [7, 15]. Dingsøy opplyser at det fysiske utbyttet skal være en rapport om målet er kunnskapsspredning eller liste med anbefaling hvis målet er forbedringer [14].

3.5 Organisatorisk læring

Læring er en prosess der kunnskap tilegnes av individer. Nonaka beskriver læring som en konverteringsprosess mellom taus og eksplisitt kunnskap [51]. Taus kunnskap er kunnskap som ikke lar seg beskrive med ord og uttrykk og det er noe som ligger implisitt i mennesket og påvirker hvordan vi handler [51]. Eksplisitt kunnskap er kunnskap som kan beskrives ved hjelp av ord og uttrykk [51]. Det er vanskelig å lære bort den tause kunnskapen til andre teammedlemmer, og derfor bør kunnskapen bli eksplisitt før andre teammedlemmer kan nyttiggjøre seg av kunnskapen [57]. I retrospektivmøter foregår det en konverteringsprosess fra taus kunnskap til eksplisitt kunnskap som kalles for eksterialisering i modellen til Nonaka [51]. Eksterialisering skjer ved at et teammedlem reflekterer over dens erfaringer og deler denne erfaring med andre teammedlemmer [14]. Dette omfatter teknikker som KJ-sesjon, idémyldring, årsaksanalyse og flere andre teknikker som gjør teammedlemmene i stand til å reflektere [14].

Organisatorisk læring er et begrep som omhandler hvordan en organisasjon er i stand til å tilegne, bevare og anvende eksisterende kunnskap i organisasjonen. Læring kan foregå på to nivåer, nemlig individuell læring og organisatorisk læring. Dingsøyr påpeker at retrospektivmøte kan brukes som en metode for å utnytte kunnskap fra individuell nivå til organisatorisk nivå [14]. Det kreves kunnskapsforvaltning for å nyttiggjøre kunnskapen på et organisatorisk nivå. Dette krever at kunnskapen samles inn og forbedres og deretter distribuerer kunnskapen tilbake til ansatte i organisasjonen [57]. Dingsøyr et al. formidler hvordan erfaringer fra prosjekter kan gjøres tilgjengelige for andre ansatte i organisasjonen slik at det blir nyttig for dem. Dette gjøres ved å få taus kunnskap over til eksplisitt kunnskap (eksterialisering) og lagre denne kunnskapen i ”kunnskapsdatabasen” slik at den er tilgjengelig som støtte for fremtidig prosjekter [57].

Argyis og Schön har definert organisatorisk læring som følgende [1]:

”Organisatorisk læring oppstår når individer i en organisasjon opplever en problematisk situasjon og undersøker den på organisasjonsvegne. De opplever en overraskende ujevnheter mellom forventet og faktisk resultat av en handling. Derfor svarer de på denne ujevnheten gjennom en prosess med tanker og ytterligere handling som fører dem til å endre deres ”bilde” av organisasjonen eller deres forståelse av organisatoriske fenomener. Dette brukes for å restrukturere sin virksomhet for å bringe resultater og forventinger til rette, og dermed endre bruksteori. For å bli organisatorisk må man lære at resultatene fra organisatorisk henvendelse² holdt i dens medlemmers sinn og/eller i de epistemologiske artefakter (kartet, hukommelse og programmer) innebygd i organisatoriskmiljøet.”

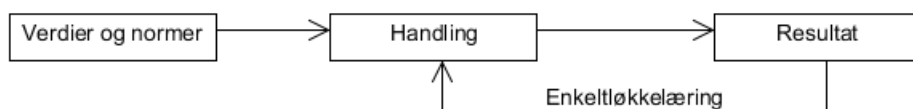
²Engelsk: Organizational Inquiry

En organisasjonen lærer gjennom at individene i organisasjonen lærer. Argyis og Schön har utarbeidet to læringsmodeller for hvordan en organisasjon lærer gjennom at individene lærer. Individene har hindringer for å lære, som kan resultere at organisasjonen ikke lærer [1]. Dette rammeverket brukes for å undersøke organisatorisk læring i retrospektivmøter fordi den gir innsyn hvilke hindringer systemutviklings-team har for læring. I neste delkapittel skal det gås gjennom to former for læring, enkeltløkkelæring og dobbeltløkkelæring. Deretter ser vi på begrepet bruksteori og introdusere to læringsmodeller, der dette bruksteori brukes.

3.5.1 Enkeltløkkelæring

Enkeltløkkelæring er ett uttrykk for å endre gjeldende prosedyrer etter hvert som avvik og feil oppstår. I følge Argyis og Schöns defineres enkeltløkkelæring som følgende³:

”By single loop learning we mean instrumental learning that changes strategies of action or assumptions underlying strategies in ways that leave the value of a theory of action unchanged...In such learning episodes, a single feedback loop, mediated by organizational inquiry, connects detected error – that is, an outcome of action mismatched to expectations and, therefore, surprising to organizational strategies of action and their underlying assumptions. The strategies of action and their underlying assumptions. These strategies or assumptions are modified, in turn, to keep organizational value and norms. The values and norms themselves ... remain unchanged.”



Figur 3.2: Viser hvordan enkeltløkkelæring fungerer slik Argyis og Schön definerer det [1]

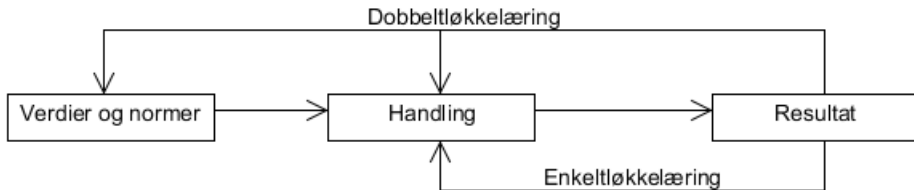
Enkeltløkkelæring går ut at det oppdages en feil eller avvik og deretter prøve å finne metode (handling) for å rette opp feilen eller avviket for å løse problemet, og utfallet av handlingen er resultatet. De underliggende årsakene til utfordringen undersøkes ikke, og derfor blir ikke verdiene og normene endret, se figur 3.2. Et praktisk eksempel ved bruk av enkeltløkkelæring er at det meldes inn en feil med produktet fra en kunde til utviklerne. Utviklerne finner feilen ved å gjennomføre testing (handling) og deretter retter opp feilen (resultat).

³Vanskeligheter med å finne passende terminologi på norsk på dette området har ført til at jeg bruker den engelske definisjonen.

3.5.2 Dobbeltløkkelæring

Dobbeltløkkelæring er en annen form for læring. Dobbeltløkkelæring innebærer at man undersøker underliggende årsaker til problemet. Dette innebærer at man undersøker hvilke antakelser eller forutsetninger (verdier og normer) som er grunnlaget for problemet og løse disse - med andre ord går man til roten på problemet istedet for å behandle symptomene på problemet. Argyis og Schön definerer dobbeltløkkelæring som følgende [1]⁴:

”By double loop learning, we mean learning that results in a change in the values of theory-in-use, as well as in its strategies and assumptions. The double loop refers to the two feedback loop that connect the observed effects of actions with strategies and values server by strategies. Strategies and assumptions may change concurrently with, or as a consequence of, change in values. Double loop learning may be carried out by individuals, when their inquiry leads to changes in the values of their theories-in-use or by organizations, when individuals inquire on behalf of an organization in such a way as to lead to change in the values of organizational theory-in-use.”



Figur 3.3: Viser hvordan dobbeltløkkelæring fungerer slik Argyis og Schön definerer det [1]

Dobbeltløkkelæring kan forklares ved å bruke samme eksemplet som ble brukt i enkeltløkkelæring. Kunden melder inn feil med programvaren til utviklerne. Utviklerne vil finne den underliggende årsaken (verdier og normer) til hvorfor feilen oppsto, istedet for å bare rette opp feilen. Gjennomføres det en årsaksanalyse (handling) for å undersøke hvorfor feilen oppsto, der kommer de fram til at manglende domenekunnskap (resultat) er den underliggende årsaken til problematikken. Det er nødvendig at utviklerne forstår hva som skal utvikles, hvis ikke vil det føre til gjentatte feil med programvaren.

⁴Vanskeligheter med å finne passende terminologi på norsk på dette området har ført til at jeg bruker den engelske definisjonen.

3.5.3 Bruksteori

Det er få bedrifter som klarer å mestre dobbeltløkkelæring påpeker Argyis og Schon. For å forstå dette skal vi se nærmere på bruksteori og hva dette innebærer. En bruksteori, også kalt *Theory-in-use* går ut på at et individ har dannet seg meninger og oppfatninger av virkeligheten basert på sine tidligere erfaringer, gjerne tidlig i livet [1]. Slike holdninger utløser forsvarsmekanisme for å bevare teoriene og motarbeider for å danne dobbeltløkkelæring. Argyis og Schon bruker to bruksteorier for å vise hvordan enkeltløkke- og dobbeltløkkelæring fremmes og hemmes. Disse bruksteoriene er presentert som modell I og modell II.

3.5.4 Model I

Argyis og Schön beskriver modell I ved hjelp av styringsvariabler [1]. Styringsvariabler har tilhørende handlingsstrategi, konsekvenser for arbeidsmiljø, effektivitet og læring. En handlingsstrategi er handlingene og planene som brukes for at folk holder sine styringsverdier innenfor det akseptable området. En konsekvens i denne sammenhengen er hva du føler deg forpliktet til å gjøre eller forhindret til å gjøre. Argyis og Schön har utarbeidet en tabell for model I, og som er gjengitt på tabell 3.5.

Å definere mål og prøve å oppnå dem er den første styringsvariabelen. Dette innebærer at aktøren sjelden prøver å utvikle samme forståelse av formålet og de er heller ikke åpen for å påvirke sin oppfatning av oppgaven. Handlingsstrategien er styre og konstruere omgivelsene ensidig slik at aktøren har kontroll over dem. Når det gjelder arbeidsmiljø, så fører dette til aktørens atferd kan bli sett som forsvarende, inkonsekvent, inkongruent, kontrollerende, engstelig for å bli sårbar, tilbakeholder følelser, altfor bekymret for seg selv og/eller for bekymret om andre. Når det gjelder konsekvensene for læring så er aktøren selvklebende, mens konsekvenser for effektivitet så reduserer det langvarig effektivitet.

Maksimere vinning og minimere tap er den andre styringsvariabelen. Aktører føler det er en svakhet å endre på mål som allerede er bestemt. Handlingsstrategien for denne styringsvariabelen er at aktøren ønsker å eie og ha kontroll over oppgaven. Dette fører til at aktøren vil at andre i teamet skal prøve å se ting på aktørens måter og det er lite hjelp å få. Når det gjelder konsekvenser for læring, så fører dette til enkeltløkkelæring.

Minimalisert generering eller uttrykke negative følelser er den tredje styringsvariabelen som aktøren i organisasjonen skal tilfredsstille. Handlingsstrategien er ensidig beskyttelse av seg selv ved å bruke defensiv handling som å skyld på andre, stereotyper og undertrykke følelser. Når det gjelder arbeidsmiljø vil dette endre seg til å forsvare normer i form av mistillit, mangel på ta risiko, konformitet, ekstern forpliktelser, vektlegger diplomati, maktsentret konkurranse og rivaliserende. I tillegg vil aktøren snakke abstrakt og unngår å nevne referanser for å hindre å bli sårbar. Aktøren vil holde tanker og følelser for seg selv, for å skape mysterium

rundt sine handlinger. Når det gjelder konsekvenser for effektivitet vil ikke aktøren teste sine teorier ”offentlig” og som er en hindring for dobbeltløkkelæring.

Være rasjonell er den siste styringsvariabelen som aktøren i en organisasjon skal tilfredsstillte. Denne styringsvariabelen er et motstykke til den tredje styringsvariabelen. Handlingsstrategien er ensidig å beskytte andre fra å bli såret. Å holde tilbake verdifull og viktig informasjon, fortelle hviteløgner, undertrykke følelser og vise falsk sympati er eksempler på denne handlingsstrategien. Det er de samme konsekvensene for arbeidsmiljø, effektivt og læring som den tredje styringsvariabelen.

Modell I viser til aktørene i organisasjonen har en motstridende situasjon, typisk vinn-tapssituasjon. Styringsverdiene i denne modellen hindrer at aktørene danner dobbeltløkkelæring på grunn av at aktørene er motstridende. Eksempelvis er det å holde tilbake viktig informasjon, fortelle hvite løgner og unngå konfrontasjoner for å hindre fra å bli såret, er motstridende situasjoner som hindrer dobbeltløkkelæring.

3.5.5 Modell II

Den andre modellen er modell II. I modell I var aktørene tilbakeholdene for å ikke utsette seg for sårbarheter og var kontrollerende ovenfor omgivelsene, mens i modell II derimot inviteres aktørene for å konfrontere sine synspunkter. I modell II vises det gjensidig respekt for hverandre og har friere valg og felles kontroll over oppgavene. Argyis og Schön har utarbeidet en tabell for model II, og som er gjengitt på tabell 3.6.

Modell II har tre styringsvariabler. Den første styringsvariabelen er *gyldig informasjon*. Dette involverer aktørene til å ta gjennomtenkte valg basert på gyldig informasjon, altså at aktørene gir korrekt informasjon som inkluderer at informasjon ikke holdes tilbake. Den andre styringsvariabelen er *frie og informert valg* går ut på at aktørene gir informasjonen om bakgrunnen for sine valg og aktørene står frie til å ta de valgene de selv ønsker. Den siste styringsvariabelen er *interne forpliktelser og konstant evaluering av handling i forhold til valget*. Denne styringsvariabelen involverer aktørene til å forplikte seg internt og reflektere jevnlig over sine handlinger.

Disse tre styringsvariablene har fire handlingsstrategier i Modell II. Den første handlingsstrategien er at deltakerne i teamet kan utøve høy grad av medbestemmelse. Å kontrollere oppgaven i fellesskap er den andre handlingsstrategien og dette gir større grad av innsyn på hva den enkelte aktøren holder på med i organisasjonen. Den tredje handlingsstrategien er å være orientert mot personlig vekst og faglig utvikling⁵. Den fjerde handlingsstrategien er at aktørene tar hensyn til egne og andre grenser samt har åpenhet for hvor disse grensene går.

⁵På engelsk: ”Protection of self is a joint enterprise and oriented towards growth”

Det er fire konsekvenser for arbeidsmiljø når det gjelder styringsvariablene i Modell II. Den første konsekvensen er aktørene betraktes som minimal defensiv. Den første henger sammen med den andre som påpeker at det er lite defensive mellommenneskelige og gruppedynamiske relasjoner. Ved at aktørene inviteres til å konfrontere hverandres synspunkter vil det føre til læringsorienterte normer. Den siste konsekvensen for arbeidsmiljø er at det er høy grad av valgfrihet, indre forpliktelser og risikotaking.

I den andre modellen er tre konsekvenser for læring. Siden aktørene er mindre defensive og har åpenhet, så er det oftere testing av teori i felleskap i teamet. Modell II krever at synspunktene skal formidles gjennom påstander som kan bekreftes av andre og dermed utelukkes selvoppyllende resonnementer fra Modell I. På bakgrunn av dette fremmes dobbeltløkkelæring både for tekniske og personlig utfordringer i teamet.

Økt langvarig effektivitet er den eneste konsekvensen for effektivitet i Modell II. Dette er på bakgrunn av det aktørene samarbeider i stedet for å konkurrere mot hverandre slik de gjorde i Modell I. Når grunnlaget for å danne dobbeltløkkelæring finner sted og det er ingen hindringer tilstede slik som i Modell II, så vil dette øke effektiviteten.

Modell I beskriver hvordan aktørene i bedriften mestrer enkeltløkkelæring og samtidig som den viser hvordan disse styringsvariablene hemmer dobbeltløkkelæring. I Modell II beskrives det hvordan en organisasjon kan mestrer dobbeltløkkelæring, og ifølge Arygis og Schön er det relativ få organisasjoner som mestrer dette. Arygis og Schön beskriver mer om utfordringene rundt organisatorisk læring, i boken "Organizational Learning II – Theory, Method and Practice"[1].

Styringsvariabel	Handlingsstrategi	Konsekvenser for arbeidsmiljø	Konsekvenser for læring og effektivitet
Definere mål og prøve å oppnå dem	Design og forvalte omgivelsene ensidig (være overbevisende, appellere til større mål)	Aktøren sett på som defensiv, inkonsekvent, kontrollerende, redd for å være sårbar, tilbakeholder følelser, altfor bekymret seg selv og andre, eller lite bekymret om andre.	Selvkleberende Reduserer effektivitet langvarig
Maksimere vinning og minimere tap	Eier og kontrollerer oppgaven (Gjør krav på eierskap av oppgaven, være veileder av definisjonen og gjennomføring av oppgaven.)	Defensive mellommenneskelige og grupperelasjoner (avhengig av aktør og lite hjelp som gis til andre)	Enkeltløkkelæring
Minimalisert eller uttrykke følelser	Beskytter seg selv ensidig (Uttaler seg abstrakt som følge av liten eller ingen observerende data, være blind for å bli påvirket av andre og uoverensstemmelse; bruker defensive handlinger som å skyld på andre, stereotyping, undertrykke følelser)	Defensive normer (mistilitt, mangel på risikotaking, konformitet, eksterne forpliktelser, vektlegging på diplomati, makt-senteret konkurranse og rivalisering).	Liten testing av teorier offentlig, Mye testing av teorier privat
Være rasjonelle	Beskytter andre fra å bli såret (holder tilbake informasjon, lager regler for å sensurere informasjon og adferd, holder private møter)		

Tabell 3.5: Model I bruksteori [1]

Styringsvariabel	Handlingsstrategi	Konsekvenser for arbeidsmiljø	Konsekvenser for læring	Konsekvenser for effektivitet
Gyldig informasjon	Uforme situasjoner hvor deltakere kan være opphavet til handlingen og opplever høy personlig kausaltitet	Aktøren opplever som minimal defensiv	Avkretfelsesprosess	Økt langvarig effektivitet.
Fritt og informert valg		Minimal defensiv mellommenneskelig relasjoner og gruppedynamikk	Dobbeltløkkelæring	
Interne forpliktelser og konstant evaluering av handling i forhold til valget	Oppgaven er felles kontrollert		Ofte offentlig testing av teorier.	
	Å være orientert mot personlig vekst og faglig utvikling	Læringsorienterte normer		
	Tosidig beskyttelse av andre	Høy grad av valgfrihet, interne forpliktelser og risikotaking		

Tabell 3.6: Model II bruksteori [1]

Kapittel 4

Distribuert og samlokalisert team

I dette kapitlet beskrives kommunikasjon og ressursbruk i forhold til samlokalisert og distribuert team.

Distribuerte team også kalt *virtuelle team* er geografiske distribuert i fysisk forskjellige arbeidssteder innen samme land eller distribuert over flere land, med enten samme eller forskjellig tidssoner [65], i forhold til samlokaliserte team der alle arbeider på ett arbeidssted til samme tid. Tabell 4.1 illustrerer grovt forholdet mellom geografisk avstand og tidsforskjell, der tilfelle A er karakterisert som samlokalisert og tilfelle C og D er karakterisert som distribuert team. I dette studiet fokuserer vi hovedsakelig på distribuerte team innenfor samme land og tidssone samt samlokaliserte team, tilfelle A og C i Tabell 4.1.

Tilfelle	Geografisk avstand	Tidsforskjell	Beskrivelse
A	Ingen	Ingen	Teamene arbeider samtidig i fysisk samme arbeidssted (samme land) med samme tidssone.
B	Ingen	Stor	Teamene arbeider skrift i fysisk samme arbeidsted (samme land) med samme tidssone.
C	Stor	Ingen	Teamene arbeider i samme land og/eller i flere land med samme tidssone.
D	Stor	Stor	Teamene arbeider i et globalt med flere land med flere tidssoner.

Tabell 4.1: Beskriver hvordan geografisk avstand og tidsforskjeller påvirker team-sammensetningen, basert på informasjon fra [2].

Distribuerte team kan brukes for å få tilgang til nødvendig kunnskap, erfaring, ressurser, ekspertise, kontinuerlig utvikling og spare kostnader [64, 65, 66]. I samlokaliserte team har deltakere større kjennskap til hverandre, samhandler i større grad med hverandre, koordinerer arbeidet gjennom spontane interaksjoner på bedriften [67] og tillitt opparbeides raskt i teamet [64].

Det er flere utfordringer tilknyttet distribuerte team avhengig av dens geografiske spredning - alt fra språklig, kulturelle, geografiske forskjeller til kommunikasjonsutfordringer mellom distribuerte team og kunderelasjon [65, 67, 68]. Det kan nevnes at Ågerfalk et al. utviklet et rammeverk for å redusere utfordringene knyttet til distribuert utvikling. Rammeverket inneholdt tre faktorer som tidssone, geografisk avstand og sosiokulturell utfordring. Disse faktorene ble beskrevet i forhold til tre kategorier: kommunikasjon, koordinering og kontroll [69]. Flere studier har vist til at kommunikasjonsutfordringer kan være grunnlaget for å danne andre utfordringer, eksempelvis observerte Mishra et al. at dårligere kommunikasjon fører til et dårligere samarbeid og koordinering for innad i utviklingsteamet [18]. Med dette i bakhodet kan kommunikasjon spille en vesentlig rolle for at distribuert utvikling skal lykkes med et prosjekt eller en utgivelse av ny versjon av en programvare [12]. Herbsleb og Mockus beskrev at minimal kommunikasjon mellom distribuerte teammedlemmene var en av årsakene til at det førte til forsinkelser i distribuerte team [13].

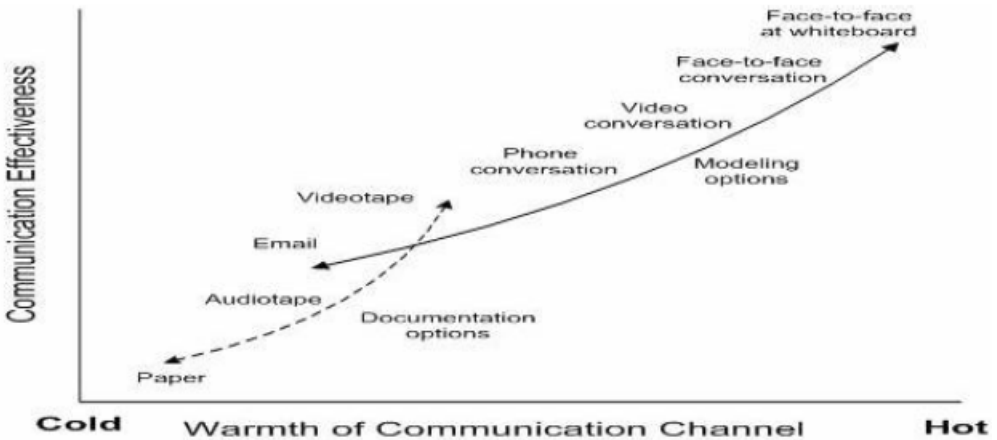
4.1 Kommunikasjon

I dette delkapitlet beskrives kommunikasjonsteori. Det første underkapitlet beskriver om innholdsrik kommunikasjon, deretter presenteres begrepene formell og uformell kommunikasjon og kommunikasjonsmuligheter relatert til fysisk avstand. I tredje underkapittel forklares relasjonen mellom gruppebevissthet og kommunikasjon. I fjerde underkapittel presenterer at ulike personer har ulike fortolkninger basert på deres historikk og i femte underkapittel presenteres sammenhengen mellom koordinering og kommunikasjon. Tilslutt beskrives hvordan kommunikasjonen er tilrettelagt i smidig utviklingsmetodikk.

I følge Mishra et al. er kommunikasjon definert som en formidling eller veksling av tanker, meninger eller informasjon via tale, skrift eller tegn [18]. Deltakerne i et team har behov for å kommunisere med hverandre og andre nødvendige personer for å gjennomføre sine arbeidsoppgaver [3]. Denne kommunikasjonen kan gjerne være formell eller uformell og verbal eller ikke-verbal kommunikasjon.

4.1.1 Innholdsrik kommunikasjon

Det finnes flere kommunikasjonsformer for å kommunisere, men innholdsriheten¹ i kommunikasjonsformene varierer. Den mest innholdsrike formen for kommunikasjon er ansikt-til-ansikt kommunikasjon ved bruk av verktøy som whiteboard for å konkretisere tanker og idéene [18]. Det er flere studier som viser at ansikt-til-ansikt kommunikasjon er den mest innholdsrike formen for kommunikasjon som brukes daglig, da den konstant legger til rette for tilbakemelding både visuelt og auditivt [8, 18, 70, 71]. Kunnskap tilegnet gjennom ansikt-til-ansikt kommunikasjon kan bli husket for en begrenset periode, og for at informasjonen skal bli langvarig bør det brukes sammen med andre verktøy som whiteboard og ark for å lagre informasjonen som skal brukes i fremtiden [18]. Videokonferanse er mindre innholdsrik i forhold til ansikt-til-ansikt kommunikasjon, da det begrenser det visuelle bildet, men betydelig bedre enn en telefonkonferanse [8]. En telefonkonferanse har ingen visuell informasjon, men har auditivt informasjon og er bedre enn en epost [8]. De ulike kommunikasjonsformene er illustrert på figur 4.1 og hvilket kommunikasjonsform som brukes er gjerne avhengig av dens formål, geografisk avstand og tidsforskjeller [2, 8], eksempelvis kan det vises til at sannsynligheten for å ha mest mulig spontan kommunikasjon er avtagende relatert til den geografiske avstanden [2].



Figur 4.1: Beskriver hvor innholdsrik kommunikasjonsformen er, der "cold" viser at innholdet er fattig i motsetning til "hot" som viser at innholdet er rikt samt hvor effektiv kommunikasjonen er. [8]

¹Engelsk: Information richness

4.1.2 Formaliteten på kommunikasjon og fysisk avstand

Kommunikasjonen kan være formell eller uformell kommunikasjon. En beskrivelse og sammenligning av formell og uformell kommunikasjonen er gitt på tabell 4.2.

Formell kommunikasjon	Uformell kommunikasjon
Planlagt på forhånd	Uforventet
Arrangerte deltakere	Tilfeldige deltakere
Deltakerne har en bestemt rolle	Deltakerne har ikke en bestemt rolle
Enveis	Interaktiv
Improvisert innhold	Innholdsrik
Formell språk	Uformell språk

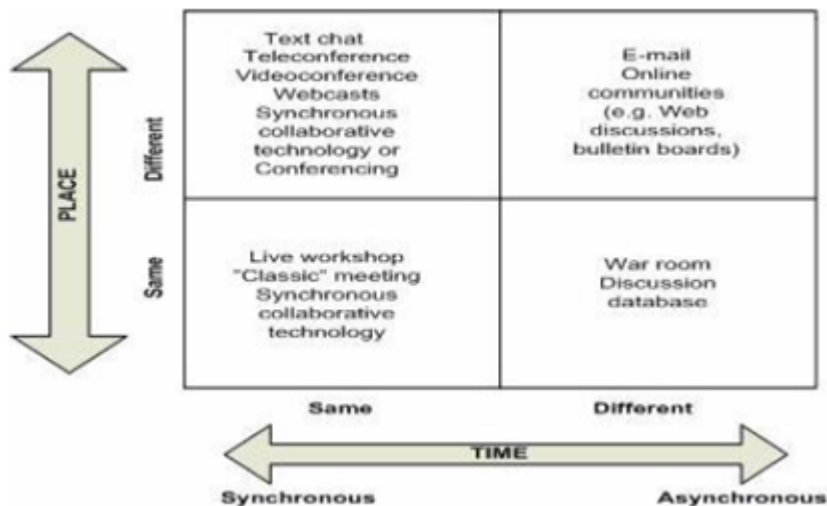
Tabell 4.2: Beskriver og sammenligner formell og uformell kommunikasjon [3].

Uformell kommunikasjon er spontan, interaktiv og innholdsrik. Denne type kommunikasjon skjer uforventet der deltakerne, temaet og stedet for samtalen ikke er planlagt på forhånd. Kraut et al. deler kommunikasjon inn i fire kategori, etter i hvilken grad kommunikasjon oppstår spontan; planlagt, uforventet, opportunist og spontan [3]. De tre sistnevnte beskrives som uformell kommunikasjon, mens førstnevnte kan anses som formell kommunikasjon. Kraut et al. har gjennomført observasjoner på samtaler i en organisasjon ved gå gjennomgå alle områder i organisasjonen én gang og deretter gjennomførte deltakerne i samtale en spørreundersøkelse. De fant at ni av ti tilfeller var kommunikasjonen uformell, der ansikt-til-ansikt kommunikasjon var mest utbredt. I tillegg rapporteres det at tidsbruken på samtale er avtagende etter spontanitet i kommunikasjonen, der spontane samtalsvarighet var under en tredjedel i forhold til planlagte samtaler.

Den fysiske avstanden har betydelig påvirkningskraft i forhold til kommunikasjonen, spesielt uformell kommunikasjon. Kraut et al. anbefaler at organisering av arbeidsplasser i bedriften skjer med tanke på kollegaer som sitter nærme hverandre kan senke kommunikasjonskostnader, særlig gjelder dette ansatte med beslutningsansvar [3]. Når ledere tar beslutninger, ønsker de gjerne å ta en prat med andre kollegaer for å luften idéer med dem. De ansatte avgrensner sin kontakt med de som sitter fysisk nærmest seg, og i disse fysiske omgivelsene foregår det meste av den spontane kommunikasjonen, spesielt gjelder dette samlokaliserte team [3]. O'Leary og Cummings fastslo at kortere geografisk avstand førte til mer kommunikasjon mellom teamene [2].

Mishra et al. beskriver at fysisk åpen kontorlandskap hjelper med å forbedre kommunikasjon, koordinering og samarbeid med sine medarbeidere. I tillegg viser funnene at individer som jobber i samme team bør bli plassert ved siden av hverandre slik at de kan samhandle med hverandre når det er nødvendig for avklaring og dermed effektivisere samarbeidet og fremdriften i prosjektet. Konsekvensen ved å plassere individene på feil sted kan føre til lavere produktivitet og dårligere kvalitet på produktet. Det bør brukes halvhøye glassbarrierer mellom hver arbeidspult

slik at utviklerne kan være synlige overfor hverandre, og dermed blir det enklere å få tak i hverandre som fører til at kommunikasjonsfrekvensen øker [18].



Figur 4.2: Kommunikasjonsmuligheter avhengig av tid og sted [8]

Det brukes forskjellige former for kommunikasjon i et distribuert team i forhold til et samlokalisert team. Kommunikasjonsformen som brukes er sterkt preget av geografisk avstand og tidsforskjeller [2]. Figur 4.2 viser åpenbart at synkroniserte samlokaliserte team har åpenbart flere muligheter til den mest innholdsrike kommunikasjonsformen, ansikt-til-ansikt i forhold til distribuerte team [18]. I distribuerte team kan videokonferanse brukes for å få det visuelle aspektet som kroppsspråk, samtidig som whiteboard brukes for å konkretisere tanker og ideér. Det er begrenset hvor mye av det visuelle aspektet som kan gis ved bruk videokonferanse [70].

Kommunikasjon er et verktøy for at samarbeidet skal fungere godt i et virtuelt team. Distribuerte team har begrenset muligheter til å ha ansikt-til-ansikt kommunikasjon i en fysisk omgivelse, da fysiske avstanden mellom teamene er stor [2]. I tilfeller distribuerte team ønsker å ha ansikt-til-ansikt kommunikasjon i en fysisk omgivelse ved å besøke hverandre gir dette mer innholdsrik kommunikasjon i forhold til å bruke virtuelle virkemidler som videokonferanse [8], men denne prosessen er tidskrevende og reisekostnadene kan være høy avhengig av den geografiske avstanden [71]. Ved daglig reise mellom distribuerte team anbefaler French og Layzell at avstanden bør være korte, slik kommunikasjonskostnadene er lave [71]. O'Leary og Cummings undersøkte kommunikasjonen til flere distribuerte team og fant at kommunikasjonsfrekvensen var avtagende relatert til spredningen av teamet [2]. I samlokaliserte team har deltakerne muligheten til å se hvem som er tilgjengelig og dermed raskt invitere hverandre til ansikt-til-ansikt kommunikasjon [72].

4.1.3 Bevissthet og kommunikasjon

Kommunikasjonen trenger ikke å være verbal for å kunne fungere godt mellom deltakerne i et team som er distribuert. I disse tilfellene er bevissthet² en avgjørende mekanisme for å hindre kommunikasjonsutfordringer. Gutwin et. al har undersøkt hvordan kommunikasjonsutfordringene i tre store opensource prosjektene har blitt løst på tross av at de i større grad er distribuert og at kommunikasjonen hovedsakelig er tekstbaserte [52]. Funnene fra studiet viser at gruppebevissthet³ er nøkkelen for å løse kommunikasjonsutfordringene i et opensource prosjekt som er distribuert. Dette innebærer at utviklerne har detaljert kunnskap om hvem som arbeider i prosjektet, hvor i koden de arbeider, hva de gjør og hva de planlegger å gjøre. Denne kunnskapen er nødvendig for at flere skal arbeide med de samme oppgavene og at det skal være enklere å koordinere prosjektet. Denne detaljerte kunnskapen hentes gjennom tre kommunikasjonsformer som epost, tekstchat og commits. Mesteparten av kommunikasjonen var offentlig for alle involverte i prosjektet, selv om samtalen foregikk mellom to utviklere. På denne måten kan enhver utviklerne oppdatere seg på hva de andre utviklerne i prosjektet holdt på med til enhver tid. Den formelle kommunikasjonen var å sende eposter til en felles epostadresse som var tilgjengelig for alle involverte i prosjektet, og mesteparten av kommunikasjonen foregikk på denne måten. Den uformelle kommunikasjonen foregikk gjennom tekstchat, der ble arbeidsrelaterte problemstillinger diskutert og en oppsummering av samtalen ble sendt til felles epostadressen. Gjennom commits ble det formidlet hvor i koden de enkelte deltakerne var aktiv, men det var tidskrevende å gå gjennom hele commitsloggen for å få nærmere beskrivelse av utviklet funksjonalitet. En kortversjonen av utviklet funksjonalitet ble også sendt til felles epostadressen, der ble hele informasjonen beskrevet i emnetittelen for å gjøre det mindre krevende å følge med. Gutwin et. al beskriver ytterligere hvordan opensouce utviklere kommuniserer for å komme i kontakt med de rette utviklerne og utfordringene rundt denne problematikken. Opensource utvikling er et eksempel på at kommunikasjonen trenger ikke å være ansikt-til-ansikt for at kommunikasjonen skal fungere godt mellom personer som deltar i et prosjekt over lengre geografisk avstand.

4.1.4 Historikk og fortolkning

Kommunikasjon brukes til å formidle informasjon mellom deltakerne i et team. Star og Griesemer understreker at forskjellige personer har ulik historikk som fører til at ulike personer tolker informasjonen i lys av denne bakgrunnen og likevel klarer personene å samarbeide med hverandre [73]. Charlie beskriver at deltakerne har ulik historikk, og derfor vil det oppstå begrensinger i forhold til å ha felles syntaks. Han poengterer viktigheten av å anerkjenne at fortolkende forskjeller ofte eksisterer [74]. Derfor argumenterer Figueiredo et al. at forskjellige interensenter trenger å bruke de samme begrepene for å være sikre på at de forstår hverandre [75]. På en annen side beskriver Melo et al. behovet for å ha samme forståelse når det gjelder gjennomføring av arbeidsoppgaver, hvis ikke kan det påvirke produktiviteten [9].

²Engelsk: Awareness

³Engelsk: Group Awareness

4.1.5 Kommunikasjon og koordinering

Det trengs kommunikasjon for å koordinere teamene på tvers av lokasjonene. Dingsøy et al. argumenterer for at koordinering av programvareprosjekter krever interaksjon mellom deltakerne i teamet. Denne interaksjonen kan foregå hjelp av kommunikasjon [76]. Epsinosa et al. viser til at flere empiriske studier har funnet at betydelig mengde koordinering i programvareutvikling foregår gjennom uformelle samtaler og møter på offentlige møteplasser der kaffetrakter eller vannkjøleren er plassert i bedriftene [67]. En studie gjennomført av Kraut et al. viser til at verbal interaksjon utgjør 75 % av en prosjektleders arbeidsdag, der halvparten av dette er uforventet ansikt-til-ansikt kommunikasjon og 12 % er telefonsamtaler som ikke var planlagt [3]. Med dette i bakhodet kan det være vanskelig for distribuerte team å gjennomføre ansikt-til-ansikt kommunikasjon.

Moe et al. har funnet at når deltakerne får tildelt eget programvaremodul, så er de ikke interesserte for å høre hva de andre deltakerne i teamet holder med og deres status på arbeidet [77]. Herbles har funnet at det hjelper ikke med å tildele programvaremoduler til bestemte lokasjoner i et distribuert team for å hindre avhengigheter på tvers av lokasjonene. Han utdyper at det er nødvendig å kommunisere på tvers av lokasjonene for å gjennomføre oppgavene godt nok [78]. Drury et al. gjennomførte en fokusgruppe og fant at prioritering av arbeidsoppgaver var en utfordring for systemutviklingsteam [62].

4.1.6 Kommunikasjon i smidig utviklingsmetodikk

I smidig utviklingsmiljø er effektiv kommunikasjon viktig for at utviklingsteamet legger til rette for raskt kunnskapsdeling mellom deltakerne. Effektiv kommunikasjon trengs for å forstå kravspesifikasjon fra kunden og hjelpe andre deltakere for å utføre aktiviteter tilknyttet utvikling [70]. Anderson et al. gjennomførte laboratoriebaseret undersøkelse på virtuelle møter og funnene viser at økt kommunikasjon på det ene møtet fører ikke til kommunikasjonen øker på det andre teamet de har videokonferanse med [79]. I tilfeller der organisasjon ønsker åpen kommunikasjon i et virtuelt møte, trengs det forberedelse og trening slik at samarbeidet blir godt gjennomført, og det oppfordres til å bruke fasilitator som oppmuntrer til åpen kommunikasjon på tvers av distribuerte team. Den store fordelen med direkte interaksjoner mellom teammedlemmer på tvers av distribuerte team er at den øker den fellesforståelsen av problemstillingen [79]. Korkala et al. understreker at deres observasjoner støtter argumentet om at smidig praksiser er de mest effektive i et miljø der effektiv kommunikasjon er aktiv. Bakgrunnen for dette er at effektiv kommunikasjon nedskalere tidsbruken på å ta store beslutninger og avgjørelser [12]. Smidig møter som daglig scrum hjelper med å holde framdriften i et distribuert utviklingsprosjekt, og planleggingsmøtene er et godt grunnlag for å unngå misforståelser om hva som skal utvikles [80]. Denne forståelsen hjelper med å hindre situasjonen som er illustrert på figur 2.2.

4.2 Ressursbruk

Ressursbruk er en betegnelse på hvor mye tid og kostnader som brukes for å gjennomføre et prosjekt. Herbsleb og Mockus sine funn viser at distribuerte team bruker 2,5 ganger mer tid sammenlignet med samlokalisert team for å gjennomføre de omtrent de samme oppgavene. Hovedårsaken til disse forsinkelsene i distribuert team er utfordringer knyttet til kommunikasjon og koordinering. Når gjelder koordineringsutfordringene, så kreves det at stort antall deltakere involveres seg når distribuerte endringer forekommer og dette fører til forsinkelser. Enkelte teammedlemmer kommuniserer med deres distribuerte teammedlemmer mindre en gang i uka i forhold til deres lokale teammedlemmer der kommunikasjonen foregår flere ganger daglig. Herbsleb og Mockus poengterer at forståelsen om gjennomføringen av arbeidsoppgavene skapes gjennom kommunikasjon, slik at vi unngår at oppgaven blir halvferdig og ekstra ressurser skal brukes for å fullføre oppgaven. [13]. Deltakerne i distribuerte team beskriver at det er vanskelig å finne rett person med rett ekspertise for å løse arbeidsoppgavene og at oppgaven gis til en deltaker som har deler av ekspertisen som kreves og som fører til forsinkelse [13].

Det kan tenkes at distribuerte team ønsker å møte hverandre fysisk for å skape en sosialsamhold og fortrolighet på tvers av teamene [3, 64]. Logisk sett, kan distribuerte team som ønsker fysisk tilstedeværende møter bidra til å øke kostnader i form av reiseutgifter. Webster og Wong gjennomført en spørreundersøkelse for å sammenligne virtuelle team med tradisjonelle (samlokalisert) team og funnene viser at overraskende nok at distribuerte team var mer tilfredsstillende med deres prosjekt i forhold til samlokalisert team. Dette skyldes tilgang til distribuert ekspertise, mer frihet til koordinere sitt arbeid og en bedre balanse mellom arbeid og familieliv [64].

Kapittel 5

Metode

I dette kapitlet begrunnes valget av vitenskapelige metoder og casene som er brukt i dette studiet. Først beskrives metodevalgene og deretter de ulike casene og be-driftene som har deltatt i dette studiet.

5.1 Valg av forskningsmetode

Det benyttes kvalitativt studie for å gjennomføre dette studiet på bakgrunn av at dette gir tilgang til et rikt og detaljert informasjon som ikke bare omfatter tall, men også visuelle og auditive informasjon [81]. Denne informasjonen kan brukes til å finne underliggende årsaker og motivasjon til handlinger og valg. Denne dybde informasjonen er vanskelig å få gjennom et kvantitativt studie, og derfor velges et kvalitativt studie. Spørreundersøkelser som er et kvantitativt metode gi stor bredde for kommunikasjonsutfordringer mellom distribuerte team, men det er vanskelig å finne underliggende årsakene og se hendelser i sammenheng ved å bruke denne forskningsmetoden. Derfor er ikke kvantitativt metoder egnet for forskningsformålet.

Case studie ble valgt som forskningsmetode da jeg har muligheten til å ut-forske læringseffekten og kommunikasjonsutfordringene ved gå inn situasjoner og undersøke disse fra innsiden av casene (prosjektene). På denne måten får jeg de-taljrisk informasjon på utvalgte casene som brukes for å forklare sammenhengen mellom ulike handlinger og årsaken til disse. [81]

Det er valgt case studie av typen *Multiple holistic explanatory study* for å svare på forskningsspørsmålet. *Multiple* er valgt fordi forskningen krever at det er minst et case som er samlokalisert og et case som er distribuert for å kunne sammenligne disse med hverandre, noe som ikke lar seg gjøre med et enkelt case. *Holistic study* er valgt fordi studiet fokuserer på innhenting av kompleks informasjon, der retro-spektivpraksisen blir studert som helhet i stedet for å fokusere på enkelt faktorer. *Explanatory study* er valgt for å undersøke hvilke underliggende faktorer som forkla-

rer hva som skjer i retrospektivmøter, særlig i forhold til læringseffekten i møtene. I tillegg er studiet *longitudinal study* da studiet har som hensikt å undersøke lærings-effekten over tid i retrospektivmøter, og i tillegg se hvordan utfordringene utvikler seg [81].

5.2 Forskningsdesign

Formålet med forskningen har endret seg radikalt siden studiet begynte i høsten 2015. Den opprinnelige planen var å gjennomføre ett studie på bruk av verktøy i distribuerte retrospektivmøter og sammenligne utfordringer knyttet til samlokaliserte team med distribuerte team. I tillegg til dette var formålet å undersøke organisatorisk læring i retrospektivmøter, der det skulle sammenlignes mellom læringen som foregår i distribuert retrospektivmøter med samlokalisert retrospektivmøter.

Det er gjennomført en observasjon angående bruk av aktuell verktøy i distribuerte team. En konsekvens av begrenset datamaterial har ført til at bruken av verktøy i distribuerte retrospektivmøter er ikke en del av det endelige resultatet av forskningen. Resultatet av arbeidet som er gjennomført i forhold til dette formålet er vedlagt i vedlegg A. Vedlegget omfatter undersøkelse av 18 forskjellige verktøy for å se hvilke verktøy som oppfyller kravene til å kunne bruke i distribuert retrospektivmøter.

Når det gjelder å sammenligne utfordringer i distribuerte team med samlokaliserte team så er antall observasjoner ujevnt fordelt. Det betyr at det er langt flere observasjoner av samlokaliserte team i forhold til distribuerte team, noe som ikke var den opprinnelige planen. Dette har ført til det er en skjevfordeling i resultatet av forskningen, der samlokalisert team har større tyngde på datamaterialet i forhold til distribuert team når disse sammenlignes. Når det gjelder å undersøke organisatorisk læring i forhold til retrospektivmøter er ikke dette formålet påvirket av endringene i underveis i studiet, bortsett fra at det ikke sammenlignes mellom læringsutbytte i samlokalisert team med distribuert team.

Den opprinnelige planen var å ha tre caser, der to av casene var distribuerte team og ett samlokalisert team. Et av distribuerte casene skulle bruke et verktøy jeg har undersøkt, mens det andre teamet skal være et ordinært distribuert team, der jeg skulle undersøke utfordringene knyttet til distribuerte team. I løpet av studiet har dette endret seg til å ha et stort case som er samlokalisert og to små case som er distribuert, der kun ett av dem benyttes aktivt i studiet.

En samlet implikasjon av endringene som har skjedd i løpet av studiet har ført til at det samlokaliserte caset har hovedtyngden på datamaterialet i forhold til de distribuerte casene. Konsekvensen av dette har vært at forskningen om kommunikasjon som primært fokuserer på distribuert team har blitt til ett supplement i forhold til organisatorisk læring. Dette er årsaken til at det hovedsaklig fokuseres på organisatorisk læring i denne avhandlingen.

5.3 Bakgrunn for casestudiet

Det er tre caser i dette casestudiet, der to av dem er distribuerte og ett er samlokalisert. De distribuerte casene tilhører samme bedrift, heretter kalt bedrift A, mens det samlokalisert caset tilhører en annen bedrift, bedrift B. Først beskrives bedriftene før casene presenteres. Tilslutt begrunnes hvorfor casene ble valgt og det gis en oppsummering og sammenligning av casene.

5.3.1 Bedriftene

Bedrift A

Bedrift A er et ansatteid selskap med over 100 IT-konsulenter som leverer sakbehandling-løsninger til norske virksomheter, hovedsakelig til offentlig sektor. Selskapet er et konsulent selskap, noe som betyr at de utvikler programvareløsninger for kundene sine. Virksomheten hadde en omsetning over 100 millioner norske kroner i 2014, noe som betyr at selskapet kan klassifiseres som et middelsstort selskap. Etter en lang tid med tradisjonelle prosjektmetodikk foretrekker de mer smidige metoder som Scrum og Kanban. Da smidige metoder gir fleksibilitet, enkelthet og bedre løsninger. Selskapet har to kontorer i Norge, som heretter kalles by1 og by2.

Bedrift B

Bedriften har underkant av 100 ansatte som omfatter systemutviklere og andre fagpersonell. Virksomheten har spesialisert seg mot ett fagområde og har utviklet sine egne produktet innenfor dette fagområdet som selges til sine kunder. Utviklerne i bedriften har tett samarbeid med fagpersonellet innen fagområdet for å utvikle produktet, noe som betyr at utviklingsteamet har tverrfaglig kompetanse. Deres programvarer brukes av over ti tusen kunder og fagpersoner. Virksomheten har en omsetning over 100 millioner norske kroner i 2014, noe som betyr at selskapet kan klassifiseres som middelsstort selskap. Bedriften utvikler programvareproduktene sine etter smidig metodikk, nærmere bestemt etter Scrum-metodikken.

5.3.2 Casene

Case D1

Dette caset følges fra oppstart av prosjektet, nærmere bestemt fra andre retrospektivmøtet og arbeider i bedrift A. Deltakerne i dette teamet arbeider distribuert mellom by1 og by2, mens en av deltakerne oppholder seg i by3 og som er det samme stedet som kunden oppholder seg. Det selvorganiserte består av 11 deltakere, der fem arbeider fra by1, fem fra by2 og en fra by3. Deltakerne er stort sett erfarne utviklere og har god kompetanse innenfor fagfeltet. Teamet bruker Scrum-metodikken for å gjennomføre prosjekt med fire ukers sprinter, men denne sprintlengden har senere blitt endret under studiet. Retrospektivmøtet gjennomføres med og uten

kunden, men studiet følger kun retrospektivmøte uten kunden. Når teamet har retrospektivmøtet med kunden, reiser hele teamet til by3 der både sprint review og retrospektivmøter gjennomføres.

Teamet følger Scrum-metodikken som er beskrevet i vedlegg B, men samtidig har de tilpasset den til sitt eget formål. De to lokasjonene, by1 og by2 har hver sin squadmaster som har ansvaret for å hindre avhengigheter mellom hverandre når en utviklingsoppgave skal gjennomføres. I tillegg har de fått hovedansvaret for testingen. Squadmaster er heretter referert som scrum-master da jeg ikke skiller mellom disse to rollene. De to lokasjonene reiser til hverandre for å arbeide tettere sammen dersom det er behov og har økonomisk støtte for det. De to distribuerte teamene, by1 og by2 bruker samme produktkø, sprintkø og burndownchart.

Dette caset har også blitt fulgt av Tina Christin Syversen som skrev egen masteroppgave om suksessfaktorer og koordinering på tvers distribuerte team. I denne anledningen har hun observert to sprintplanleggingsmøter og to backloggromming møter som jeg har fått tilgang til. Dette materialet jeg har brukt for å verifisere at deltakerne har fulgt forbedringstiltakene fra retrospektivmøtene og utfordringer i forhold til kommunikasjon på tvers av distribuerte team. Det understrekes at ingen utdrag fra disse observasjonene er gjengitt i denne oppgaven.

Case D2

Deltakerne i dette caset arbeider distribuert mellom by1 og by2 og er ansatt i bedrift A. Deltakerne i dette caset skulle opprinnelige undersøke bruken av verktøy i distribuert retrospektivmøte. Slik det er beskrevet innledningvis så har jeg ikke tilstrekkelig material til å kunne gjennomføre dette, derfor er dette caset lagt til side, men det brukes som ett støtte case når det kun gjelder kommunikasjonsutfordringer i distribuerte team.

Det er relativt begrenset informasjon om caset, da det er kun gjennomført en observasjon av retrospektivmøtet. Når observasjonen ble gjennomført har deltakerne kommet halvveis i prosjektet. Teamet består utelukkende av systemutviklere som er fordelt mellom to lokasjoner, der fire utviklere arbeider i by1 og seks utviklere i by2. Disse utviklerne har ulik erfaring, alt fra studenter som arbeider deltid til senior utviklere. Kunden er en privat aktør som oppholder seg i by2. Det brukes Scrum-metodikken for å utvikle programvaren med en sprintlengde på fire uker, se vedlegg B. De to lokasjonene reiser til hverandre for å kunne arbeide tettere sammen dersom det er for behov og har økonomisk støtte for det.

Case S

Deltakerne i dette teamet arbeider samlokalisert hos bedrift B. Teamet har kontinuerlig utgivelser (releaser) av programvaren som utvikles. Teamet består av ni deltakere som har ulik fagkompetanse og erfaring, der fem deltakere er systemutviklere (inkludert testere) og resten er fagpersonell. Deltakerne i teamet har bred

erfaring innen systemutvikling, mens ansienniteten i prosjektet varierer på deltakerne i teamet. Derfor finnes deltakere som nylig var ansatt i prosjektet, og disse er relativt ferske på domenet. Underveis i studiet har teamet mistet en nøkkeldeltaker i studiet. Teamet arbeider på fem forskjellige produkter og har planlagt å komme med ett nytt produkt i løpet av neste kalenderår. Produktene som utvikles blir solgt til flere kunder, og har til forskjell fra tradisjonelle bedrifter flere kunder å forholde seg til. Teamet skal på oppfordring fra kundene lage en tilpasset versjon av produktet for deres bruk.

Teamet bruker Scrum-metodikken for å utvikle programvareproduktene sine med en varighet på to-ukers sprints, se vedlegg B. Retrospektivmøtene gjennomføres med prosjektleder tilstede. Dette kan hjelpe teamet med å ha et flerperspektiv retrospektivmøte, der både fagpersonell, systemutviklere og prosjektleder er samlet. Retrospektivmøtet foregår annenhver uke etter at en sprint er avsluttet.

Oppsummering og begrunnelse for valg av case

Kategori	Case D1	Case D2	Case S
Team	Distribuert	Distribuert	Samlokalisert
Bedrift	A	A	B
Virksomhet	Konsulent	Konsulent	In-house
Teamstørrelse	11 deltakere	10 deltakere	9 deltakere
Tverrfaglighet	Nei	Nei	Ja
Antall produkter som utvikles	Ett	Ett	Fem
Kundesektor	Offentlig	Privat	Privat og offentlig
Antall kunder	Én kunde	Én kunde	Flere kunder
Metodikk	Scrum	Scrum	Scrum
Sprintlengde	4 uker	4 uker	2 uker
Retrospektiv	Både med og uten kunden	Uten kunde	Uten kunde
Bruk av verktøy	Ikke i begynnelsen, men senere i studiet	Ja	Nei

Tabell 5.1: En sammenligning av casene i studiet

En sammenligning og oppsummering av casene er gjengitt i tabell 5.1. Det kan nevnes at jeg har fått tilgang til disse casene gjennom SINTEF, nærmere bestemt av Torgeir Dingsøy som har kjennskap til disse bedriftene. Derfor har det vært enklere for å selge meg inn i disse casene. Disse casene oppfyller kravene som stilles for å kunne svare på forskningsspørsmålene. Det opprinnelig formålet med forskningen var å kunne sammenligne samlokalisert retrospektiv med distribuert retrospektiv samt bruk av verktøy i distribuert retrospektivmøte. For å kunne sammenligne samlokalisert team med distribuert team er det nødvendig å ha minst ett case som er samlokalisert og ett som er distribuert. Disse casene bør samtidig bruke

smidig utviklingsmetodikk og gjennomføre retrospektivmøter etter ”boka”. Dette er bakgrunnen for at case D1 og case S er valgt for å gjennomføre studiet. Dette kravet tilfredsstiller også case D2 som i utgangspunktet var åpen for å undersøke bruken av verktøy i distribuerte retrospektivmøter.

5.4 Datainnsamling

I dette studiet er de brukt observasjoner og uformelle intervjuer som datainnsamlingsmetode. Observasjonen som datainnsamlingsmetode vil kunne gi meg informasjon om deltakernes holdning og adferd i form av kroppsspråk samt gi innblikk i atmosfæren blant deltakerne i teamet. Denne informasjonen er nyttig for å kunne se læringen i sammenheng med adferden i et retrospektivmøte. Jeg som observatør kan komme bort i situasjoner der det er vanskelig å forstå intensjonen, meningen eller grunnen til at oppførselen oppstår [81]. Derfor ble det gjennomført en tilbakemeldingssesjon med ett av temaene, der jeg ga tilbakemelding, samtidig som jeg fikk oppklart uklarheter som oppsto under observasjonene fra retrospektivmøtene. Et semi-strukturert intervju gir enda dypere forståelse fra samtidige deltakere i studiet som kan brukes til å bekrefte funnene i observasjonen [81], men dette har ikke blitt gjennomført.

5.4.1 Observasjon

Det finnes ulike observasjonstyper, men *fullstendig observatør*¹ ble benyttet for å gjennomføre datainnsamlingen. Dette betyr i praksis at jeg er en tilskuer som observerer det som foregår i retrospektivmøtet og noterer handlingen og tidspunktet for handlingen uten å ta aktiv del i retrospektivmøtet [81]. Jeg plasserte meg unna teamet slik at jeg ikke ble oppfattet som en deltaker i møtet, men som en observatør.

Det ble ikke brukt bærbar PC eller andre elektroniske maskinvarer for å notere observasjonene fordi det kan forstyrre deltakerne og dermed påvirke resultatet av observasjonene. I stedet ble det brukt penn og papir (kladdebok) for å notere ned observasjonene. Samme dag som observasjonen ble gjennomført ble observasjonsnotatene fra kladdeboka overført til PCen – slik at det var enklere for å gjennomføre dataanalysen. Jeg bestemte meg først for å ikke ta lydopptak av retrospektivmøtene, men senere valgte jeg å bruke lydopptak. Dette var fordi jeg mistet nødvendig informasjon ved å kun notere handlingene jeg fikk med meg. Derfor ble det spurt om tillatelse for bruk av lydopptak fra samtidige deltakere i studiet. Det ble brukt mobiltelefon til å ta lydopptak og som senere ble transkribert ved bruk av Express Scribe Transcription eller Windows Media Player. I praksis fungerte transkriberingen ved å skrive ned ord for ord hva deltakerne har sagt i retrospektivmøtet, noe førte til at det ble brukt åtte til ti timer for å transkribere et retrospektivmøte med varighet på to timer. Ved å gjøre det på denne måten får jeg tilgang til mer innholdsrik datamateriale i forhold til å bare notere det jeg klarer

¹Engelsk: Complete observer

å få med meg under møtene, og se tabell 5.2 for hvilke møter som ble tatt opp på lyd.

Case D1	Case S
Møte 2	Møte 3
	Møte 4
	Møte 5
	Møte 6
	Møte 7

Tabell 5.2: Lydopptak av retrospektivmøtene

Jeg hadde ingen forhåndsdefinerte tema som skulle fokuseres på under observasjonen, og derfor ble all observasjon notert på kladdeboka med nøyaktig tidspunkt (time og minutt). Jeg fikk tildelt bakgrunnsinformasjonen om teamene før observasjonsstudiet av teamene begynte. Denne bakgrunnsinformasjonen hjelper meg med å forstå konteksten i retrospektivmøtet – med tanke på hva som har skjedd i teamet. Det ble gitt tilbakemelding til teamene som deltok i observasjonen etter at resultatene fra studiet ble klart. Deltakerne i studiet ble informert at jeg skrev masteroppgave om retrospektivmøter og at dataene som ble samlet inn skal brukes til dette formålet samt at lydopptakene blir slettet etter at masteroppgaven ble levert.

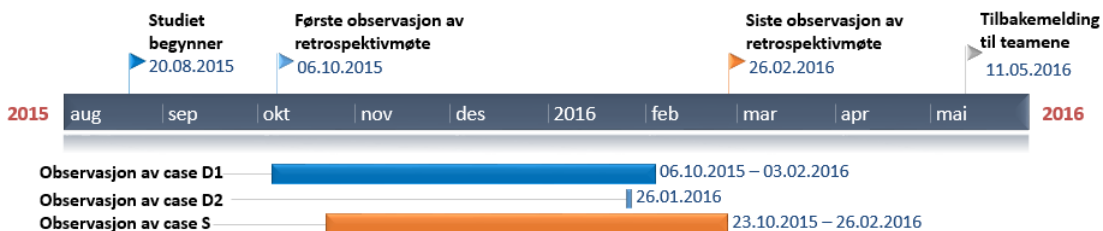
I distribuerte retrospektivmøter ble jeg presentert av møtelederen slik at de samtlige deltakerne forsto at de ble observert. Jeg fulgte begge casene fra by1. Distribuerte retrospektivmøte ble gjennomført ved bruk av videokonferanse, men jeg var ikke synlig for de virtuelle deltakerne under retrospektivmøtet unntaket er det teamet som jeg oppholdte meg hos. Denne plasseringen var viktig for å unngå forstyrrelser for gjennomføring av retrospektivmøte.

I det samlokaliserte retrospektivmøte var jeg godt synlig for teamet, selv om jeg prøvde skjermeseg unna teamet. Jeg satt ett stykke unna slik at deltakerne ikke skal få indikasjon på hva som ble notert ned på observasjonsnotatet under møtet. Dette er fordi deltakerne kan endre sin adferd som kan påvirke resultatet av observasjonene. I enkelte tilfeller har deltakerne i teamet henvendt seg til meg for å høre hva som ble notert, men jeg valgte ignorere forespørselen, da dette kan føre til endret adferd og vurderingsskjevhet (bias) [81].

5.4.2 Ustrukturerte intervjuer

Det ble brukt ustrukturerte (uformelle) intervjuer for å bekrefte observasjonene mine fra retrospektivmøtene. Disse intervjuene var korte samtaler før og etter retrospektivmøtene, gjerne på vei inn og ut av møterommet. På denne måten fikk jeg informasjon fra deltakerne angående endringer i forhold til den opprinnelig planen. Denne informasjonen gjorde at jeg lettere kunne forstå adferden og hendelsene som foregikk i retrospektivmøtene og bekrefte om hendelsene jeg observerte var korrekte.

5.4.3 Framdrift og omfanget av datainnsamlingen



Figur 5.1: Tidsplanen til datainnsamlingen

Dette er mitt første forskningsprosjekt og derfor tok det tid før forskningen kom godt i gang. I løpet av de første tre månedene ble det brukt mye tid på å lese litteratur for å tilegne seg kunnskap innenfor forskningsfeltet. Selve feltarbeidet begynte i oktober 2015 og varte fram til slutten av februar 2016, der det ble gjennomført observasjon av retrospektivmøter i samlokalisert og distribuert team, se figur 5.1. Det kan nevnes at det siste planlagte retrospektivmøtet var i midten av mars, men observasjonen av dette møtet ble ikke gjennomført.

Kategori	Case D1	Case D2	Case S
Antall deltakere	11	3	9
Møtevarighet	2 timer	1 time	2 timer
Antall observasjoner	2	1	7
Antall sider notater	12 A4-word sider	4 A4-word sider	49 A4-word sider

Tabell 5.3: Omfanget av datainnsamlingen

Bakgrunnsinformasjon om selve datainnsamlingen er beskrevet på tabell 5.3. Kategorien *antall deltakere* viser til antall deltakere i teamet som har deltatt på minst ett retrospektivmøte og det har ikke blitt tatt hensyn til hvor ofte deltakerne deltar i retrospektivmøter. Kategorien *møtevarighet* indikerer varigheten til retrospektivmøtet pr observasjon. Resultatene fra observasjonene er observasjonsnotater og samlet antall dataskrevne sider av alle observasjonene i caset er gitt i kategorien *antall sider notater*. Dette betyr at det totale dataskrevne materialet er basert på 65 dataskrevne sider.

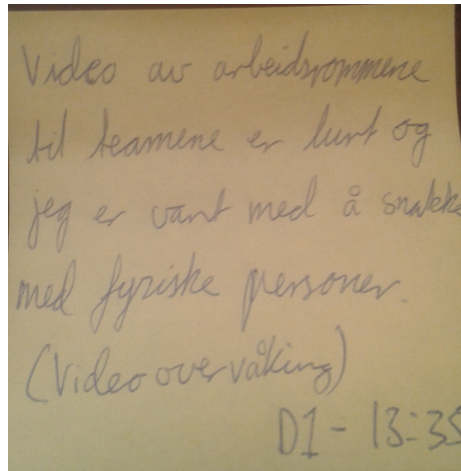
For å kunne sammenligne samlokalisert retrospektivmøte med distribuert retrospektivmøte bør antall observerte retrospektivmøter være lik. Dette er ikke mulig i dette studiet da de har forskjellig sprintlengde. Derfor er det flere observasjonsnotater i case S i forhold til case D1. I tillegg har jeg ikke gjennomført observasjoner av case D1 i tre tilfeller, da både case S og case D1 hadde retrospektivmøter på samme tidspunkt eller fordi møtet ble avlyst.

5.5 Datanalyse

Det er både gjennomført kvalitativ og kvantitativ datanalyse for å kunne svare på forskningsspørsmålet. Tematisk analyse er en kvalitativ dataanalysemetode som ble anvendt for å gjennomføre dataanalysen. Selve gjennomføringen er gjennomført i henhold til Oates beskrivelse av metoden og tilpasset mitt studiet [81]. Når det gjelder organisatorisk læring har jeg i tillegg brukt Argyis og Schön læringsmodell for å gjennomføre analysen [1].

5.5.1 Tematisk analyse

Jeg har kommet fram til fire forskjellige steg ut i fra beskrivelsen om tematisk analyse som Oates har beskrevet. Disse er forklart nedenfor.



Figur 5.2: Konteksten på en post-it lapp

I første steg har jeg lest gjennom datamaterialet mitt for å få generell overblikk over hva datamaterialet handlet om. Dette er fordi det er lenge siden jeg har skrevet observasjonsnotatene og trenger en oppfriskning når gjelder konteksten i de forskjellige observasjonsnotatene.

Roller (Forkortelse)	Eksempel
Prosjektleder (PL)	PL _{D1} - Prosjektleder i case D1
Scrum-master / squad-master (SM)	SM _S - Scrum-master i case S
Utvikler (U)	U _{D1} - Utvikler i case D1
Fagpersonell (FP)	FP _S - Fagpersonell i Case S

Tabell 5.4: Forkortelse for rollene

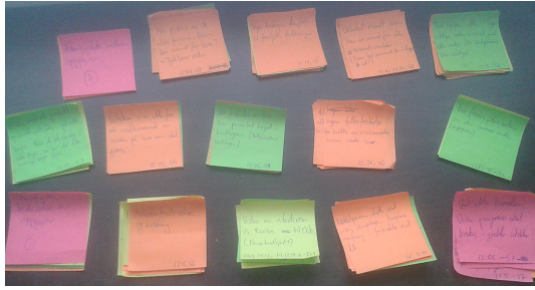
I andre steg gikk jeg gjennom hver av observasjonsnotatene mine og skrev post-it lapper i forhold til det som ser ut til å være interessant for forskningsspørsmålet mitt. På en post-it lapp har jeg skrevet ett utsagn (metafor i parentes) sammen med et klokkeslett og tilhørende nummer som identifiserer observasjonsnotatet, se figur 5.2. Når utsagnene fra observasjonene gjengis i resultat- og diskusjonskapittelet er navnet på deltakerne erstattet med deres roller i teamet etterfulgt med nedsenket bokstav som indikerer hvilken case deltakeren tilhører, se tabell 5.4. Det kan nevnes at fargen på post-it lappene har ingen betydning.

Etter hvert som jeg skrev post-it lapper strukturerte jeg lappene etter tilhørende observasjonsnotat og derfor hadde jeg ti forskjellige bunker med post-it lapper, se figur 5.3. Jeg har valgt å bruke post-it lapper framfor å bruke ett dataprogram, da det er enklere å spre lappene utover og prøve å se sammenhenger med disse. Denne fordelene beskrives også av Oates [81].



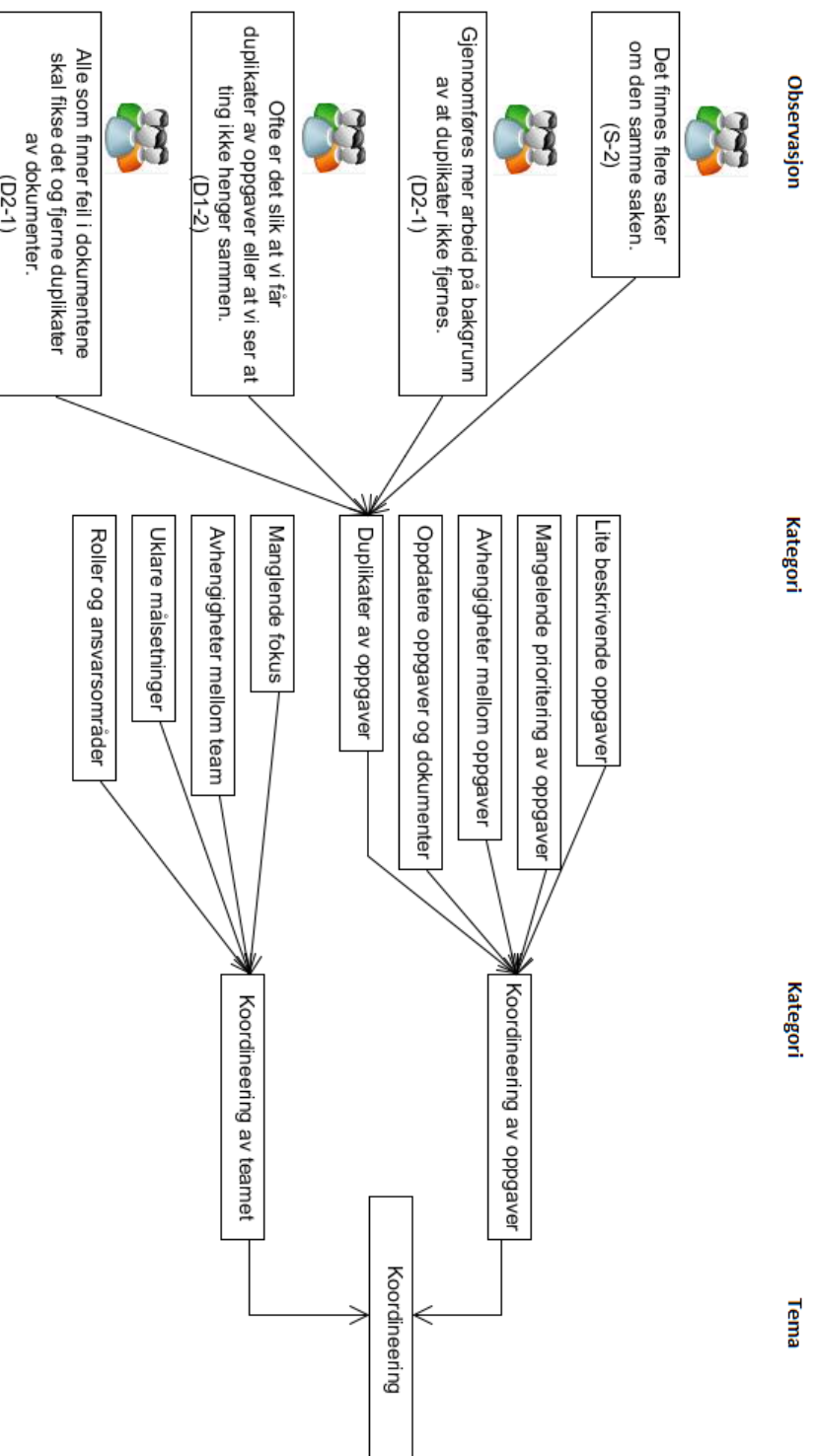
Figur 5.3: Etter å skrevet post-it lapper for alle observasjonsnotatene

I det tredje steget gikk jeg gjennom alle post-it lappene og aggregerte post-it lappene og identifiserte kategorier. Kategoriene ble identifisert ved hjelp av eksisterende teori og begreper samtidig som jeg var åpen for å være påvirket av nye begreper som oppstår ved gjennomgang av post-it lappene. På denne måten hindrer jeg å ikke være for mye bundet av eksisterende teori og begreper, dermed ikke gå glipp av andre interessante temaer i datamaterialet. Etter å gjennomført dette trinnet har jeg fått 15 kategorier, som er vist på figur 5.4.



Figur 5.4: Etter å ha identifisert 15 forskjellige kategorier

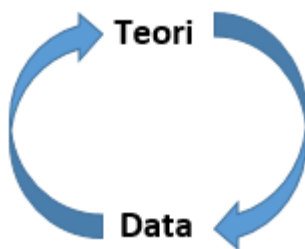
Det fjerde og siste trinn besto av å aggregere kategoriene slik jeg at jeg fikk færre kategorier å forholde meg til. Dette ble gjennomført ved å identifisere sammenheng mellom dem. Resultatet av denne aktiviteten førte til å redusere 15 kategorier til 2 kategorier. Sammenslåingen av kategorier førte til at jeg endte opp med to temaer som var sentrale, der en av dem var koordinering og den andre var kommunikasjon. En helhetlig oversikt over analyseprosessen for temaet koordinering er gjengitt i figur 5.5. Senere viste det seg at jeg ikke kunne forklare funnene til koordinering uten å beskrive kommunikasjonsutfordringene, da disse er to sider av samme sak. Derfor har de blitt slått sammen og presentert som ett tema, nemlig kommunikasjon. Tilslutt ble mine funn beskrevet i forhold til eksisterende litteratur i innenfor kommunikasjon.



Figur 5.5: En oversikt over analyseprosessen for temaet koordinering (kommunikasjon)

5.5.2 Dataanalyse for organisatorisk læring

Datanalysen for organisatorisk læring ble gjennomført ved bruk av modellen til Argyis og Schönén [1]. Det ble brukt post-it lapper på tilsvarende måte som er beskrevet i forrige delkapittel. I første omgang ble det gjennomført en deduktivt analyse ved at jeg prøvde å identifisere styringsvariablene og konsekvensene for arbeidsmiljø, læring og effektivitet i datamaterialet mitt. Det er gitt ett eksempel for analyseprosessen for styringsvariabelen, gyldig informasjon i modell II, se figur 5.7 og utsagnene er hentet fra case S.



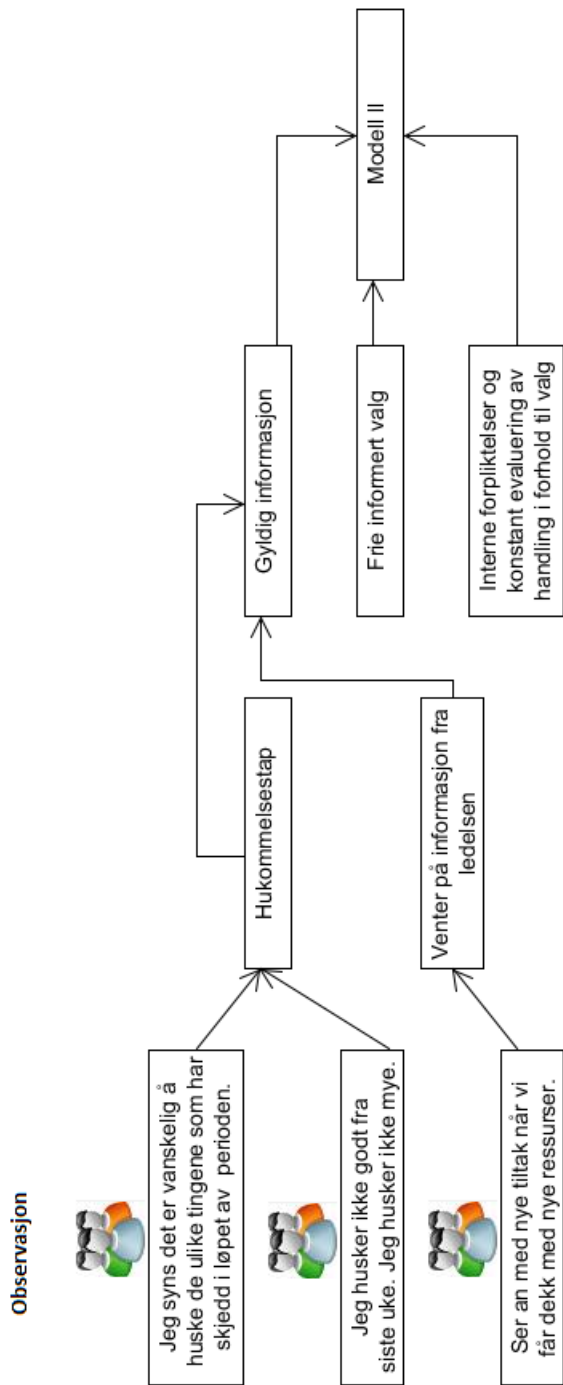
Figur 5.6: Analyseprosessen

Når det gjelder identifisering av enkeltløkke- og dobbeltløkkelæring har jeg gjennomført dette i flere iterasjoner. Dette er fordi synet mitt på enkelt- og dobbeltløkkelæring gradvis endret seg, ved at jeg tilegnet meg mer kunnskap om læring underveis i studiet. Derfor gikk jeg tilbake på tidligere observasjonsnotater for å få tanker og idéer om enkeltløkkelæring og dobbeltløkkelæring som har modnet seg. På denne måten fant jeg at det som tidligere var identifisert som enkelt- og dobbeltløkkelæring var ikke det likevel. Ved å gjennomføre dataanalysen på denne måten øker kvaliteten på resultatet. Denne prosessen er illustrert på figur 5.6.

En av utfordringene jeg møtte ved identifiseringen av dobbeltløkkelæring, hvorvidt forbedringstiltak må være gjennomført for å danne dobbeltløkkelæring. Det endte med at det var nødvendig å gjennomføre forbedringstiltak for å danne dobbeltløkkelæring. I case D1 har jeg hatt tilgang til data som bekrefter om forbedringstiltakene er gjennomført eller ikke, men dette har ikke vært tilfelle i case S. Gjennomføring av forbedringstiltakene i case S er basert på retrospektivobservasjoner, noe som kan føre til at det kan oppstå avvik mellom mine tolkninger og de faktiske resultatene.

Tematikk i retrospektivmøter

Når det gjelder tematikken i retrospektivmøtene, så har jeg observert hvor ofte temaene diskuteres ved bruk av datanalysemetoden ”frekvens av metaforer” [81]. Metoden går ut på å telle opp antall ganger metaforen blir brukt over en tidsperiode, der det settes en kryss for hver gang metaforen blir diskutert. I mitt tilfelle har jeg kun satt kryss hvis deltakerne har diskutert temaet i retrospektivmøtene og blankt felt hvis ikke de har diskutert temaet i retrospektivmøtet. Denne datanalysemetoden er kun anvendt i case S, da jeg har tilstrekkelig antall observasjoner til å kunne benytte denne dataanalysemetoden. Jeg har avgrenset temaene som tas med i resultatet etter at analysen ble gjennomført. Resultatet viser kun temaene som gjentas minst tre ganger i løpet av de syv retrospektivmøtene.



Figur 5.7: Analyseprosessen for styringsvariabelen gyldig informasjon i modell II

5.6 Refleksjon

Studiet er gjennomført som en fortolkningsstudie. Det er brukt Klein og Mayers syv prinsipper om fortolkningsstudier for å gi deg som leser dypere forståelse om hva jeg legger til grunne for å ta de konklusjoner jeg kommer fram til [69]. Disse syv prinsippene er beskrevet i forhold til casestudiene som er gjennomført.

Prinsippet om hermeneutisk sirkel innebærer at man må forstå helheten for å forstå deler og at man må forstå deler for forstå helheten. I begynnelsen av studiet hadde jeg lite kunnskap om retrospektivmøter og hvordan disse i praksis ble gjennomført. Jeg tilegnet meg ny kunnskap om retrospektivmøter og teamene etterhvert som retrospektivmøter ble observert. Etter å ha observert to møter, gikk jeg ofte tilbake til tidligere observasjonsnotater og leste disse igjen. Dette ført til at jeg fikk ny forståelse for hendelsene som har skjedd og kunne sette disse i en større sammenheng. Det å kunne sette enkelt faktorer som har skjedd opp mot hverandre med ny tilegnet kunnskap har gitt meg bedre forståelse. Denne forståelsen har ført til det har blitt større endringer underveis i avhandlingen.

Prinsippet om kontekstualisering innebærer at du som leser har den samme kontekstuelle forståelsen om casene i studiet som jeg hadde når studiet ble gjennomført. Derfor er bakgrunnsinformasjonen om casene gitt i kapittel 5.2 og 5.3, samtidig er retrospektivspraksisen beskrevet i 6.1. Datamaterialet mitt er hovedsaklig basert på observasjon, og derfor er resultat mitt preget av mine fortolkninger. Hvert menneske fortolker situasjoner og bearbeider dataene ulikt. Basert på dette kan enkelte lesere være uenig i mine fortolkninger. Likevel håper jeg at kontekstualisering bidrar med å forstå mine beslutninger.

Prinsippet om interaksjon mellom forsker og subjekt innebærer at forskeren øker forståelsen ved å interagere med deltakerne i studiet. Min forståelse av casene ble økt gjennom korte samtaler med enkelte deltakere før og etter retrospektivmøtene. Samtidig ble det tatt en runde med tilbakemeldingsmøter med casene som deltok i studiet og for å sikre at resultatene mine var troverdig. I denne anledning påpekte deltakerne at de kjente seg igjen i resultatene, selvom enkelte var usikre på fortolkningene av forbedringstiltakene stemte med deres syn. Derfor har jeg i denne oppgaven presentert læringseffekten i retrospektivmøter slik jeg har erfart det og supplert med informasjon fra deltakerne i studiet.

Prinsippet om abstraksjon og generalisering innebærer at resultatet av studiet skal være generaliserbar. Retrospektivmøtene gjennomføres forskjellig, da dette er avhengig av hva som har skjedd i den siste sprinten og hvem som deltar i møtet. Derfor vil det være vanskelig å gjenskape resultatene fra dette studiet nøyaktig på et detaljert nivå. På en annen side er det brukt Argyis og Schörens læringsmodeller og kommunikasjonsteori for å beskrive resultatene, og derfor vil være mulig å gjenskape resultatene på ett overordnet nivå.

Prinsippet om dialogisk argumentasjon går ut på mulige forskjeller mellom teori og funn. Jeg har analysert enkeltløkke- og dobbeltløkkelæringer ved å lese gjennom observasjonsnotatene i flere iterasjoner. I begynnelsen klassifiserte jeg læringen enten som enkeltløkke- og dobbeltløkkelæring. Senere da jeg gjennomførte en ny iterasjon med mer åpent syn, kom jeg bort i situasjoner som hverken var enkeltløkkelæring eller dobbeltløkkelæring og som tidligere har vært klassifisert som enkeltløkke eller dobbeltløkkelæring. I løpet av studieperioden har synet mitt på læringen endret seg fra å kun skille mellom enkelt og dobbeltløkkelæring til å finne andre læringstilfeller.

Prinsippet om ulike fortolkninger innebærer at deltakerne i studiet kan ha forskjellig fortolkning av samme situasjon, altså at deltakerne beskriver samme fenomenet forskjellig. En formell intervju av samtlige deltakerne i studiet kunne ha vært en mulighet for å få fram forskjellige fortolkninger og dermed styrke troverdigheten i datamaterialet. Det er en svakhet at det er hovedsaklig brukt observasjon som datainnsamlingsmetode eller datakilde. Dette fører til at troverdigheten til innsamlet datamateriale svekkes. I tre møter ble det brukt penn og papir for å skrive ned observasjonene hvor jeg kan ha gått glipp av informasjon. Det er likevel en grunn til å tro at budskapet i notatene er bevart.

Prinsippet om vurderingskjevhet² innebærer blant annet at deltakerne kan endre adferd og som følge av dette påvirkes resultatet av forskningen. Jeg har valgt to (tre) caser som jeg vet på forhånd har en eksisterende praksis i forhold til gjennomføring av retrospektivmøter, noe som ikke er tilfelle hos samtlige systemutviklingsteam. Dette kan føre til vurderingskjevhet, da disse teamene vil være mer erfarne til å gjennomføre retrospektivmøter i forhold til et team som ikke gjennomfører retrospektivmøte så ofte. Dette er mitt første forskningsprosjekt der jeg gjennomfører observasjon av retrospektivmøtet, da har jeg nødvendigvis ikke lært alle metodene på hvordan jeg gjennomfører observasjon, noe som kan ha påvirket resultat av studiet. Det er ikke sikkert at deltakerne var komfortabel med lydopptak av retrospektivmøtene, og derfor kan det tenkes at deltakerne endret adferd eller holdt tilbake informasjon under møtene. Samtidig kan det tenkes at det tar en stund før deltakerne er komfortabel med at en tredjeperson er tilstede under møtet. Deltakerne har underveis i studiet spurt meg om hva jeg synes om deres retrospektivpraksis og hvilke funn jeg har funnet, men jeg har ikke gitt svar på disse spørsmålene da dette kan påvirke resultatet av forskningen.

²Engelsk: Bias

Kapittel 6

Resultater

Dette kapitlet viser resultatene fra studiet hvor alle hendelsene er kategorisert etter hvor i retrospektivmøtet de oppsto. Det har blitt observert 10 retrospektivmøter i dette studiet og disse retrospektivmøtene er trekket sammen og presentert som ett retrospektivmøtet nedenfor. Det betyr at aktivitetene som presenteres beskriver de ulike aktivitetene fra de ulike retrospektivmøtene som ett retrospektivforløp. Modellen som er basert på eksisterende litteratur for gjennomføring av retrospektiv (se kapittel 3.4) brukes for å forklare retrospektivpraksisen til case S og D1 og resultatene i forhold til organisatorisk læring og kommunikasjon. Når det gjelder case D2, er selve retrospektivpraksisen beskrevet i vedlegg A.4, da datamaterialet er begrenset.

Retrospektivmøtet i case S gjennomføres ulikt fra case D1, da det ene teamet er samlokalisert, mens andre er distribuert. Det har foregått signifikante endringer i forhold til selve gjennomføringen av retrospektivpraksisen i disse casene. Eksempelvis har deltakerne i case D1 gått fra å bruke fysiske lapper til å bruke verktøy for gjennomføring av retrospektivmøtene. Disse variasjonene beskrives nedenfor. Det brukes konkrete eksempler for å gi deg som leser innsikt i hva som har skjedd i retrospektivmøtene.

6.1 Gjennomføring av retrospektiv

6.1.1 Trinn 1: Forberedelse

Dette trinnet handler om forberedelsene til retrospektivmøtet for både deltakerne og fasilitatoren. Forberedelsene inkluderer hvem som skal inviteres, hjemmelekse og valg av fasilitator.

Case S	Case D1
Prosjektleder	Prosjektleder
Scrum-master	Squadmaster (SM)
Utviklere	Utviklere
Fagpersonell	

Tabell 6.1: Rollene som har deltatt i møtene

I begge casene har alle som har bidratt til å gjennomføre prosjektet, inkludert prosjektlederen blitt invitert til å delta i møtet, se tabell 6.1 for nærmere informasjon. Ledelsen er ikke med i retrospektivmøtet, men prosjektlederen er invitert i begge casene. Underveis i studiet har deltakerne vært fraværende fra retrospektivmøtet i begge casene, da de har vært borte i løpet av siste sprint. En annen årsak for at deltakerne ikke deltar i møtet forklares av PL_{D1} :

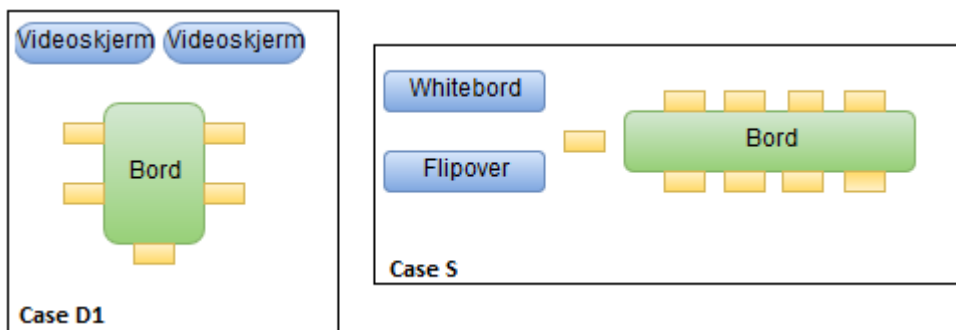
”Når det gjelder deltakelse skal alle som har jobbet i sprinten være med, der har vi frafall fra ymse grunner. Det er valgfritt for de nye, og det er ingen som stiller. Det er ett godt tegn på at de vil jobbe istedet.”

I både case S og case D1 gis det ingen hjemmelekse, der deltakerne skal reflektere over spørsmål i forkant av møte. Datainnsamlingen gjennomføres i forkant av møtet i case D1, dersom deltakeren på forhånd vet at vedkommende vil være fraværende på retrospektivmøtet.

Det ble hovedsaklig brukt intern fasilitator i begge casene, bortsett fra det første retrospektivmøtet i case S, da var fasilitatoren ekstern. Ingen av casene hadde en fast fasilitator for retrospektivmøtet. I case S hadde scrum-masteren fått ansvaret for gjennomføring av retrospektivmøtet. Aktivitetene som ble brukt i retrospektiv varierte i stor grad, da scrum-master rollen gikk på rundgang. Scrum-masteren var fasilitatoren i case D1 for det første møtet, mens prosjektlederen var det i andre møtet. Forberedelsen til fasilitatorene besto hovedsakelig av å utforme agendaen for møtet og undersøke hvilke aktivitetene som kan gjennomføres for å få teamet til å reflektere.

6.1.2 Trinn 2: Introduksjon

Det andre trinnet er introduksjon, der deltakerne får presentert agendaen for møtet og skal oppfordres til å dele sine forventinger til møtet, da dette vil skape god atmosfære. En oversikt over hvordan deltakerne satt i forhold til hverandre i begge casene er illustrert på figur 6.1 (gul rektangel indikerer hvor deltakere sitter). Denne figuren vil brukes for å beskrive hvordan gjennomføringen av retrospektivmøtet foregår praksis i neste trinnene.



Figur 6.1: Hvordan deltakerne sitter i forhold til hverandre

Retrospektivmøtene begynner med å presentere agendaen i begge casene, men fasilitatoren har ikke bedt deltakerne til å fortelle om sine forventinger til møtet i samtlige observerte retrospektivmøter. Isteden har fasilitatoren bidratt for å oppmuntre deltakerne i teamet for å skape en god atmosfære, SM_S:

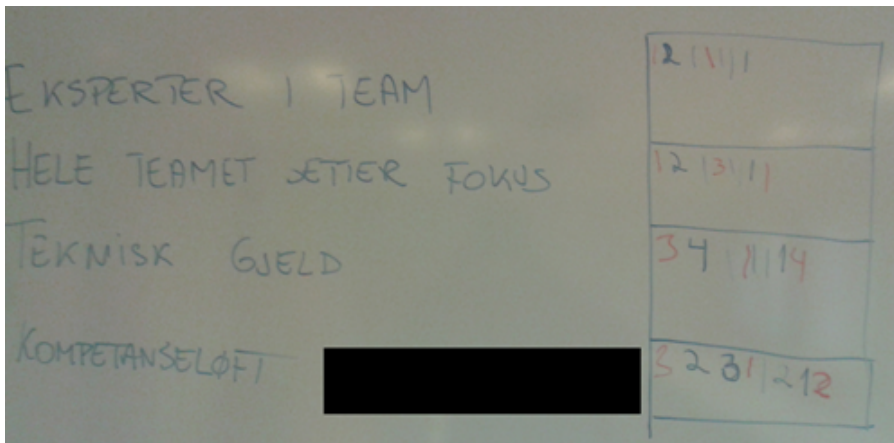
”Vi er alle en del av teamet, og hvis noen har lyst til å ta fram hva de personlig kan bli bedre, så gjerne det. Vi som team er aldri bedre enn summen av oss alle, og vi utfyller hverandre veldig godt.”

Utsagnet viser at deltakerne har evnen til å tro at alle sammen har gjort det sitt beste i løpet av sprinten.

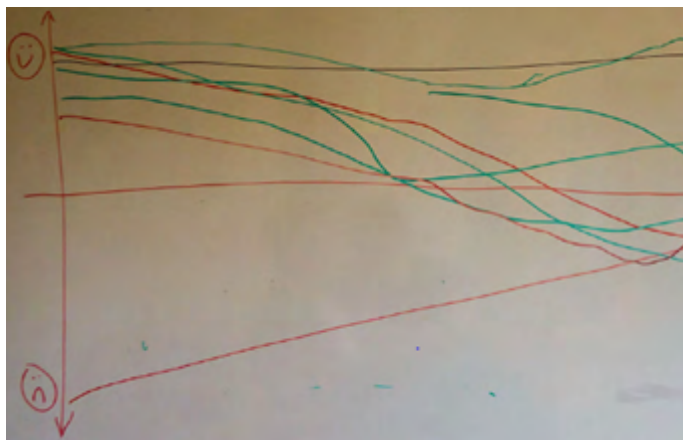
Isteden for å dele sine forventinger til møtet, begynner deltakerne rett på sak med en aktivitet som forbereder dem på neste trinn, datainnsamling. Dette er ikke hensikten i følge modellen, men aktivitetene som er gjennomført er forklart. I case S har deltakerne gjennomført flere forskjellige aktiviteter for det andre trinnet: tidslinje, fornøydhetsskurven, vurdering av tidligere forbedringstiltak og termometer. I case D1 har deltakerne brukt to forskjellige aktiviteter: beskrivelse av forrige sprint med ett ord og vurdering av tidligere forbedringstiltak. Vi skal se nærmere på to av disse aktivitetene, vurdering av tidligere forbedringstiltak og fornøydhetsskurven.

Vurdering av tidligere forbedringstiltak har blitt gjennomført én gang av case S og case D1. I dette tilfellet beskriver vi hvordan case S har gjennomført denne aktiviteten. I aktiviteten skal deltakerne i teamet ta stilling til følgende spørsmål,

SM₅: "Sette en vurdering i hvilken grad de selv føler at de har bidratt til at tiltakene holdes i livet". I praksis gjøres dette ved at deltakerne går opp en og en til whiteboard, der fasilitatoren har tegnet en tabell med to kolonner. I den ene kolonnen har fasilitatoren skrevet forbedringstiltakene fra forrige retrospektivmøte og deltakerne skal skrive ett tall i den andre kolonnen, se figur 6.2. Tallene som brukes er fra skala 1-5, der 5 indikerer at deltakerne har gjort mye for at forbedringstiltakene skal gjennomføres. Resultatet av aktiviteten viser at deltakerne har gjort minimal innsats for å gjennomføre forbedringstiltakene, der gjennomsnittet fra alle forbedringstiltakene er avrundet til 1,7, altså nærmere 2.



Figur 6.2: Vurdering av tidligere forbedringstiltak



Figur 6.3: Gjennomføring av fornøydhetsskurven

I case S har fornøydhetsskurven blitt brukt to ganger som den første aktiviteten i retrospektivmøtet. Hensikten med fornøydhetsskurven er å se hvordan humøret til de enkelte deltakerne har vært gjennom sprinten og denne aktiviteten reflekterer også humøret til teamet som helhet. Aktiviteten begynner med at fasilitatoren tegner et todimensjonalt koordinatsystem, der x-aksen er sprintlengden (tiden) og y-aksen er humøret (positivt til negativt), se figur 6.3. Deretter går hver deltaker opp til whiteboard og tegner grafen som gjenspeiler deres humørsvingninger på forrige sprint, mens de forklarer overfladisk begrunnelsen for grafen. Deltakerne har hatt flere utfordringer knyttet til scrum-master rollen. Disse utfordringene gjenspeiles når deltakerne skal formidle humørsvingningene sine i sprinten. U_S tegnet en graf med en store svingninger i det første retrospektivmøtet og forklart dette med: *”Årsaken til bunnpunktene er scrum-master rollen”*. Bakgrunnen for dette har vært at ingen av deltakerne ønsker å være scrum-master da dette krever mye ansvar og forpliktelser. En annen deltakerne sliter med å tegne fornøydhetsskurven og bakgrunnen for dette, beskrives av U_S :

”Jeg har hatt [fornøydhetsskurve] tidligere, og jeg har slitt med å huske hva som har skjedd i de siste 14 dagene”.

6.1.3 Trinn 3: Datainnsamling

Hendelsesforløp
1. Skrive lapper om hva som har gått bra.
2. Presentere positive lapper
3. Skrive lapper som hva som kunne ha gått bedre
4. Presentere forbedringslapper lapper.
5. Gruppere positive lapper
6. Gruppere forbedringslapper lapper

Tabell 6.2: Hendelsenforløpet i trinn 2, bortsett fra ved bruk av verktøy

Dette er hovedtrinnet i retrospektivmøtet ifølge Derby og Larsen [7], der alle dataene samles inn. Hensikten med dette trinnet er at deltakerne i teamet skal ha samme utgangspunkt for å diskutere utfordringene i neste trinn som er diskusjon og utarbeide forbedringstiltak. Dette trinnet gjennomføres i seks steg i både case S og case D1, der fire første stegene utgjør de to hovedstegene. Det første hovedsteget består av å skrive post-it lapper om *hva som har gått bra i forrige sprint* (positive lapper) deretter presentere lappene, mens det andre steget består av å skrive post-it om *hva som kunne ha gått bedre i forrige sprint* (forbedringslapper) og deretter presentere lappene, se tabell 6.2. Den eneste forskjellen mellom disse to hovedstegene er spørsmålet som skal besvares av deltakerne. Derfor presenteres begge hovedstegene samlet nedenfor.

Det er forskjeller på gjennomføringen av datainnsamlingen i case S og case D1. I case D1 har det vært begrensninger for hvor mange lapper en deltaker kan skrive for at de ikke skal bruke for lang tid. Derfor kan deltakerne maksimalt bruke fire lapper hver på hver av hovedstegene. I case S har det ikke vært begrensninger for antall lapper de kan skrive.

Etter at deltakerne har skrevet lappene, så presenteres lappene for resten av teamet. I case S går deltakerne enten opp til flip-over eller whiteboard (avhengig av fasilitatoren) og presenterer lappene som de har skrevet slik at resten av teamet forstår konteksten i lappen. Nedenfor illustreres ett utsagn da U_S presenterte en positiv lapp i første retrospektivmøtet og uttalsen var uklar. Fasilitatoren var flink til å følge opp uklare uttalelser.

U_S : *"Det er rett fokus og godt samarbeid"*

SM_S (fasilitator): *"Utdyp hva som menes med rett fokus og godt samarbeid"*.

Det fjerde steget er å presentere forbedringslappene for resten av temaet. I tredje retrospektivmøtet presenterte U_S en utfordring knyttet sammenhengen mellom bugs og ressurser: *"Fremdeles mange bugs, så mangler vi ressurser og vi kommer ikke av flekken"*. Dette viser at case S har varige utfordringer i forhold til bugs og ressurser som påvirker framdriften til teamet. En utfordring deltakerne har hatt tidligere, ble presentert igjen i fjerde retrospektivmøtet, U_S : *"Alle oppgavene blir prioritert like høyt"*. I sjette retrospektivmøtet U_S presentert at nytteverdi av retrospektivmøtet var redusert:

"Jeg synes vi har retrospektiv for ofte og hvertfall slik vi har hatt før. Det har vært repetisjon hver gang. Hva slags utbytte har vi av det? Liten. Hvis vi ikke finner en bedre måte å gjøre det på, så ser ikke jeg noen grunn til ha det så ofte".

I case D1 har deltakerne to skjermer, der den ene skjermen brukes for å vise deltakerne som er med via videokonferanse, men den andre skjermen brukes til å vise det som skrives ned av fasilitatoren. Dette trinnet har blitt gjennomført forskjellig i begge retrospektivmøtene. I første retrospektivmøtet ble post-it lapper benyttet for gjennomføre datainnsamlingen, mens i det andre retrospektivmøtet ble verktøy benyttet, nærmere sagt Padlet, se vedlegg A.1. I det første retrospektivmøtet har deltakerne skrevet post-it lapper og deretter presentert dem ved å lese lappene høyt for resten av teamet, inkludert for sine distribuerte deltakerne som er med via videokonferanse. Fasilitatoren skrev ned det som ble sagt i en confluence side som ble vist på skjermen. På denne måten kan de distribuerte deltakerne få med seg det som blir sagt og skrevet.

I første retrospektiv presenterte U_{D1} en positiv lapp om ledelsen: *"Flott ledelse som kjemper om å få riktig person til riktig prosjekt"*. Dette indikerer at deltakeren er fornøyd med innsatsen til ledelsen. I begge retrospektivmøtene handlet forbedringslappene om kunden, arbeidsoppgaver i Jira og kommunikasjon. I første retrospektivmøte understreket U_{D1} :

"Noen ganger er det vanskelig å avklare med kunden. Kunden er fraværende og har manglende eierskap til produktet."

Denne utfordringen ble ikke løst i det andre retrospektivmøtet som ble observert, U_{D1} :

"Sånn overordnet så er de reaktive og ikke proaktive. Vi skal hele tiden gi dem konkrete oppgaver også maser vi på dem et par ganger før det i det hele tatt blir gjort noe."

I første retrospektivmøtet i case D1 har U_{D1} formildet utfordringer knyttet til kommunikasjon: *"Det er et skille mellom [by1] og [by2]. Vi bør ha kontinuerlig video av arbeidsrommene"*. Dette viser at deltakerne har en utfordringer i forhold til å kommunisere på tvers av lokasjonene i distribuert team. I andre retrospektivmøtet har et problem om arbeidsoppgavene i sprintkøen blitt presentert av U_{D1} :

"Ofte er det slik at vi får duplikater av oppgaver eller at vi ser at ting ikke henger sammen og kvaliteten på Jira er ikke så god som gjerne skulle sett at den var."

I andre retrospektivmøtet har SM_{D1} presentert en lapp som viser at det oppstår forskjellige fortolkninger:

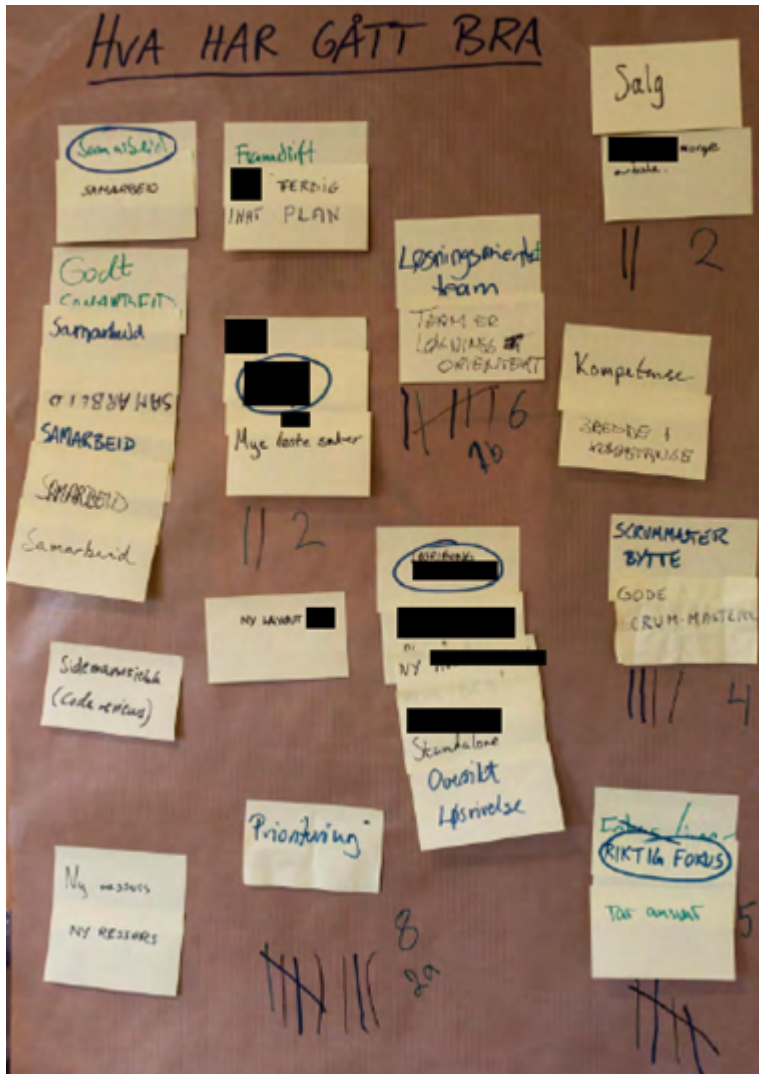
"Vi mangler den felles referanserammen. Vi har ikke felles vokabular og felles be- greper. Når du spør noen, så får du forskjellige svar avhengig av hvem du spør"

I det andre retrospektivmøtet ble verktøyet benyttet, forskjellen fra tidligere var at alle deltakerne hadde med seg en bærbar PC som de brukte for å skrive virtuelle lapper. I tillegg var deltakerne anonyme ved å skrive disse lappene. Fordelen med å bruke verktøyet var å kunne skrive lappene i forkant av møtet hvis deltakeren visste at vedkommende vil være fraværende fra retrospektivmøtet.

Etter at lappene er presentert så er det neste steget å gruppere lappene, både når det gjelder positive og forbedringslapper. I case S foregår grupperingen av lappene ved at fasilitatoren gruppere lappene på whiteboard eller flipover som hører sammen, og deretter setter navn på de grupperte lappene, mens resten av teamet tar en pause. Grupperingen av positive og forbedringslappene gjøres hver for seg, og hver gruppe med lapper representerer et tema, se figur 6.4. Eksempelvis ble lappen "Problem med pullrequest" og "Mye bugs" gruppert sammen når det gjelder forbedringslapper og ga grupperte lappene navnet "Tekniske feil", disse er ikke vist

på figuren. I neste de trinnene forholder teamet seg til de temaene som de grupperte lappene representerer og temaet er gitt ved å navngi grupperte lappene. Hensikten med grupperingen av lapper er å gjøre det enklere og mer oversiktlig for å stemme over lappene som deltakerne selv har skrevet. Heretter refereres grupperte lapper som tema.

I case D1 er det ikke like enkelt å gruppere lappene, da man sitter fysisk adskilt. Denne utfordringen ble løst ved at fasilitatoren og andre deltakerne så over listen på Confluence og diskuterte seg fram til grupperingen av lappene. Denne utfordringen forsvant når deltakerne i case D1 gikk over til å bruke verktøyet, da de enkelt kunne gruppere virtuelle post-it lapper.



Figur 6.4: Hvordan votering og gruppering av positive lapper vanligvis foregår i case S.

6.1.4 Trinn 4: Diskusjon og utarbeide forbedringstiltak

Hendelsesforløp
1. Votering av temaer
2. Diskusjon
3. Utarbeide forbedringstiltak

Tabell 6.3: Hendelsesforløpet i trinn 3

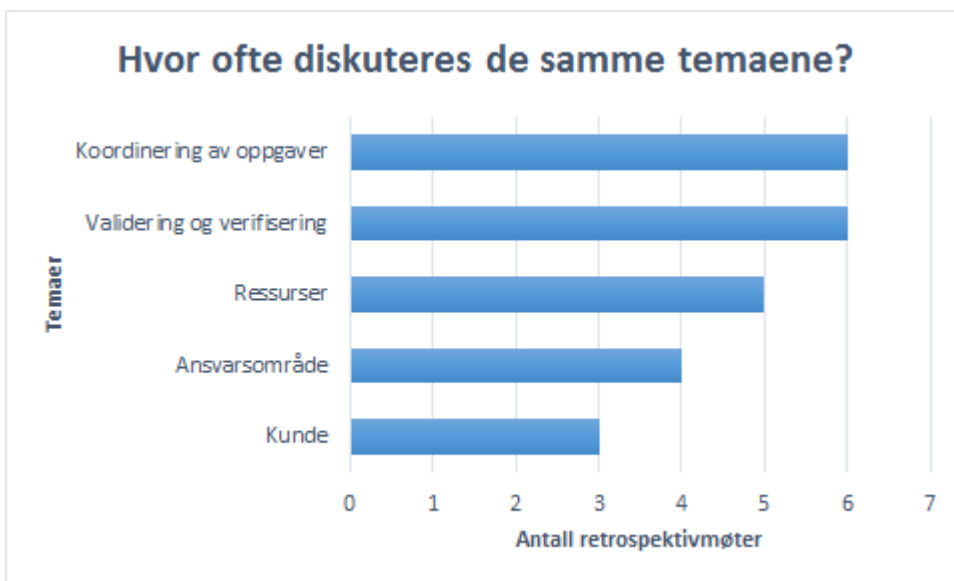
Derby og Larsen opplyser formålet med dette trinnet er å se hvordan teamet kan arbeide mer effektivt [7]. I retrospektivmøte har ikke deltakerne muligheten til å diskutere samtlige temaene som tas opp, og derfor må deltakerne i teamet prioritere eller velge ut hvilke temaer de ønsker å diskutere nærmere både når det gjelder positive og forbedringslapper. I dette studiet er dette trinnet hovedsakelig gjennomført i tre steg som består av votering, diskusjon og utarbeide forbedringstiltak, se tabell 6.3. Fokuseret retrospektivmøte er at deltakerne fokuserer på en eller to bestemte utfordringer, se tabell 6.4. Det betyr at hovedtyngden på ligger på disse utfordringene i diskusjonen.

Case S	Case D1
Møte 3	Møte 2
Møte 4	
Møte 7	

Tabell 6.4: Fokuseret retrospektivmøter

I case S har deltakerne prioritert de ulike temaene ved å gjennomføre votering. Voteringen foregår ved at deltakerne har tre stemmer hver som brukes til å stemme på temaene. De velger selv hvordan de fordeler disse tre stemmene på de ulike temaene, eksempelvis kan deltakerne gi tre stemmer til et bestemt tema eller fordele disse på tre temaer. I praksis foregår dette ved at hver deltaker går opp en og en til der lappene henger, og bruker tussj for å sette en strek som indikerer en stemme, se figur 6.4. Dette er en motsetning til hvordan case D1 gjennomførte vteringsprosessen ved at fasilitatoren spurte deltakerne om hvilke temaer de ønsker å diskutere og ut ifra stemmingen i teamet valgte fasilitatoren ett hovedtema for diskusjonen. I case S diskuteres de samme temaene gjentatte ganger, se diagram 6.5. Tallene som diagrammet viser omfatter kun negative utfordringer knyttet til temaet og nærmere beskrivelse av temaene er gitt i vedlegg C. Deltakerne er også bevisste på at de samme temaene gjentas flere ganger, SM_S:

"[...] Så må vi i tilfelle komme opp med ett nytt fokusområde som gjør at vi ikke repeterer oss selv - jeg hører at vi repeterer oss selv, så såpass lenge har jeg vært med nå. Det er er stort sett de samme problemområdene som kommer fra gang til gang også har vi bestemt oss for noen tiltak."



Figur 6.5: Oversikt over de mest diskuterte temaene i løpet av studiet.

Temaene som fikk flest stemmer prioriteres når diskusjonen begynner i et retrospektivmøte. I case S gjennomføres diskusjonen på to forskjellige måter avhengig av fasilitatorens ønske. Den ene måten er å dele deltakerne i to grupper, mens den andre måten er å diskutere de ulike temaene i felleskap, se tabell 6.5. Når deltakerne deles i to grupper, diskuterer den ene gruppen temaet som er høyest prioritert for positive lapper, mens den andre gruppen diskuterer temaet som er prioritert høyst for forbedringslapper. Når deltakerne diskuterer i felleskap, diskuteres flere temaer i forhold til å diskutere i smågrupper. Resultatet fra små gruppediskusjoner er årsaker og sammenhenger for hvorfor behovene eller utfordringene finner sted tilknyttet utdelt tema, samtidig som det utarbeides forbedringstiltak for disse. Gruppene presenterer disse resultatene for hverandre og deretter foretas det en kort fellesdiskusjon for begge temaene.

Små grupper:	Kun felleskap:
Møte 1	Møte 2
Møte 3	Møte 5
Møte 4	Møte 6
Møte 7	

Tabell 6.5: Diskusjonsstrukturen i retrospektivmøtene i case S

Strukturen på diskusjonen kan være strukturert eller åpen. I case S har deltakerne hatt to strukturerte diskusjoner og resten har vært åpne diskusjoner. I en av de strukturerte diskusjonene har de diskutert hvordan de skal ta over kompetansen til en nøkkeldeltaker som slutter i teamet. Ved å gjennomføre strukturerte retrospektivmøter får deltakerne muligheten til å ha like stor innflytelse på diskusjonene. Ved å gjennomføre åpen diskusjon så har deltakerne fått muligheten til å ta de utfordringene de selv synes er viktig.

I tredje retrospektivmøtet i case S ble bugfixing votert som tema for å diskutere nærmere. Under diskusjonen kommer det fram ulike synspunkter for hva som hindret deres framdrift. En deltaker fortalte at *"vi har få ressurser (få utviklere), slik at vi rekker ikke å gjøre noe annet enn å fikse bugs"*, mens en annen deltaker har ett annet synspunkt *"Jeg vet ikke hva jeg skal teste"*. En tredje deltaker forklarer at vedkommende har manglende domenekunnskap, som er nærmere beskrevet i kapittel 6.2.2. Samme tilfellet har skjedd tidligere før dette studiet begynte, der fikk de ikke nytte av kurset som forklares av U_S :

"Vi var på et møte om [domenekunnskap] og poenget var å begynne og jobbe og det har ikke ennå skjedd, vi har sikkert glemt kunnskapen. Vi må sikkert ta møtet igjen"

I dette retrospektivmøtet har teamet bestemt at det skal utarbeideres testplan for å sikre at deltakerne skal vite hva som skal testes. I tillegg har teamet i dette tilfellet vurdert å evaluere effekten av forbedringstiltakene, der SM_S utdyper følgende:

"Vi har kommet fram til at det å måle bugs er helt greit da det er kvantitativt, og det er vanskelig å bruke retrospektiv da det er subjektiv. Å måle bugs er objektivt og kan ses på over tid - man bør se at teamet produserer ny funksjonalitet samtidig som antall bugs går ned - da er målet nådd."

I case S har scrum-masterrollen vært ett av utfordringene som har vært en gjenganger i retrospektivmøtene. Før studiet begynte hadde case S en fast scrum-master, men etterhvert ombestemte vedkommende på bakgrunn av:

"Problemet mitt var at jeg fikk ansvar for andre ting enn en scrum-master har. Det dro på seg. Du har en å peke over og du er scrum-master og må gjøre det"

Diskusjonensstrukturen i case D1 har vært åpen, der deltakerne bidrar med sine synspunkter. Fasilitatoren begynner diskusjonen med å stille et innledende spørsmål om årsaken og utfordringene angående temaet som er valgt. Det ble ikke gjennomført aktiviteter slik case S gjorde det, og årsaken til dette kan være at teamet er distribuert og derfor anser de dette som vanskelig. Etter diskusjonen er gjennomført, diskuteres forbedringstiltak for utfordringene som teamet står overfor. Fasilitatoren skriver forbedringstiltakene i Confluence, mens resten av deltakerne kommer med forslag til forbedringstiltak.

I andre retrospektivmøtet i case D1 har fasilitatoren i samråd med deltakerne valgt å diskutere arbeidsprosess og effektivitet. Under diskusjonen blir kommunikasjonsutfordringene i distribuert team tydeligere. U_{D1} utdyper utfordringen i forhold til arbeidsprosessen:

"Kanskje oppfatterer dere at det ikke er noe poeng i oppdatere Jira og det synes jeg er kjempedumt."

Når det gjelder effektivitet har U_{D1} utdypet at avhengighetene mellom de distribuerte teamene reduserer effektiviteten:

"Jeg føler at alt jeg har gjort i de to siste ukene har måtte koordinert med [navn på tre distribuerte deltakere], da har jeg i praksis vært fryktelig mindre effektiv enn jeg kunne ha vært i sprinten hvis vi satt sammen."

En annen deltaker, U_{D1} utdyper at kommunikasjonsutfordringene hindrer effektiviteten:

"Årsaken til forsinkelsen er kommunikasjon, der jeg prøver å få kommunisert litt rundt omkring. [...] Jeg har sittet med å arbeidet med [funksjonalitet] og alt de greiene der, og sikkert burde ha vært i [hos sin distribuerte team] i et par dager i uka i stedet"

Deltakerne i case D1 reflekterte over utfordringen og kom fram med følgende forslag for å løse problemet ovenfor:

"Jeg tror det vi kan gjøre er å vurdere å reise mer, når vi har konkrete behov, altså når vi skal jobbe med noe som er veldig spesifikk - jobbe og sitte sammen."

Effektiviteten til teamet har også blitt påvirket av kunden som skildres av U_{D1} :

"Det er skrevet 12 linjer med kode som spesialbehandler mellom valget [to teknologier]. Det skaper problemer [...] Kunden vil diskutere teknologivalget før teamet setter igang. Kunden har avtalt møte på fredag. Dersom kunden ikke tar valget, må ledelsen gripe inn fordi vi kan ikke vente lengre enn fredag."

6.1.5 Trinn 5: Planlegging for gjennomføring av forbedringstiltak

I dette trinnet skal deltakerne i retrospektivmøte planlegge hvordan teamet skal gjennomføre forbedringstiltakene. Både case S og case D1 har planlagt hvordan de skal gjennomføre forbedringstiltakene, men gjennomføres vanligvis ikke grundig.

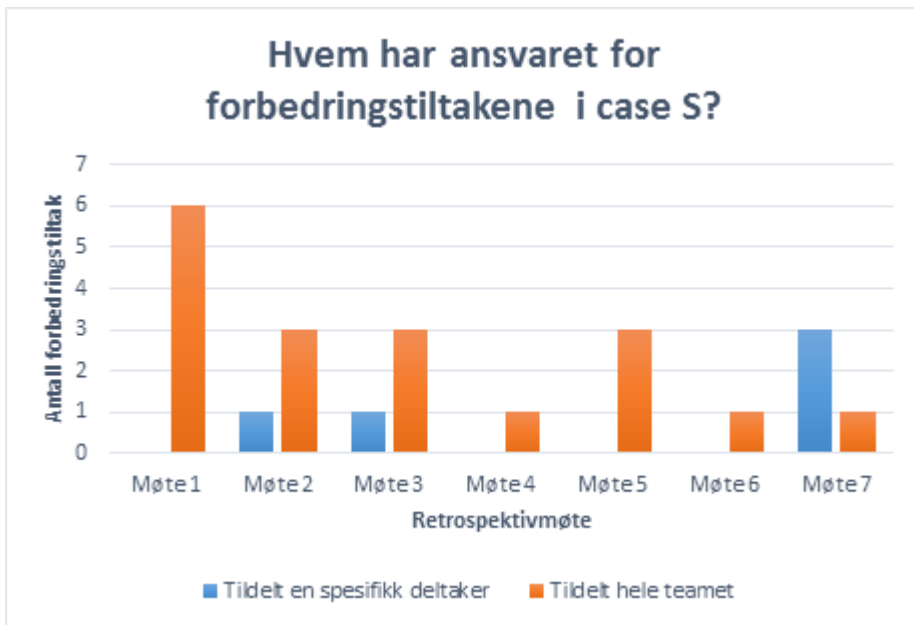
I case S har forbedringstiltakene blitt utarbeidet i hver retrospektivmøte, men selve planlegging av hvordan disse forbedringstiltakene skal følges opp har vært fraværende, bortsett fra når de har hatt fokuserte retrospektivmøter. Eksempelvis ble *"Mindre bugs"* utarbeidet i andre retrospektivmøte som ett forbedringstiltak for å fjerne tekniske feil fra produktet. En annen utfordring har vært oppfølgingen av codereview-tiltaket i andre retrospektivmøtet: *"Ingen har tatt på ansvaret for å gjennomføre codereview. Jeg har til og med sagt det på standup."* Det har også forekommet at deltakerne har utsatt utarbeidelse av enkelte forbedringstiltak da de venter på informasjon fra ledelsen som innebærer flere ressurser (deltakere) i teamet. Ansvaret for gjennomføring av forbedringstiltakene har blitt tildelt til hele teamet bortsett fra fem ganger da har forbedringstiltaket blitt tildelt til en spesifikk deltaker i teamet, se figur 6.6. I følge SM_S er dette en av årsakene til at forbedringstiltakene ikke blir gjennomført:

"Det er slik at hvis man tildeler en oppgave til en ansvarlig så øker sjansen med 30 % og hvis vi setter en tidsfrist så øker sjansen dramatisk for det blir gjort."

Når deltakerne i case S hadde fokuserte retrospektivmøter, så planlagte de nøyaktig hva som skulle gjøres for at disse forbedringstiltakene skal gjennomføres. Planleggingen ble gjennomført ved at fasilitator stilte spørsmål til deltakerne om hvordan forbedringstiltakene skal gjennomføres, der deltakerne kom med forslag. I retrospektivmøtet fikk deltakerne beskjed om at en nøkkeldeltaker skal slutte i teamet. Derfor hadde teamet fokus på hvordan de skal overta kompetansen til denne deltakeren før nøkkeldeltakeren sluttet. Deltakerne i teamet kom fram til tre mål, som hadde konkrete oppfølgingspunkter for hvordan kompetanseoverføringen skulle foregå. Ett av oppfølgingspunktene var:

"Det står på sånn liste at man skal gjøre minst 3 oppgaver per person også skal man velge oppgaver som man ikke er komfortabel med å gjøre og få opplæring i det og hjelp fra [navnet på nøkkeldeltaker]. Dersom noen av de oppgavene du har er av den karakter, så passer det godt å ta dem, hvis det er veldig enkelt ting som vi uansett hadde klart så kan det settes til side"

I case D1 har deltakerne kun utarbeidet forbedringstiltak for ett av to retrospektivmøter, der hele teamet har fått ansvaret for forbedringstiltakene. Bakgrunnen for forbedringstiltakene ikke ble utarbeidet i det ene retrospektivmøtet var fordi de brukte for mye tid på å diskutere utfordringene teamet står overfor. I det andre retrospektivmøtet ble det utarbeidet forbedringstiltak og utformet oppfølgingspunkter for hvordan disse skal gjennomføres.



Figur 6.6: Hvor ofte forbedringstiltak tildeles til et team eller en spesifikk deltaker

6.1.6 Trinn 6: Avslutt retrospektiv

Dette trinnet innebærer at deltakerne dokumenterer utfallet av retrospektivmøtet og gjennomfører en refleksjon på retrospektivmøtet. I case S dokumenteres utfallet av retrospektivmøtet med å ta bilde av whiteboard eller rive av arket som henger på flipover. På denne måten har deltakerne tatt vare på datainnsamlingen fra retrospektivmøtet. Forbedringstiltakene blir notert av SM_S i en notatbok, som senere blir delt med teamet. I case D1 dokumenteres både lappene og forbedringstiltakene på Confluence som gjøres tilgjengelig for alle deltakerne i teamet. Dette vil si at ingen av casene dokumenterer årsaken til hvorfor det gikk bra eller dårlig, men kun forbedringstiltakene.

Når det gjelder refleksjon på retrospektivpraksisen, så har case D1 gjennomført denne aktiviteten på begge retrospektivmøtene. Denne aktiviteten går ut på at fasilitator spør hver deltaker om vurdering av retrospektivpraksisen. Denne vurderingen skal beskrives med ett eller flere ord avhengig av tiden. Deltakeren velger selv om beskrivelsen skal positivt eller negativt. Eksempelvis har U_{D1} fortalt i første retrospektivmøte: *"Jeg blir sliten av å være sitte i to timer, uten pause"*. Dette førte til at neste retrospektivmøtet skal ha en pause på 5-10 minutter.

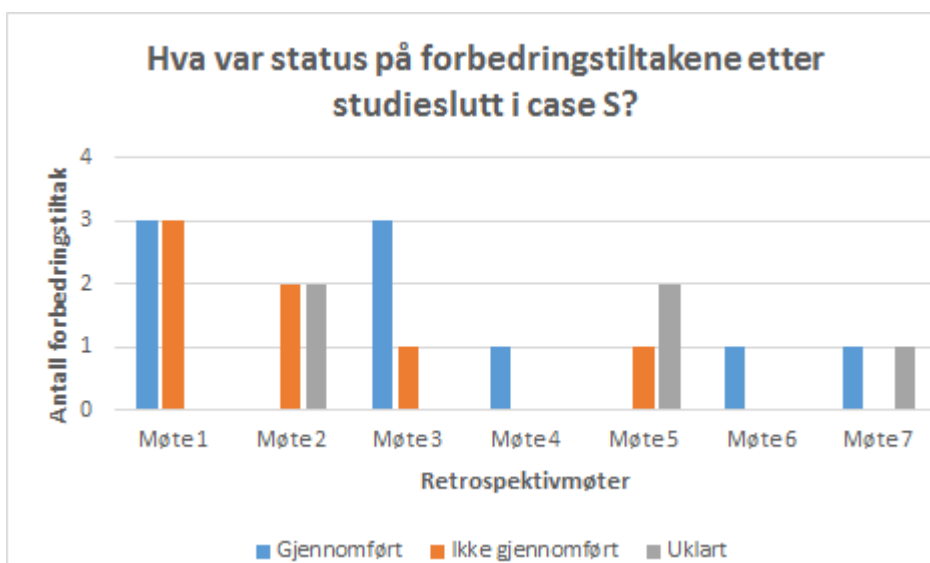
I case S derimot har de kun gjennomført denne aktiviteten én gang, da brukte de Plus/delta modellen som også er en av aktivitetene som Derby og Larsen anbefaler [7]. Der Plus illustrer hva som var de positive sidene, mens delta illustrer hva

som var dårlig sidene. I denne aktiviteten reflekteres hva som var bra med retrospektivmøtet (Plus), og hva som kunne ha blitt bedre (Delta). U_S understreket nytteverdien av det første retrospektivmøtet: *”Dypere refleksjon enn vanlig, mer omfattende referat, mer fokus på læring og mindre på tiltak.”* og en annen deltaker nevnte forbedringspotensialer med retrospektivpraksisen: *”Tidsbruken ble for mye”*.

6.2 Utbytte av retrospektivmøter

Læringen er utbyttet fra retrospektivmøtene. Gjennomføring av forbedringstiltak er en mulighet for å lære fra retrospektivmøtene. Antall enkel -og dobbeltløkkelæring er annen måte å vurdere læringen. Disse resultatene er presentert nedenfor.

6.2.1 Gjennomføring av forbedringstiltak



Figur 6.7: Statusen på forbedringstiltakene ved studieslutt.

Gjennomføringen av forbedringstiltakene kan gi en indikasjon på hvor stor nytteverdi deltakerne har fra retrospektivmøtene. Samtidig bør en være klar over at alle forbedringstiltakene gir ikke like stor nytte, der ett forbedringstiltak kan gi mer nytteverdi i forhold til ett annet. Gjennomføringen av forbedringstiltakene ser ut til å være en utfordring for case S, der kun 42,9 % av forbedringstiltakene har blitt gjennomført, se figur 6.7 og tabell 6.6. Det har ikke vært mulighet til å gjennomføre lignende analyse for case D1 da datamaterialet er begrenset.

Status	Antall	Antall prosent
Gjennomført	9	42,9 %
Ikke gjennomført	7	33,3 %
Uklart	5	23,8 %

Tabell 6.6: Statusen på forbedringstiltakene ved studie slutt i case S.

6.2.2 Enkeltløkke-og dobbeltløkkelæring

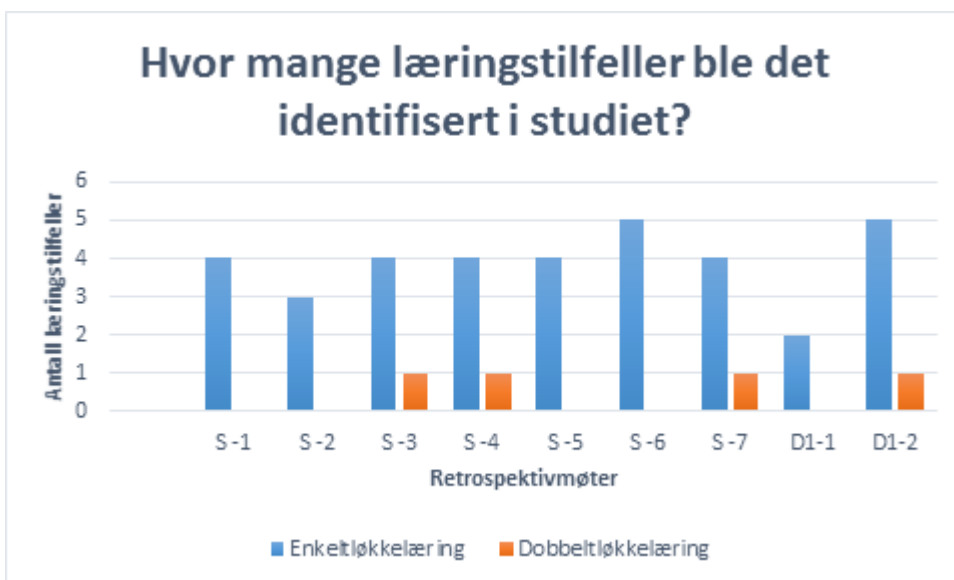
Enkeltløkke-og dobbeltløkkelæring kan være en indikasjon på læringseffekten til deltakerne i retrospektivmøtet. I dette studiet har det blitt identifisert 89,7 % enkeltløkkelæring og 10,2 % dobbeltløkkelæring, se tabell 6.7. Disse tallene understreker at læringen identifiseres i retrospektivmøtene, men det er samtidig viktig å påpeke at deltakerne kan lære mer eller mindre i de ulike læringstilfellene, altså at graden av selve læringen kan variere fra en læringstilfelle til en annen læringstilfelle. Derfor gir ikke tabellen (6.7) eller diagrammet (6.8) det helhetlige bildet av læringen som foregår i retrospektivmøtet.

Case:	Type læring:	Antall:	Prosent:
S	Enkeltløkkelæring	28	90,3 %
	Dobbeltløkkelæring	3	9,7 %
D1	Enkeltløkkelæring	7	87,5 %
	Dobbeltløkkelæring	1	12,5 %
S og D1	Enkeltløkkelæring	35	89,7 %
	Dobbeltløkkelæring	4	10,2 %

Tabell 6.7: Antall enkeltløkke-og dobbeltløkkelæring i case S og D1 i løpet av studiet.

Vi ser på figur 6.8 at deltakerne i case S har tre dobbeltløkkelæring i løpet av studieperioden. I case D1 har ikke dobbeltløkkelæring blitt identifisert i det første retrospektivmøtet, og som tidligere nevnt ble ikke forbedringstiltak utarbeidet. Dette er en imotsetning til det siste retrospektivmøtet der det utarbeidet forbedringstiltak og ble identifisert én dobbeltløkkelæring. Informasjonen om at forbedringstiltaket ble gjennomført i det siste møtet ble formidlet på tilbakemeldingsmøtet fra case S.

I case S har deltakerne hatt fokuserte retrospektivmøter tre ganger, mens case D1 har hatt en gang, se tabell 6.4. Det er i disse møtene dobbeltløkkelæring er identifisert, se figur 6.8.



Figur 6.8: Antall enkeltløkke-og dobbeltløkkelæring i case S og case D1.

Eksempler på enkeltløkke-og dobbeltløkkelæringer

Det gis eksempler på enkelt-og dobbeltløkkelæring i avsnittene nedenfor. Først gjengis to eksempler på enkeltløkkelæring fra case S og D1, før samtlige dobbeltløkkelæring som ble identifisert i case S illustreres. Det kan samtidig nevnes disse sitatene tilhører trinn 4 - diskusjon og utarbeide forbedringstiltak.

Det er nedenfor gitt samme eksempel på enkeltløkkelæring i begge casene.

U_S beskriver situasjonen som følgende i andre retrospektivmøte:

"Det finnes flere saker [oppgaver] om den samme saken, og det hjelper ikke å søke etter saken før det opprettes en ny sak fordi det er beskrevet ulikt."

U_{D1} beskriver lignende situasjon i andre retrospektivmøtet:

"Ofte er det duplikater av oppgaver eller at vi ser at ting ikke henger sammen i Jira."

Forbedringstiltaket for denne utfordringen blir å behandle symptomene framfor underliggende årsaker. Det betyr i praksis at begge teamene bestemte seg for at forbedringstiltaket ble å rydde sprintkøen i Jira.

Dobbeltløkkelæring betyr at deltakerne håndterer underliggende årsaker til utfordringen. Samtlige dobbeltløkkelæringer i case S som identifisert er gjengitt i kronologisk rekkefølge som eksempler på dobbeltløkkelæring.

I tredje retrospektivmøte har deltakerne hatt store utfordringer knyttet til å finne feil (bug) i programvaren som de utvikler og illustreres av PL_S:

"Vi kan finne mange effektive metoder som skal funke i teorien, men i hverdagen raser vi sammen på grunn av bugs. [...] Vi må skrike litt høyere for å få litt flere ressurser."

Mangel på deltakerne ser ut til å være utfordringen som hindrer framtiden deltakerne, men dette stemmer ikke inntil U_S forklarer utfordringen sin:

"Jeg vet ikke alle bruksscenarioer for kundene, og jeg kan ikke [navnet på arbeidsprosessen] og alle de måtene de blir brukt på, og derfor er det enkelt å ta feil avgjørelser."

Dette er dobbeltløkkelæring fordi deltakerne finner den underliggende årsaken til hva som hindrer framdriften deres, nemlig manglende domenekunnskap. Det hjelper ikke med flere ressurser, dersom en deltaker produserer flere feil i programvaren ved å ha manglende domenekunnskap.

I fjerde retrospektivmøtet diskuterte deltakerne utfordringene knyttet til kunden. U_S beskriver situasjonen som følgende:

”Mange av de problemsstillingene vi får opp skyldes nye kunder – oppstartssituasjoner. Altså at det går en stund, så kommer det tilbake, så blir som å skyldes på at det var en ny kunde med spesielle behov. [...] Alle våre kunder fører [arbeidsprosess] på forskjellige måter.”

SM_S reflekterer over den underliggende årsaken og kommer med ett forslag til hvordan slike situasjoner kan håndteres:

”Forslag for å komme videre kan være at vi ser på oppstartsrutiner i forhold til [arbeidsprosessen til kunden]. Det kan fange opp noe tenker jeg. Dette bør vi gjøres før kunden blir kunde.”

Dette er dobbeltløkkelæring fordi den reflekterer over de underliggende årsakene til hvorfor kunden melder fra feil når de først har kjøpt produktet. Enkeltløkkelæring i dette tilfellet ville ha vært at kunden melder inn feilen og utviklerne retter opp feilen uten å undersøke de underliggende årsakene til utfordringen.

I det syvende retrospektivmøtet diskuterte deltakerne utfordringer knyttet til kommunikasjonssvikt. Denne utfordringen ble fremstilt av PL_S:

”På onsdag var vi enig om å legge ut (release) [produktnavn] og på torsdag som var det slik ’nei, vet ikke’, og det var ingen som visste at vi skulle legg ut og dere har tydeligvis ikke snakket om det på dagen heller.”

I dette tilfellet ser utfordringen til å være kommunikasjonssvikt mellom prosjektlederen og deltakerne i teamet. Dette er ikke tilfelle, der den underliggende årsaken forklares av U_S:

”Jeg savner klarer rutiner blant annet når vi skal release ting. Altså hvem som er ansvarlig for å få det dokumentert eller hvordan finner vi ut hva vi skal gjøre. Slik at vi slipper å finne eller spekulere hvem som skal legge ut? Eller skal vi gjøre det eller skal han gjøre det?”

Prosjektlederen beskriver utfordringen som kommunikasjonssvikt, men det viser seg at manglende oversikt over ansvarsområde er den underliggende årsaken til utfordringen. Forbedringstiltaket ble å definere roller på de ulike ansvarsområdene og tildele det til en bestemte deltakere.

Kapittel 7

Diskusjon

I diskusjonskapitlet skal forskningsspørsmålene bevares og derfor er kapitlet strukturert etter forskningsmålene. Dette betyr at hvert delkapittel besvarer hvert sitt delspørsmål. Samtidig vil jeg understreke som nevnt tidligere at kommunikasjon er et supplement, og hovedfokuset ligger på organisatorisk læring. Derfor vil diskusjonskapitlet om kommunikasjon være betydelig mindre sammenlignet med organisatorisk læring. Det overordnet problemsstillingen er å *undersøke utfordringer knyttet til organisatorisk læring og kommunikasjon i smidig utviklingsmiljø med utgangspunkt i retrospektivmøter*. Problemsstillingen er spesifisert i to delspørsmål:

1. *Hvilke utfordringer har systemutviklingsteam i forhold til å lære gjennom retrospektivmøter?*
2. *Hvilke utfordringer har distribuerte team sammenlignet med samlokalisert team når gjelder kommunikasjon?*

I første delspørsmålet beskriver hvilke utfordringer systemutviklingsteam har i forhold til å lære gjennom retrospektivmøter ved bruk av organisatorisk teori. I dette delkapittel brukes læringsmodellene til Argyis og Schön, modell I og modell II som også omfatter enkeltløkke- og dobbeltløkkelæring for å beskrive utfordringer knyttet til organisatorisk læring i et smidig utviklingsteam.

I andre delspørsmål diskuteres kommunikasjonsutfordringene mellom samlokalisert og distribuert team. Kommunikasjonsutfordringer inkluderer koordinering, da disse er to sider av samme sak.

7.1 Organisatorisk læring

I dette kapitlet bevares forskningsspørsmålet om utfordringer knyttet til organisatorisk læring i retrospektivmøter. Først brukes modellene til Argyis og Schön, som er gjengitt i kapittel 3.4 til å diskutere funnene i studiet. Derfor reflekteres det over utfordringene knyttet til læringen i retrospektivmøtet. Deretter beskrives faktorene som har bidratt til dobbeltløkkelæring, før en modell presenteres basert på funnene i studiet for gjennomføring av retrospektivmøtet.

7.1.1 Model I

Det er fire styringsvariabler i modell I som er beskrevet i kapittel 3.4.1. I dette studiet har jeg funnet at det er ingen av disse styringsvariablene som har blitt identifisert i dette studiet. En nærmere diskusjon om hvorfor disse styringsvariablene ikke ble identifisert blir beskrevet nedenfor. Deretter blir deres konsekvenser for arbeidsmiljø, læring og effektivitet diskutert. En oppsummering blir gjengitt i tabell 7.1.

Den første styringsvariabelen er *definere mål og prøve å oppnå dem*. Disse verdiene som denne styringsvariabelen indikerer har ikke blitt identifisert. I stedet har det blitt observert at deltakerne i retrospektivmøte bidrar til at teamet får en fellesforståelse om problemsstillingen ved å angripe den fra alle synspunkter. På denne måten er deltakerne villig til å endre sin oppfattingen av problemsstillingen og bidrar til å utvikle fellesforståelse om den aktuelle problemsstillingen. I case S har deltakerne hatt flere utfordringene knyttet til bugs i programvaren, der de diskuterer hvorfor disse oppstår. I dette tilfellet har deltakerne utforsket problemsstillingen fra ulike synspunkter og funnet fram til forskjellige årsaker til problemsstillingen.

Den andre styringsvariabelen, *maksimere vinning og minimalisere tap* er heller ikke identifisert i dette studiet. Deltakerne i studiet har gitt klare indikasjoner på at det ikke er en svakhet å endre på et valg som allerede er tatt, dersom valget viser seg å ikke fungere. Med andre ord kan dette beskrives som at det er greit å prøve noe annet hvis den nye iverksatte praksisen ikke fungerer. Eksempelvis har case S en vanlig praksis at temaet som diskuteres i retrospektivmøtet er ikke forhåndsbestemt. Da deltakerne fikk vite at de mister en nøkkeldeltaker i teamet så endret teamet retrospektivpraksisen sin til å ha fokusert retrospektivmøte, der målet var å diskutere hvordan teamet kan ta over kompetansen til deltakeren. Etter denne hendelsen byttet case S tilbake til vanlig retrospektivpraksis, før de igjen endre det tilbake til å ha fokuserte retrospektivpraksis for unngå at de samme temaene blir diskutert gjentatte ganger, som U_S uttrykker: "Vi tømte oss der og da, og det er de samme lappene som går igjen". Ett annet eksempel på at denne styringsvariabelen er fraværende illustreres i case D1, der det i begynnelsen av prosjektet ble bestemt at teamet skal ha fire ukers lange sprinter. Det ble senere gjort store endringer i forhold til denne praksisen ved å ha to ukers mini sprinter i en stor fire-ukers sprint på bakgrunn av at det var vanskelig å planlegge konkrete og detaljerte

oppgaver som skulle gjennomføre i løpet av sprinten. Den store sprinten er fire uker lang fordi team skal i følge kontrakten med kunden ha sprint review hver fjerde uke.

Den tredje styringsvariabelen er *minimalisert generering eller uttrykke negative følelser*. Retrospektivmøter kan brukes til å ytre både positive og negative følelser. I alle casene har jeg observert at deltakerne har utvekslet negative følelser med hverandre. I case S har en deltaker beskrevet av at vedkommende får lite utbytte av retrospektivmøter, da det er de samme utfordringene og temaene som diskuteres gjentatte ganger uten at vedkommende får nytte av det. Konsekvensen av dette er som nevnt ovenfor at teamet gikk over til å ha fokuserte retrospektivmøter. Dette viser at en kan lære av negative følelser, men de bør ikke være en arena der negative følelser tømmes, da dette kan påvirke atmosfæren i retrospektivmøtet. I case S har deltakerne hatt mer fokus på negative utfordringer i forhold til positive utfordringer som gjengis av U_S: *"Det har vært mye negativt at det ikke blir noe positivt"*. Det kan neves at positive utfordringer er minst like viktig som negative utfordringer, da det kan bidra til å oppmuntre deltakerne i teamet [7, 14].

Den siste styringsvariabelen er *å være rasjonell*. I et retrospektivmøtet blir deltakerne oppmuntret til å dele både positive og negative følelser med resten av teamet. Det å ha ett godt fungerende team som er inkluderende, bidrar til at deltakerne føler seg trygge på hverandre slik at både positive og negative følelser kan utveksles med hverandre. Med dette i bakhodet ser det ut til at denne styringsvariabelen er fraværende for samtlige team i dette studiet. Dette samsvarer med Zedwitz som beskriver at et inkluderende team bidrar til å øke læringseffekten [60].

Konsekvenser for arbeidsmiljø, læring og effektivitet

Et resultat av diskusjonen overfor viser at konsekvenser for arbeidsmiljø for modell I oppstår meget sjelden. Det er observert flere tilfeller der deltakerne i samtlige case konfronterer hverandre med sine synspunkter. Disse konfrontasjoner er ikke ment å være forsvarende, men for å danne en diskusjon der deltakerne kan komme fram til en løsning på en utfordring teamet står overfor. Derfor kan det konkluderes med det er ikke identifisert konsekvenser for arbeidsmiljø, når det gjelder å være defensiv, forsvare normer og å ha defensive mellommenneskelige relasjoner.

Når det gjelder konsekvenser for læring, så er de både identifisert og fraværende. Når det gjelder konsekvenser for læring, så er enkeltløkkelæring blitt identifisert flere ganger, noe som er overraskende når ingen av styringsvariablene har blitt identifisert. De andre konsekvensene for læring er fraværende, og dette gjelder mangel på offentlig testing av teorier, mye privat testing av teorier og selvklebende (self-sealing). Bakgrunnen for hvorfor enkeltløkkelæring oppstår når ingen av styringsvariablene er identifisert diskuteres i kapittel 7.1.3.

Det har ikke vært muligheten til å måle effektiviteten og derfor kan ikke konsekvensene av disse beskrives. Det er nærliggende å tro at denne konsekvensen ikke oppstår, på tross av at styringsvariablene ikke ble identifisert.

<i>Model I</i>	<i>Identifisert</i>
Styringsvariabel:	
Definere mål og prøve å oppnå dem	Nei
Maksimere vinning og minimere tap	Nei
Minimalisert generering eller uttrykke negative følelser	Nei
Vær rasjonell	Nei
Konsekvenser for arbeidsmiljø:	
Forsvare aktører	Nei
Forsvare mellommenneskelige og grupperelasjoner	Nei
Forsvare normer	Nei
Konsekvenser for læring og effektivitet:	
Selvklebende	Nei
Redusere langvarig effektivitet	Ikke observert
Enkeltløkkelæring	Ja
Lite testing av teorier offentlige	Nei
Mye testing av teorier privat	Nei

Tabell 7.1: Oppsummering av diskusjonen i forhold til modell I.

7.1.2 Model II

Styringsvariablene i modell II som er beskrevet av Argyis og Schön er identifisert i dette studiet. I avsnittene nedenfor vil funnene fra dette studiet diskuteres i forhold til disse styringsvariablene og deres konsekvenser før det avsluttes med en oppsummeringstabell der resultat fra diskusjonen oppsummeres i tabell 7.2.

Gyldig informasjon er den første styringsvariabelen i modell II til Argyis og Schön. Dette innebærer at det samles inn gyldig informasjon fra deltakerne i retrospektivmøter. Deltakerne i dette studiet har brukt verktøy og post-it lapper samt tidslinje som teknikker for å gjennomføre datainnsamlingen. I case S har U_S erfart en utfordringer i forhold til datainnsamlingen i retrospektivmøter: *"Jeg synes det er vanskelig å huske de ulike tingene som har skjedd i løpet av perioden"*. I dette tilfellet omfatter hukommelsessvikt når deltakeren sliter med å huske hva som har skjedd i løpet av sprinten, men det er også observert at deltakerne i case S sliter med å huske forbedringstiltakene fra tidligere retrospektivmøter. Denne utfordringen er identifisert av Kranstoff som forbinder hukommelsessvikt som en hindring for datainnsamlingen i retrospektivmøtene, og fører til at deltakeren kommer med mindre troverdig data i forhold til dataene som samles under sprinten [58]. På en annen side beskriver Baid at hukommelsen er avtagende med tiden, og det er vanskelig å gjenfortelle informasjonen mest mulig nøyaktig [59]. Deltakerne i case D1 har lengre tidsintervall mellom retrospektivmøtene i forhold til case S, og de har ikke gitt indikasjon for vanskeligheter med å huske informasjon fra dette tidsperspektivet. Med dette i bakhodet kan det en argumentere for at dette er en utfordring knyttet til case S.

Mangel på gyldig informasjon har ført til at deltakerne i retrospektivmøte utsetter diskusjonen på bestemte utfordringer eller utarbeidelse av bestemte forbedringstiltak inntil de har informasjonen tilgjengelig. Eksempelvis har SM_S beskrevet følgende: *"Ser an med nye tiltak når vi får dekk med nye ressurser"*. Sitatet formidler at case S har utsatt utarbeidelse av forbedringstiltak på bakgrunn av at de ventet på informasjon fra ledelsen om flere ressurser, slik at de kan basere seg på denne informasjonen. I tillegg har enkelte deltakere valgt å ikke delta på retrospektivmøtet, særlig gjelder dette nyansatte. Dette har ført til at teamet mister informasjonen, se kapittel 6.1.1. Informasjonen kunne ha bidratt til å identifisere og styrke utfordringer og behov. Dingsøyr påpeker at alle som har bidratt til prosjekt bør være med i møtet, bortsett fra kunden og ledelsen [14]. Dubinsky og Hazzan argumenterte for at hele teamet bør aktivt delta i retrospektivmøtet både før og etter retrospektivmøtene [63].

Frie og informert valg er den andre styringsvariabelen i modell II til Argyis og Schön. Denne styringsvariabelen betyr i forhold til retrospektivpraksisen at teamet står fritt til å velge hvordan de gjennomfører retrospektivmøte for å forbedre arbeidspraksisen sin. I følge smidig manifestet skal teamet være selvstendige til å ta sine egne valg uten store inngrep fra utenforstående som også inkluderer ledelsen [50]. Denne interessen ser ut til å være ivaretatt da ingen av deltakerne i studiet har gitt signaler om at de ikke har tid til å gjennomføre retrospektivmøter eller at de ikke får tillatelse til å gjennomføre retrospektivmøter fra ledelsen. I stedet har det blitt observert at deltakerne selv bestemmer hvordan de selv ønsker å gjennomføre retrospektivmøter slik at de får mest mulig læringsutbytte av denne prosessen. Dette samsvarer ikke med Glass som viser til at deltakerne gjennomførte ikke retrospektivmøte på bakgrunn av at ledelsen nedprioriterte retrospektivpraksisen [10].

Teamet står fritt til å gjennomføre de valgene de selv måtte ønske, bortsett når det involverer utenforstående. Dette innebærer at teamet tar avgjørelser som ikke ligger innenfor deres beslutningsområde, slik det var tilfelle i case S der deltakerne diskuterte forbedringstiltak for å få flere teammedlemmer i teamet. Deltakerne har trossalt ikke myndighet til å gjennomføre dette forbedringstiltaket, men kan legge det fram ønsket for ledelsen. Dette oppleves som en hindring for deltakerne i case S. I case D1 har teamet utarbeidet et forbedringstiltak som krever endringer fra eksterne parter som i dette tilfellet er kunden. Forbedringstiltaket indikerer at kunden skal være effektiv med å svare på henvendelser, noe som har vist seg å være vanskelig slik U_{D1} gjengir situasjonen: *"De er tydeligvis i en purrekultur. Dem gjør ikke noe før dem blir purret minst én gang. [...] Jeg tror det er kulturen deres og ikke noe vi kan gjøre noe med"*. Dette betyr at det er vanskelig for teamet å endre bedriftskulturen til teamet. En direkte konsekvens av denne purrekulturen til kunden har ført til forsinkelser når det gjelder teknologivalg for teamet, der kunden ønsker å diskutere valget før teamet setter i gang, se kapittel 6.1.4. Det er ikke identifisert andre tilfeller som hindrer at teamet gjennomfører denne styringsvariabelen bortsett fra disse to tilfellene.

Interne forpliktelser og konstant evaluering av handling i forhold til valget er den siste styringsvariabelen i modell II til Argyis og Schön. Denne styringsvariabelen har vist seg å være en utfordring for samtlige case i dette studiet. Dette betyr i praksis at utviklingsteamene har en utfordring forbundet med å gjennomføre forbedringstiltak, og dermed se forbedringspotensialet av dette forbedringstiltaket, eksempelvis ble omtrent 33 % av forbedringstiltakene i case S ikke gjennomført ved studieslutt. Når forbedringstiltakene ikke gjennomføres fører til det at teamet får ikke nytte av å gjennomføre retrospektivmøtet som U_S indikerer: *"Det er ofte de samme lappene som går igjen, uten at noe blir gjort"*. Bakgrunnen for dette er at teamet utarbeider forbedringstiltak som ikke blir fulgt opp, og samtidig evalueres forbedringstiltakene relativt sjelden. Det er tidligere nevnt at deltakerne i case S har hatt en aktivitet i et retrospektivmøte, der de evaluerte sin egen innsats for å følge opp forbedringstiltak som viser at de gjorde minimal innsats se kapittel 6.1.2.

Forbedringstiltakene er interne forpliktelser som deltakerne tar på seg, og som tidligere nevnt har det vært vanskeligheter med å håndheve disse. Utfordringene i forhold til gjennomføring av forbedringstiltakene utdypes i neste delkapittel 7.1.3. Deltakerne i studiet forstår konsekvensen av å ikke gjennomføre forbedringstiltak ved å ha ett fokus mot å oppretteholde utfordringer. Eksempelet ovenfor illustrerer dette poenget godt, der deltakeren beskriver at de får ikke løst utfordringene ved å gjennomgående diskuterer de samme temaene.

Konsekvenser for arbeidsmiljø, læring og effektivitet

De tre styringsvariablene i modell II har konsekvenser som både er identifisert og er fraværende for casene som deltar i dette studiet.

Den første konsekvensen for arbeidsmiljø er at deltakerne i retrospektivmøte betraktes som lite defensive. Dette innebærer at deltakerne er åpne for sine synspunkter og i minst mulig grad prøver å skjule informasjon for å beskytte seg selv. Generelt har deltakerne i teamet vært villig til å dele sine svake sider med de andre deltakerne i teamet, noe som gir en god indikasjon på at deltakerne er minst mulig defensive i retrospektivmøter. I case S har deltakerne vært åpne og ærlige i forhold til deres arbeidssituasjon som har innvirkning på fremdriften til teamet og som illustreres av U_S: *"Det er tidskrevende å finne feil i produktet – jeg vet ikke hva jeg skal lete etter"*. En kan argumentere for at dette er dårlig bruk av tid, men på en annen side førte dette til at teamet utarbeidet testplan slik at vedkommende kan forstå hva som skal testes.

Lite defensive mellommenneskelig og gruppedynamiske relasjoner er den andre konsekvensen for arbeidsmiljø. Det er nevnt ovenfor at deltakerne har i liten grad blitt betraktet som defensiv. Dette fører til at det har ikke vært tosidig defensive konfrontasjoner, men i stedet har deltakerne tosidig samarbeidet for å forbedre arbeidsprosessen i teamet. Bakgrunnen for samarbeidet fremheves av U_S: *"Vi kan spørre hverandre og ta beslutninger basert på informasjon fra hverandre og stole på beslutninger på bakgrunn av det"*. Når deltakerne har tillitt til hverandre i teamet

er det enklere for den enkelte deltakeren i teamet å komme opp med en utfordring deltakeren selv står overfor og som påvirker teamet som helhet. I case S har deltakerne hatt store utfordringer forbinde med å finne feil (bugs) i programvaren de utvikler, og årsaken til denne utfordringen har vært vanskeligere å finne dersom deltakerne var defensive. Årsaken til denne utfordringen var domenekunnskap der deltakeren ikke kjente til bruksscenarioer for kunden, se kapittel 6.2.2 for nærmere beskrivelser. Med dette i bakhodet kan en argumentere for at minimal defensiv mellommenneskelig og gruppedynamiske relasjoner er nært beslektet med tilliten deltakerne har til hverandre i teamet.

Læringsorienterte normer er den tredje konsekvensen for arbeidsmiljø. Retrospektiv er en kollektivlæringsarena, der deltakerne er i stand til å reflektere over synspunkter og dele dette med resten av teamet. Dette gjøres ved at deltakerne finner forbedringstiltak for behovene og utfordringene teamet står overfor ved å gjennomføre retrospektivmøtet og som det tidligere er gitt eksempler på.

Den siste konsekvensen for arbeidsmiljø er høy grad av valgfrihet, indre forpliktelser og risikotaking. I dette studiet har det vist seg, som nevnt tidligere at deltakerne har høy grad av valgfrihet så lenge det er innenfor deres beslutningsområde. Forbedringstiltak er interne forpliktelser, og er en utfordring som nevnt tidligere. Dette er fordi det mangler realistisk struktur for oppfølging samt eierskap for å gjennomføre forbedringstiltakene.

Det er tre konsekvenser for læring, oftere testing av teorier i fellesskap, avkrettesprosess og dobbeltløkkelæring hvor alle har blitt identifisert. Retrospektivpraksisen er en arena, der arbeidsprosessen til teamet kan bli kontrollert i fellesskap og som illustreres med å gjengi ett eksempel fra case S i kapittel 6.1.4. Teamet bestemte seg for å ha en fast scrum-master, og som senere ombestemte scrum-masteren seg da det ble tildelt oppgaver som ikke var utenfor scrum-masterens ansvarsområde. Belastningen ble for mye for vedkommende, og derfor bestemte teamet seg for å rullere scrum-master rollen slik at en deltaker var scrum-master for en bestemt tidsperiode. Denne praksisen fungerte heller ikke og derfor bestemte teamet seg til å gå tilbake til forrige den praksisen som følge av: *"Jeg synes det fungerte bedre når vi hadde en person over lengre tid og at det blir litt ansvarsfraskrivelse når det går på rundgang"*. Drury et al. identifisert også at teammedlemmene ønsket ikke å ta avgjørelser, og derfor ble denne oppgaven tildelt scrum-masteren [62].

Den siste konsekvensen for læring er dobbeltløkkelæring som har blitt identifisert i case S og D1. På en side har det vært en diskusjon hvorvidt det kan kalles dobbeltløkkelæring, der deltakerne klarer å finne underliggende årsakene, men gjør ikke tilstrekkelig nok innsats for å implementere forbedringstiltaket som er utarbeidet i retrospektivmøtet. Det er bestemt at implementering av forbedringstiltakene er nødvendig for å oppnå dobbeltløkkelæring. Bakgrunnen for hvorfor dobbeltløkkelæring oppstår i retrospektivmøte diskuteres i kapittel 7.1.4.

Økt langvarig effektivitet er den eneste konsekvensen for effektivitet når det gjelder styringsvariablene. Formålet med retrospektivmøte er kollektiv læring som fører til at teamet klarer å løse utfordringene og behovene de står overfor. En kan argumentere for at teamet skal være i stand til å ha en økt og langvarig effektivitet ved å gjennomføre retrospektivmøter. På en annen side har deltakerne i case D1 beskrevet purrekulturen til kunden, som har begrenset effektiviteten til teamet, se 6.1.4. I dette studiet har det ikke vært mulighet til å undersøke den langvarige effektiviteten til de ulike casene, men deltakerne selv har sagt at de har fått liten nytte av å gjennomføre retrospektiv, da de samme utfordringene gjentas. Derfor kan en argumentere for at teamet ikke har fått økt langvarig effektivitet, men på en annen side har teamene dannet dobbeltløkkelæringer.

<i>Model II</i>	<i>Identifisert</i>
Styringsvariabel:	
Gyldig informasjon	Utfordring for case S, men ikke for case D1.
Frie og informert valg	Ja
Interne forpliktelser	Utfordring
Konsekvenser for arbeidsmiljø:	
Lite defensiv	Ja
Lite defensive mellommenneskelige relasjoner	Ja
Læringorienterte normer	Ja
Høy grad av valgfrihet og interne forpliktelser	Ja, men interne forpliktelser er en utfordring
Konsekvenser for læring:	
Oftere testing av teorier i felleskap	Ja
Avkrettingsprosess	Ja
Dobbeltløkkelæring	Ja
Oftere offentlig testing av teorier	Ja
Konsekvenser for effektivitet:	
Økt langvarig effektivitet	Ingen mulighet til å undersøke.

Tabell 7.2: Oppsummering av diskusjonen i forhold til modell II.

7.1.3 Refleksjon rundt læringsprosessen

Det er som tidligere nevnt blitt identifisert både enkeltløkke-og dobbeltløkkelæring i case S og D1. Selv om antall enkeltløkke-og dobbeltløkkelæring ikke gir det helhetlig bilde av læringen som foregår i et retrospektivmøtet, så gir det indikasjon på forholdet mellom enkeltløkke-og dobbeltløkkelæring. I tabell 6.7 viser det seg at 89,7% av tilfellene er enkeltløkkelæring og 10,2 % er dobbeltløkkelæring. Det at det oppstår flere enkeltløkkelæring i forhold til dobbeltløkkelæring er som forventet, da det er vanskeligere å få dannet dobbeltløkkelæring i et retrospektivmøte og er i henhold til Argyis og Schönens teori [1]. I kapittel 6.2.1 og 7.1.2 viste det seg at den største hindringen for å danne dobbeltløkkelæring har vært implementeringen av forbedringstiltakene, noe som har ført til at deltakerne har ikke fått fullstendig nytte av retrospektivmøtene. På en annen side har det blitt dannet dobbeltløkkelæring og bakgrunnen for dette diskuteres i kapittel 7.1.4. I dette delkapittel fokuserer vi på hvorfor implementering av forbedringstiltakene har vært den største hindringen for å danne dobbeltløkkelæring samt hvorfor enkeltløkkelæring forekommer oftere i forhold til dobbeltløkkelæring.

Det er vanlig at det oppstår flere enkeltløkkelæring i forhold til dobbeltløkkelæring i et retrospektivmøte. Bakgrunnen for dette er at det er ikke realistisk å gjennomføre en grundig diskusjon eller årsaksanalyse på samtlige utfordringer teamet står overfor, da dette krever mye tid. Tid er kostbart i en organisatorisk sammenheng, og derfor er det viktig at deltakerne i retrospektivmøter gjennomfører grundig diskusjon og analyse på de viktigste utfordringene som teamet står overfor. Birk et al. formidler at deltakerne må prioritere de viktigste utfordringene og kun gjennomføre årsaksanalyse på disse, og ikke på alle utfordringene [11]. Derfor kan det tenkes at de fleste utfordringene ender som enkeltløkkelæring, der det kun forslås en handling eller en metode for å løse utfordringene det står overfor, framfor å diskutere årsaken til utfordringen.

Det er tidligere nevnt at eksterne aktører kan være en hindring for dobbeltløkkelæring. I delkapittel 7.1.2 har jeg vist til eksempler for hvordan eksterne aktører vanskeliggjør med å gjennomføre forbedringstiltak. Når læringen inkluderer eksterne aktører som kunden, så vil dette føre til at det kreves innsats fra disse, noe som har vist seg å være vanskelig, og derfor ender læringsprosessen som enkeltløkkelæring. Ett godt eksempel på denne situasjonen er illustrert i case D1:

SM_{D1}: *"Oversikt over miljøer og servere savnes. Er dette noe dem [kunden] skal lage?"*

U_{D1}: *"Dette er noe kunden skulle ha gjort siden siste retrospektiv og det viser at dette har vært ett tiltak fra forrige retrospektiv".*

I dette studiet har vist seg at interne forpliktelser og konstant evaluering av disse har vært en utfordring for spesielt case S og er derfor en hindring for å danne flere dobbeltløkkelæring. Med andre ord er implementering av forbedringstiltak vært en utfordring for å danne flere dobbeltløkkelæring. Voteringen er en teknikk som har blitt brukt for å prioritere hvilke temaer som ønskes å diskutere ytterligere. I følge Lehtinen er votering godt egnet for å prioritere utfordringer [52]. På denne måten har deltakerne implisitt valgt ut hvilke utfordringer de ønsker å diskutere dypere for å finne de underliggende årsakene til utfordringen som danner grunnlaget for dobbeltløkkelæring. I case S har det vist seg at teamet klarer å finne disse underliggende årsakene, men at forbedringstiltakene som blir utarbeidet basert på disse blir ikke gjennomført, se 6.2.1. Dette hindrer teamet fra å lære fra disse forbedringstiltakene eller handlingene og resulterer at teamet ikke forbedrer sin praksis. Salo viser til at gjennomføringen av forbedringstiltak er nødvendig for å få utbytte av retrospektivmøter og for å ta læring av dette [19]. Siden deltakerne ikke gjennomfører forbedringstiltak er det nærliggende å tro at temaene diskuteres gjentatte ganger se tabell 6.5.

I sjetten retrospektivmøte som ble observert i case S, endret de praksisen der de gjennomførte en kollektiv refleksjon på hva utbyttet av de tidligere retrospektivmøtene har vært, og der kommer det fram som nevnt tidligere at utbyttet har vært liten. Dette er fordi deltakerne har diskutert de samme problemsstillingene gjentatte ganger uten å ta godt utbytte av det. Myllyaho et al. gjennomførte en gjennomgang av studier (review) og beskrev at deltakerne ”snakket ikke bare med å endre sin måte å arbeide på, men de faktisk endret måten arbeide på” ved bruk av retrospektivmøter [61]. Stålhane et al. identifiserte behovet for å ha ett godt og planlagt retrospektivmøte. Med dette menes at møtet bør være strukturert slik at deltakerne ikke diskuterer alt som har foregått i tidsperioden [55]. Derfor kan det tenkes at dette var årsaken til at case S endret retrospektivpraksisen sin til å ha fokuserte retrospektivmøte, der scrum-master på forhånd bestemte temaene for retrospektivmøtet, slik at de unngår at de samme temaene dukker opp igjen hver retrospektivmøte.

Utfordring	Beskrivelse
Mange forbedringstiltak	Det fokuseres på mange forbedringstiltak samtidig og det kan være praktisk vanskelig å gjennomføre forbedringstiltakene.
Mangelende konkrete forbedringstiltak	Når forbedringstiltakene ikke er målebare, er det vanskelig å kunne måle effekten eller utbyttet av forbedringstiltak, og dermed vanskelig å fastslå om utfordringen er løst.
Manglende oppfølging og tilrettelegging	Forbedringstiltak påbegynnes, men gjennomføres ikke fullstendig for å oppnå nytteverdien av forbedringstiltaket, mens andre forbedringstiltak er det usikkerhet om hvem som skal gjennomføre disse.
Involvering fra eksterne aktører	Forbedringstiltaket innebærer involvering fra eksterne aktører og krever at de endrer sin praksis eller deres beslutning.

Tabell 7.3: Bakgrunnen for at forbedringstiltak ikke gjennomføres

I enkelte tilfeller har deltakerne i case S definert for mange forbedringstiltak og som følge av dette er det vanskelig å gjennomføre alle forbedringstiltakene. En av konsekvensene av å ha for mange forbedringstiltak kan være at slik U_{D1} understreker: *"Teamet bør ikke gjøre for mange endringer da det kan være vanskelig å se effekten av denne endringen"*. Denne utfordringen er også gjenspeilt i case S, der U_S poengterer: *"Grunnen til at vi er sjeldent etter plan er fordi vi planlegger masse på retrospektiv og det viser seg at dette ikke blir gjort"*. Birk et al. indikerer behovet for å prioritere utfordringene og deretter kun utarbeide forbedringstiltak på de viktigste utfordringene [11]. Dingsøyr beskrev at deltakerne fikk nytte av å prioritere utfordringene og deretter kun utarbeide forbedringstiltak på de viktigste utfordringene ved å bruke årsaksanalyse [14].

I flere tilfeller har jeg observert at manglende konkrete forbedringstiltak er nødvendig for at deltakerne i retrospektivmøte skal kunne måle effekten eller utbytte av forbedringstiltaket. I case S har deltakerne i flere tilfeller utarbeidet forbedringstiltak knyttet til bugs som ikke var målbare. Et godt eksempel er beskrevet av U_S : *"Mindre bugs er vel ett tiltak"*. I denne sammenhengen foreslås det ikke hvilken konkrete metode eller handlinger som skal gjennomføres for å måle effekten av forbedringstiltaket. Druyr et al. har funnet at deltakerne er ikke motiverte til gjennomføre retrospektivmøte, da de får ikke utbytte av det [62]. Ved å ha mulighet til å måle effekten av forbedringstiltaket synliggjøres behovet for nytteverdien av retrospektivmøtet. Dubinsky et al. uttrykker behovet for å måle effekten av forbedringstiltaket [63], slik at nytteverdien blir synliggjort i praksis .

En annen hindring for gjennomføring av forbedringstiltak er manglende oppfølging eller tilrettelegging for forbedringstiltakene. På en side fører dette til at deltakerne gjennomfører forbedringstiltakene halvveis, men gjennomfører den ikke fullstendig og oppnår dermed ikke nytteverdien. Et eksempel var når deltakeren i case S hadde manglende domenekunnskap og dermed var på kurs for å øke domenekunnskapen. Deltakeren endte med å arbeide med noe annet som ikke krevde denne kunnskapen og da deltakeren i fikk bruk for denne kunnskap senere, var kunnskapen glemt, se kapittel 6.1.4. Dette viser at forbedringstiltaket er halvveis gjennomført, og derfor er dette en hindring for å få fullt utbytte av læringsprosessen. I case S har de fleste forbedringstiltakene blitt tildelt til teamet som helhet. Drury et al. viser til at en av årsakene til forbedringstiltakene ikke gjennomføres er usikkerheten rundt hvem som har ansvaret [62]. Denne utfordringen er også gjenspeilt i case S, der en deltaker tror at en annen deltaker skal gjennomføre forbedringstiltaket og ender med at resultat er ingen av dem gjør det. Et eksempel på dette er tidligere nevnt i forhold til case S hvor en deltaker beskriver at ingen har tatt på seg ansvaret for å gjennomføre code review. Myllyaho et al. viser at forbedringstiltakene gjennomføres dersom det blir tildelt til en spesifikk deltaker som har ansvaret for gjennomføring av forbedringstiltaket [61]. I case S har scrum-master rollen vært rullerende og ansvaret for retrospektivmøtet ligger hos rolleinnhaveren. Derfor kan det tenkes at det har vært ansvarsfraskrivelse når rollene har blitt skiftet og dermed har ikke forbedringstiltakene blitt fulgt opp.

Den siste hindringen for å gjennomføre forbedringstiltak har vært involvering fra eksterne aktører enten form av at de endrer sin praksis eller venter på deres beslutning. Eksempler på dette er nevnt tidligere i forhold til at det kreves endringer fra kunden og ventende på beslutninger fra ledelsen når det gjelder flere deltakere i teamet. Disse eksemplene er beskrevet mer detaljert i kapittel 7.1.2. For øvrig er ikke dette en hindring som deltakerne i teamet kan gjøre noe med, da dette anses som å være utenfor teamet beslutningsområde, men de kan legge fram behovet for de som har beslutningsmyndigheten.

7.1.4 Bakgrunn for dobbeltløkkelæring

I dette delkapitlet skal vi se nærmere på hvorfor dobbeltløkkelæring har oppstått i retrospektivmøtene. Det er identifisert tre dobbeltløkkelæring i case S og én dobbeltløkkelæring i case D1 (se kapittel 6.2.2), noe som ifølge Argyis og Schönens er vanskelig å mestre. Underveis i studiet har jeg identifisert faktorer som mer eller mindre grad påvirker dannelsen av dobbeltløkkelæring i retrospektivmøtet. Enkelte faktorer er nødvendig, mens andre faktorer gjør det enklere for å danne dobbeltløkkelæring. Disse faktorene er presentert nedenfor:

- Fortrolighet
- Små gruppediskusjoner
- Fokuseret retrospektivmøte
- Fasilitator
- Konkrete forbedringstiltak
- Motivasjon

Fortrolighet er at deltakerne har gjensidig tillitt og åpenhet til hverandre i teamet. Denne faktoren ser ut til å være nødvendig for at deltakerne skal komme med sine synspunkter for å løse utfordringene teamet står overfor. Når en deltaker hevder at vedkommende har skylden for at utfordringen har oppstått i teamet, så bør samtlige deltakere i teamet fokusere på hvordan de kan løse utfordringen framfor å kritisere deltakeren som er skyldig. Et godt eksempel på dette var når deltakerne i case S diskuterte hvorfor det konstant oppstår bugs som igjen hindrer framdriften til teamet. Årsaken til dette var at en av deltakere hadde mangel på domenekunnskap. Denne situasjonen er nærmere beskrevet i kapittel 6.2.2. I stedet for å kritisere deltakeren, har teamet kommet fram til at deltakeren skal på kurs for å øke domenekunnskapen sin. Dette viser at gjensidig åpenhet og tillitt til hverandre er nødvendig for å danne dobbeltløkkelæring. Dette funnet samsvarer med Dingsøyr som illustrerer behovet for gjensidig åpenhet og tillit er nødvendig for å kunne reflektere og finne underliggende årsakene til utfordringene ved bruk av årsaksanalyse [14].

Små gruppediskusjoner er en av faktorene som kan brukes for å danne dobbeltløkkelæring. Denne praksisen er beskrevet i kapittel 6.1.4, og er brukt i case S ved fire retrospektivmøter. I de tre av fire tilfellene denne praksisen har blitt anvendt har det oppstått dobbeltløkkelæring. En mulig årsak til at det ikke ble identifisert dobbeltløkkelæring i det første møtet kan være mangel på lydopptak, se tabell 5.2. Dobbeltløkkelæring oppstår fordi deltakerne har muligheten til å fokusere på ett tema og undersøke temaet fra forskjellige synspunkter. Ved å gjøre dette er deltakerne i stand til å fokusere på å finne underliggende årsakene til utfordringen. Dette henger sammen med Zedtwit som viser til når deltakerne aktivt bidrar med sine synspunkter fører det til å forbedre læringsprosessen [60].

Dingsøyri har funnet når deltakerne aktivt bidrar med sine synspunkter hjelper dette med å finne de underliggende årsakene til utfordringen [14]. Eksempelvis har deltakerne i det syvende retrospektivmøtet kommet fram til å diskutere om hvorfor programvaren ikke ble lagt ut (released). Utfallet av de to smådiskusjonene har vært at deltakerne manglet en oversikt over hvem som hadde ansvaret på ulike ansvarsområdene når gjald arbeidsoppgaver innad i teamet. Etter gruppediskusjonen, endte deltakerne med å diskutere i felleskap om hvilke roller de trengte for å være sikre på at ulike arbeidsoppgavene blir gjennomført.

Fokusert retrospektivmøte er en faktor som kan brukes til å danne dobbeltløkkelæring. Dette innebærer at deltakerne forholder seg til en eller to bestemte utfordringer som diskuteres i teamet. På denne måten får deltakerne muligheten til å undersøke de underliggende årsakene til utfordringen. Zedtwit understreket behovet for at teamene burde fokusere på de viktigste utfordringene for å øke læringseffekten [60]. I dette studiet har alle dobbeltløkkelæringene blitt identifisert i fokuserte retrospektivmøter, se kapittel 6.2.

Fasilitatoren vil følge opp uklare og interessante utsagn og på denne måten bidrar fasilitatoren til å kunne grave dypere i utfordringen som diskuteres i teamet, se kapittel 6.1.3. På denne måten setter fasilitatoren fokuset på å finne de underliggende årsakene for utfordringen som diskuteres. Dette argumentet støttes av flere studier for bruk av fasilitator [11, 14, 55, 60]. Derfor kan det tenkes at fasilitatoren kan være en faktor for å danne dobbeltløkkelæring.

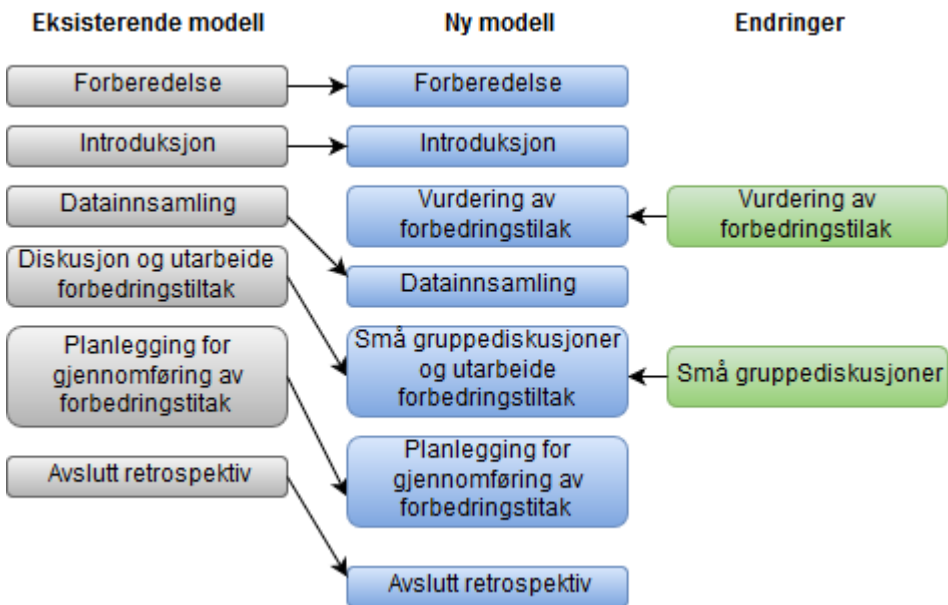
Konkrete forbedringstiltak som gjerne er målbare (muligheten til å evaluere forbedringer) er en annen faktor for å danne dobbeltløkkelæring. Det er derfor nødvendig at forbedringstiltak blir gjennomført for at teamet skal oppnå dobbeltløkkelæring. I de fire tilfellene der dobbeltløkkelæring har blitt identifisert, har forbedringstiltakene vært konkrete. Et eksempel på dette er presentert i kapittel 6.1.4 og diskutert i kapittel 7.1.2, der en deltaker beskriver at vedkommende ikke har domenekunnskap og derfor oppstår det bugs på bakgrunn av det. En lignende situasjon er også beskrevet i kapittel 7.1.3, men utfallet er forskjellig, da dette er to forskjellige møter. Mangel på domenekunnskap førte til at deltakeren ble sendt på ett kurs for å øke domenekunnskapen sin slik at antall bugs i programvaren ble redusert. Deltakerne bestemte seg for å måle antall bugs fra antall oppgaver i Jira som omfatter bugs. Selv om antall jiraoppgaver brukes som en måleenhet, kan det tenkes at det eksisterer flere bugs i programvaren sammenlignet med rapporterte bugs i Jira. Likevel gir antall oppgaver i Jira som omfatter bugs en indikasjonen om domenekurset har redusert antall bugs i programvaren.

Motivasjon er en annen faktor som er nødvendig for å danne dobbeltløkkelæring i retrospektivmøtet. Når deltakerne har motivasjon fører det til at deltakerne aktivt bidrar med sine synspunkter som kan være nyttig for å finne de underliggende årsakene til utfordringen teamet diskuterer. Zedtwit identifiserte at motivasjon er nødvendig for læringsprosessen i et retrospektivmøte [60]. I de retrospektivmøtene

det ikke ble identifisert dobbeltløkkelæring i case S har deltakerne hatt åpen diskusjon om flere temaer og uten komme til en konklusjon, som er beskrevet i kapittel 6.1.4. En konsekvens av dette har i senere tid ført til manglende motivasjon for å gjennomføre retrospektivmøter, da de samme utfordringene diskuteres gjentatte ganger, se kapittel 7.1.2. Drury et al. understreker når deltakerne ikke får utbytte av retrospektivmøter, kan det føre til manglende motivasjon for å gjennomføre retrospektivmøte [62]. Dette indikerer at behovet for motivasjon er nødvendig for å danne dobbeltløkkelæring i retrospektivmøter.

7.1.5 Min modell for gjennomføring av retrospektivmøter

Jeg har utarbeidet ett forslag til gjennomføring av retrospektivmøter basert på modellen som er presentert på kapittel 3.4 og revidert i forhold til mine funn i dette studiet. Revideringene fra eksisterende modell til ny modell, vises på figur 7.1. Endringene er nye trinn som enten er lagt som eget trinn eller endret på allerede eksisterende trinn. Den viktigste revideringen er små gruppediskusjoner som har vært en viktig faktor for å finne de underliggende årsakene til utfordringene som teamet står overfor. Dette er et nytt trinn som er slått sammen med det tidligere trinnet 'diskusjon og utarbeide forbedringstiltak'. I tillegg har jeg lagt til ett nytt trinn 'vurdering av forbedringstiltak'.



Figur 7.1: Revideringene fra eksisterende modell til ny modell

Det første trinnet er *forberedelse* for retrospektivmøtet, som foregår før selve møtet. Her avgjøres det hvem som bør inviteres til møtet, bruk av hjemmelekse og valg av fasilitator. Alle som har bidratt til prosjektet bør være invitert, spesielt nyansatte og deltakere som har vært fraværende i løpet av sprinten, selv om dette ikke er praksisen i dette studiet se kapittel 6.1. Hjemmelekse lønner seg i to tilfeller, der den ene er at deltakerne vil være fraværende i retrospektivmøtet og den andre er at deltakerne har lider av hukommelsessvikt, se kapittel 6.1 og 7.1.2. I begge tilfellene bør det foretas en datainnsamling i forkant av møtet, slik at bidraget til deltakerne styrkes. En måte å gjennomføre dette på er å bruke verktøy, der deltakerne skriver virtuelle lapper i forkant av møtet, se kapittel 6.1.3. Dingsøyr og Birk et al. argumenterer for bruk av ekstern fasilitator, da deltakerne må forklare mer gjennomtenkt [11, 14]. Intern fasilitator kan brukes dersom vedkommende klarer å følge opp interessante og uklare synspunkter.

Det andre trinnet er *introduksjon* for retrospektivmøtet, og det første trinnet i selve møtet. Hensikten med trinnet er å skape en god atmosfære for å oppmuntre deltakerne til delta aktivt i møtet og samtidig skal fasilitatoren presentere agendaen for møtet og tidsbruken på trinnene. Det er nødvendig at deltakerne er bevisste på tidsbruken for å sørge for at teamene klarer å gjennomføre agendaen for møtet. Når tidsbruken er ute av kontroll kan det føre til at deltakerne ikke utarbeider forbedringstiltak slik det skjedde i case D1, se kapittel 6.1.5. Med andre ord finner ikke teamet løsningen på utfordringene de står overfor.

Det tredje trinnet er *vurdering av forbedringstiltak* i denne modellen som brukes for å vurdere i hvilken grad forbedringstiltakene gjennomføres i teamet. Resultatet av aktiviteten vil dette være til nytte for teamet uavhengig av utfallet av aktiviteten. Når resultatet viser at deltakerne i stor grad gjennomfører forbedringstiltak kan dette føre til mer motivasjon og interesse for gjennomføring av retrospektivmøtet som også understrekes av Dingsøyr et al. [82] og Derby og Larsen [7]. I tilfeller der aktiviteten viser at deltakerne i liten grad gjennomfører forbedringstiltakene, kan teamet finne en løsning for å øke gjennomføringsevnen. På denne måten kan en i tidlig fase hindre at deltakerne mister motivasjonen for å gjennomføre retrospektivmøtene og forbedringstiltakene. Det er tidligere beskrevet at motivasjonen for å gjennomføre retrospektivmøtene har blitt redusert da forbedringstiltakene ikke gjennomføres og en konsekvens av dette har vært at deltakerne diskuterer de samme temaene gjentatte ganger, se kapittel 6.1.5.

Det fjerde trinnet er *datainnsamling* som hovedsakelig gjennomføres slik den er beskrevet i kapittel 6.1.3. Dette innebærer at deltakerne skriver lapper om hva som gikk bra og hva som kunne ha gått bedre i løpet av sprinten. Det understrekes at deltakerne må presentere hver enkelt lapp for resten av teamet og dermed hindre eventuelle misforståelser. På denne måten vil deltakerne ha bedre utgangspunkt for å diskutere utfordringene senere. Etter at lappene er skrevet, grupperes lappene hver for seg i de to kategoriene av to frivillige deltakere i teamet.

Det femte trinnet er *små gruppediskusjoner og utarbeidelse av forbedringstiltak* innebærer votering, små gruppediskusjoner og utarbeidelse av forbedringstiltak. Det begynner med at deltakerne stemmer over hvilke tema (grupperte lapper) de vil fokusere nærmere når det gjelder positive og forbedringslapper. Birk et al. påpeker at prioritering av temaene er nødvendig for å kunne fokusere på de viktigste utfordringene teamet står ovenfor, og kun utarbeide forbedringstiltak på disse [11]. Derfor bør fasilitator gjøre rede for at deltakerne skal diskutere temaet som får flest stemmer i hver kategori i små gruppediskusjoner. Det er flere teknikker for hvordan voteringen kan foregå, og ett alternativt er beskrevet i kapittel 6.1.4. Deretter blir deltakerne delt i to smågrupper. Den ene gruppen diskuterer temaet for positive lapper, mens den andre gruppen diskuterer temaet for forbedringslappene. Små gruppediskusjoner lønner seg når det er minst tre deltakere på hver gruppediskusjoner eller maks fem deltakere. Ved store gruppediskusjoner kan det være vanskeligheter med at samtlige deltakere kan bidra i diskusjonen i like stor grad og dermed kan deltakere sitte igjen med informasjon som ikke blir formidlet og som er en hindring for læring. Når deltakerne i større grad har muligheten til å bidra til diskusjonen så økes muligheten for å finne underliggende årsakene til utfordringen, se kapittel 7.1.4. Utfallet fra gruppediskusjonene er årsaker og sammenhenger for at behovet eller utfordringene finner sted tilknyttet utdelt tema, samtidig som det utarbeides forbedringstiltak for disse. Resultatet av smågruppediskusjoner presenteres for resten av teamet. Basert på dette utarbeides en liste over forbedringstiltak.

Det sjette trinnet er *planlegging for gjennomføring av forbedringstiltak* som innebærer at teamet planlegger gjennomføringen av forbedringstiltakene. I dette trinnet velger deltakerne ut de viktigste forbedringstiltakene fra listen. De valgte forbedringstiltakene må være konkrete og målbare, se kapittel 7.1.4. Målbare forbedringstiltak er nødvendig for å ha mulighet til å vurdere når et forbedringstiltak er gjennomført, slik at deltakerne i teamet kommer fram til entydig konklusjon når trinnet ”vurdering av forbedringstiltak” gjennomføres ved neste retrospektivmøtet. Forbedringstiltaket tildeles en spesifikk deltaker som har mer ansvar for oppfølging og gjennomføring av forbedringstiltaket i forhold til andre deltakere, men dette betyr ikke at det deltakeren selv må gjennomføre forbedringstiltaket. Dette er fordi det skal hindre situasjonene som er gjengitt i kapittel 7.1.3 og påpekes av Drury et al. [62] hvor en deltaker tror en annen deltaker skal gjennomføre forbedringstiltaket og ender med at ingen gjennomfører forbedringstiltaket tilslutt.

Det syvende og siste trinnet er *avslutt retrospektiv* hvor deltakerne skal reflektere over hvordan retrospektivmøtet ble gjennomført og dokumentere utfallet av møtet. Refleksjonen skal gjenspeile nytteverdien med retrospektivmøtet og skal komme med forslag til forbedring når det gjelder praksisen. Aktiviteten plus/delta som er beskrevet i kapittel 6.1.6 kan benyttes til gjennomføre dette trinnet, da aktiviteten ga deltakerne muligheten til å reflektere over både positive og negative sider ved retrospektivmøtet. Refleksjon på retrospektivpraksisen har tidligere gitt positive resultater da deltakerne i case S opplevde at de samme temaene ble diskutert gjentatte ganger, se kapittel 7.1.3, selv om denne praksisen ikke ble gjennomført som egen aktivitet. Det fysiske utbyttet skal være dokumentasjon på de viktigste utfordringene og behovene, der det utarbeides forbedringstiltak og hvordan disse følges opp og evalueres.

7.2 Kommunikasjon

I dette kapitlet diskuteres hvilke kommunikasjonsutfordringer distribuerte team har i forhold til samlokaliserte team. Resultatet av denne diskusjonen er en liste over hvilke konsekvenser dette gir for samlokalisert og distribuerte team. Kommunikasjon er et stort og vidt begrep, og derfor ser vi kun på disse faktorene som innholdsrik kommunikasjon, uformell kommunikasjon og fysisk nærhet. For å diskutere disse faktorene tas inn andre elementer i kommunikasjonen som er presentert i kapittel 4.1. I første delkapitlet diskuteres innholdsrik kommunikasjon, mens i det andre delkapitlet er uformell kommunikasjon og fysisk nærhet diskutert samlet. Dette er fordi disse henger sammen, og det er vanskelig å diskutere uformell kommunikasjon uten dra inn elementer fra fysisk nærhet.

7.2.1 Innholdsrik kommunikasjon

I distribuerte team er kommunikasjon spesielt viktig når deltakerne ikke har muligheten til å sitte fysisk sammen og har behov for å kommunisere med hverandre for å avklare misforståelser og formidle informasjon. Denne utfordringen er identifisert i case D1, der deltakerne følte at de arbeidet mindre effektivt når de måtte koordinere arbeidet på tvers av lokasjoner, se kapittel 6.1.4. Dette har en sammenheng med Herbsleb og Mockus som har funnet når informasjonen ikke blir utvekslet som ansikt-til-ansikt kommunikasjon så vil dette føre til at forståelsen av arbeidsoppgavene reduseres og av den grunn kan dette forsinke arbeidet som er utført i distribuerte team. En konsekvens av dette er at distribuerte team bruker 2,5 ganger mer tid i forhold til samlokaliserte team for å gjennomføre de samme arbeidsoppgavene [13]. Konsekvensene av avhengigheter mellom distribuerte team er selv beskrevet av SM_{D1}: *"Vi er bakpå, enda en sprint med lav måloppnåelse. Avhengigheter på tvers av lokasjoner gjør det ekstra tungt"*. Dette skyldes som Dorairaj forklarer at distribuerte deltakere tilegner seg informasjon gjennom videokonferanse, telefonsamtaler og eposter som er mindre innholdsrik sammenlignet med ansikt-til-ansikt kommunikasjon [70]. Dette er en motsetning i forhold til Gutwin et al. der de argumenterer ved bruk av opensource kommunikasjon at ansikt-til-ansikt kommunikasjon er ikke den eneste løsningen for å skape bevissthet og forståelse rundt arbeidsoppgavene som gjennomføres i distribuerte team [52]. Jeg finner dette overraskende, da dette er en direkte kontrast i forhold til mine funn, og årsaken til dette diskutere i neste delkapittel.

7.2.2 Uformell kommunikasjon og fysisk nærhet

Det ble i forrige delkapittel gjort rede for at distribuerte deltakere har ikke muligheten til å gjennomføre ansikt-til-ansikt kommunikasjon på bakgrunn av geografisk avstand.

Uformell kommunikasjon ser ut til å være en avgjørende faktor når gjelder kommunikasjonsutfordringer i forhold til distribuert og samlokalisert team. I følge Gutwin et al. er det vanlig at det oppstår avhengigheter mellom arbeidsoppgaver når flere deltakere arbeider sammen [52]. I case S er det ikke vist til bekymringer i forhold til gjennomføringen av arbeidsoppgaver hvor det har oppstått avhengigheter. Dette er en motsetning til case D1 der deltakerne har hatt flere utfordringer forbinde med avhengigheter på tvers av distribuerte team. Årsaken til denne forskjellen er beskrevet av U_S: *"Når du lurer på om du skal gjennomføre jira-oppgaven så går du til scrum-master og spør om det [..]"*. Dette viser at uformell kommunikasjon er en kommunikasjonsform som brukes for å hindre misforståelser og avklaringer i samlokalisert team, og derfor har ikke samlokalisert team bekymringer i forhold til avhengigheter mellom arbeidsoppgaver. Espinosa et al. beskriver nettopp at betydelige mengder koordinering foregår som uformell kommunikasjon i samlokaliserte team [67]. O'Leary og Cummings undersøkte kommunikasjonen til flere distribuerte team og fant at kommunikasjonsfrekvensen var avtagende relatert til spredningen av teamet [2]. Derfor kan det tenkes at distribuerte team gjennomfører ikke uformell kommunikasjon i like stor grad, og har dermed ikke muligheten til å avklare sin arbeidssituasjon like raskt som samlokaliserte team.

Det er tidligere nevnt at deltakerne i case D1 har hatt utfordringer med å oppdatere arbeidsverktøyet Jira, se kapittel 6.1.4. Årsaken til denne utfordringen henger sammen med at deltakerne i distribuert team har ikke muligheten til å kunne gjennomføre uformell kommunikasjon i like stor grad som samlokalisert team. En konsekvensene har vært at deltakerne ikke arbeider som planlagt slik PL_{D1} forklarer det:

"Det kan være kommunikasjonssvikt og jeg opplever egentlig at vi har forlatt planen, men jobber ikke som planlagt og oppdaterer ikke planen [sprintkøen]".

I dette tilfellet har U_{D1} ett annet perspektiv:

"Det er ett eller annet aspektet her, kanskje ikke det jeg jobbet på også må jeg dra det inn sprinten og sprinten blir en uke lenger. Oi hvilken rakker [oppgave] er det som skyves her, og det er ikke jeg helt riktig person til å ta stilling til og jeg orker ikke ta det i sprinten [...] Da må jeg spørre hele teamet jobber du på den her? Er det riktig av meg å ta den ut?".

Utsagnet ovenfor viser at utvikleren har ett annet synspunktet på utfordringen. Dette innebærer at utviklerne innser at oppgaven som de arbeider med, krever at en annen oppgave blir gjennomført samtidig, og denne oppgaven er ikke en del av

sprinten. Ved at de tar inn oppgaven blir det mindre sannsynlig for at sprintmålet blir nådd, og derfor må utvikleren skyve en oppgave til neste sprint. For å kunne gjøre dette må utvikleren spørre resten av teamet for å finne ut av hvilken oppgave som skal skyves til neste sprint. Siden utvikleren har ingen oversikt over hva de andre deltakerne holder på, så bør vedkommende kommunisere med samtlige deltakere på tvers av distribuerte team. Herbslebs og Grinter viser til at distribuerte deltakere i liten grad har muligheten til å gjennomføre uformell kommunikasjon, i motsetning til samlokaliserte team der deltakerne raskt kan invitere andre deltakere til kommunikasjon, da de vet hvem som er tilgjengelige [72]. Siden det er vanskelig å få kommunisert med samtlige deltakere på tvers av et distribuert team, så velger deltakeren å ikke skyve oppgaven til neste sprint, og arbeide med oppgaven som deltakeren ønsket å ha i sprinten uten å oppdatere arbeidsverktøyet Jira. Det at deltakeren har manglende oversikt over hva de andre deltakerne holder på med, viser mangel på gruppebevissthet, som ifølge Gutwin et al. var nøkkelen for å løse kommunikasjonsutfordringene i distribuert opensource prosjektene [52]. Dette viser at deltakerne må vite hva andre deltakere holder på med for at samarbeidet skal fungere. I kapittel 6.1.3 ble det beskrevet at deltakerne i case D1 vil ha kontinuerlig videooverføring av arbeidsrommene, da de opplever at det er ett skille mellom by1 og by2. På denne måten kan deltakerne i den ene byen med få med seg informasjonen angående tilgjengeligheten til deltakerne i den andre byen samtidig se visuelt hva de ulike deltakerne holder på med til enhver tid i arbeidsrommene. Informasjonen som deltakerne får med seg er mindre innholdsrik sammenlignet med ansikt-til-ansikt kommunikasjon fordi de ikke får med seg informasjonen som foregår utenfor kamerasynsvinkelen.

Informasjonen om hva andre deltakere på holder med har vist seg å være nødvendig for at samarbeidet skal fungere. I case D1 har deltakerne diskutert ett forslag for å hindre at det oppstår høy kohesjon mellom lokasjonene, på bakgrunn av arbeidsoppgavene deles på tvers av lokasjonene, U_{D1} : *"Når oppgavene er større grad eid og isolert [..]. Jeg vil tro det vil være mindre avhengigheter hvis vi hadde delt horisontalt med at det var bare ett felles API imellom oss"*. En kan argumentere for at mindre avhengigheter mellom lokasjonene kan føre til mer produktivitet i de forskjellige lokasjonene, men dette stemmer ikke. Herbles understøtter ved å tildele en programvaremodul til en lokasjon og ikke på tvers av lokasjoner, så fører det ikke til færre avhengigheter mellom lokasjonene, da man i større grad trenger å kommunisere på tvers av lokasjonene for å løse arbeidsoppgavene godt nok [78]. Dette er en indikasjon på at avhengighetene ikke løses ved å tildele programvaremodul til hverandre uten av det oppstår avhengigheter.

Det er tidligere nevnt at distribuert deltakere har ikke stor mulighet til å gjennomføre uformell kommunikasjon sammenlignet med samlokalisert team. Melo et al. har funnet at samlokaliserte team har en tendens til å forbedre kommunikasjonen og samarbeidet blant deltakerne i teamet [9]. I kapittel 6.1.4 ble det beskrevet at kommunikasjonsutfordringene hindret framdriften til deltakerne, og der valgte de å reise til hverandre for å arbeide tettere sammen ved behov. Dette samsvarer

med Herblebs og Grinter som har funnet at samlokaliserte deltakere kan raskere invitere hverandre til kommunikasjon ved å se om de er tilgjengelige [72]. Derfor kan det tenkes når distribuerte deltakerne er samlet på ett fysisk sted, så vil kommunikasjon mellom deltakerne være ansikt-til-ansikt kommunikasjon. Denne type kommunikasjon gir bedre forståelse som vil øke effektiviteten rundt arbeidet som skal gjennomføres. French og Layzell understreker konsekvensene ved at distribuerte deltakere reiser til hverandre, der den største konsekvensen er store kommunikasjonsutgifter avhengig av den geografiske avstanden, og ved daglig reiser bør den geografiske avstanden være kort [71].

I både samlokalisert og distribuerte team er det viktig at deltakerne som arbeider sammen sitter ved siden av hverandre for å redusere utfordringene knyttet til kommunikasjon. I dette studiet har kun case D1 identifisert denne utfordringen, da deltakerne i case S sitter allerede samlet. U_{D1} har beskrevet denne situasjonen i retrospektivmøtet: *”Vi bør sitte ved siden av hverandre for å jobbe mer effektivt. Nytt medlem i teamet sitter litt lengre unna. [...]”*. Når deltakerne ikke sitter ved siden av hverandre så kan dette føre til at effektiviteten av gjennomføringen av arbeidsoppgavene reduseres. Dette betyr at en fysisk plassering av individene i et team i et kontorlandskap er en vesentlig del av kommunikasjonsutfordringene i samlokalisert og distribuert team. Misha et al. har funnet at individene bør bli plassert ved siden av hverandre for å effektivisere samarbeidet og øke fremdriften i prosjektet. Ulempene med at deltakerne i teamet blir plassert på feil sted, kan føre til lavere produktivitet og dårligere kvalitet på produktet [18]. Det er tidligere gitt ett eksempel der deltakerne i case S spør scrum-master for avklaringer, noe som viser at fysisk nærhet kan føre til gruppebevissthet oppstår ved at de kan gjennomføre uformell kommunikasjon ved behov.

Historikk og fortolkninger

I både case S og case D1 er forståelsen av informasjonen som blir formidlet viktig for at deltakerne i teamet skal få nytte av kommunikasjonen. I case D1 har deltakerne nevnt at manglende felles referanseramme og felles vokabular og felles begreper har ført til at de får forskjellig svar avhengig av hvem de spør, se kapittel 6.1.3. Yu og Petter argumenterer at uten fellesegenskaper blant teamet i form av forståelse for oppgaver og relasjoner innad i teamet vil det være umulig å oppnå teamets mål [83]. Basert på denne informasjonen kan en person påpeke at fellesforståelse er nødvendig for å kunne samarbeide med hverandre, noe som ikke stemmer. Star og Griesemer understreker at enhver person har ulik historikk som fører til at ulike personer tolker informasjonen i lys av denne historikken og likevel klarer deltakerne og samarbeide med hverandre [73].

Produktkøen er en artefakt som flere deltakere forholder seg til for å se hvilke arbeidsoppgaver som er prioritert. Produktkøen kan ses som ett eksempel der det foregår overføring av informasjonen mellom deltakerne. Figueiredo et al. argumenterer for at forskjellige interessenter trenger å bruke de samme terminologiene for

å være sikre på at de forstår hverandre og for å kunne samarbeide med hverandre [75]. I case D1 er ikke dette tilfelle, da det brukes forskjellige terminologier for å beskrive de samme arbeidsoppgavene i produktkøen og som har ført til at det oppstår duplikater av arbeidsoppgaver. Melo et al. har funnet at det er nødvendig at deltakerne har samme forståelse når det gjelder utføring av arbeidsoppgaver, hvis ikke kan det påvirke produktiviteten [9]. Det er vanskeligere for deltakerne i distribuerte team å ha samme forståelse til arbeidsoppgaven i forhold til samlokaliserte team. I Case D2 har ikke deltakerne vært klare over hvilke instruksjoner de skal følge når en arbeidsprosess skal gjennomføres og konsekvensene ved at disse ikke følges er formidlet av U_{D2} : *"Det var installasjon av [funksjonalitet] som var utdatert og jeg må videregående til e-mails med nye informasjon. Det var noe som ikke fungerte der, så enda en ny enhet. [...] Jeg fikk tips om pdf, om dem var oppdatert eller ikke – vet ikke jeg"*. Utsagnet understreker at dårlig kommunikasjon og ansvarsfraskrivelse på tvers av lokasjoner kan føre til merarbeid for deltakere som brukte det konkrete dokumentet til å gjennomføre arbeidsoppgavene sine. Denne utfordringen har en sammenheng med Herblebs og Mockus som beskriver at utfordringer knyttet til koordinering krever et stort antall deltakere involveres når distribuerte endringer forekommer og på sikt vil det føre til forsinkelser. I tillegg beskriver de at forståelsen rundt arbeidet kan skapes gjennom kommunikasjon, og denne måten kan man hindre at oppgaven blir halvferdig og dermed slippe å bruke ekstra ressurser for å fullføre arbeidsoppgaven [13]. Det er nevnt tidligere i dette delkapitlet at samlokaliserte team har i større grad mulighet til å kommunisere med hverandre i forhold til distribuerte team. Derfor kan det tenkes at merarbeid ser ut til å være en større utfordring for distribuerte team i forhold til samlokaliserte team.

Det at deltakerne bruker samme terminologi for å kunne forstå hverandre og samarbeide er en faktor. Charlie nevner at siden deltakerne har ulik historikk vil det oppstå begrensninger i forhold til å ha felles syntaks og derfor understrekes viktigheten av å anerkjenne at fortolkende forskjeller ofte eksisterer [74]. Eksempelvis når U_S har hatt ett møte med en kunde og skal legge til en arbeidsoppgave i Jira: *"Han har ikke sett andre oppgaver som ligger der [...] selvfølgelig er oppgaven du kommer den viktigste siden du har snakket med kunden, og kunden er skikkelig sur"*. Utsagnet viser at det er kun utvikleren som har fått informasjonen fra kunden, og tolker denne arbeidsoppgaven som høyest prioritert i forhold til hva andre deltakere har gjort da de ikke har den samme historikken. Når flere slike tilfeller inntreffer kan dette føre til slik U_S understreker: *"Det hjelper ikke å legge viktige saker øverst fordi alle saker blir kategorisert som viktigst"*. Drury et al. gjennomførte en fokusgruppe og fant at prioritering av arbeidsoppgaver var en utfordring for systemutviklingsteam [62]. Derimot ser ikke prioritering av arbeidsoppgaver til å være en utfordring i case D1 og en mulig årsak til denne utfordringen er at deltakerne ikke følger sprintkøen som nevnt tidligere. Moe et al. har funnet når deltakerne kun arbeider med sine programvaremoduler, så er de lite interesserte for å høre hva de andre deltakerne holder på med og deres status på arbeidsoppgavene [77]. Derfor kan det tenkes at det var likegyldig om sprintkøen var prioritert korrekt eller ikke for deltakerne i case D1.

7.2.3 Oppsummering

Det er flere konsekvenser på bakgrunn av at det oppstår kommunikasjonsutfordringer i samlokaliserte og distribuerte team. Disse konsekvensene er beskrevet i tabell 7.4

Konsekvens	Samlokalisert (case S)	Distribuert (case D1 og D2)
Uformell kommunikasjon	Uformell kommunikasjon hindrer avhengigheter mellom deltakere og arbeidsoppgaver.	Manglende uformell kommunikasjon kan føre til at teamet ikke følger planen. Eksempelvis at sprintkøen blir ikke oppdatert
Fysisk nærhet	Deltakerne sitter sammen og gir uttrykk for at dette øker effektiviteten.	Viser til tegn til at det er ineffektivt å arbeide når de ikke sitter sammen ved store avhengigheter.

Tabell 7.4: En oversikt over hvilke kommunikasjonsutfordringer distribuerte team har i forhold til samlokalisert team.

Uformell kommunikasjonen viser tydelig skillet mellom samlokalisert og distribuert team. I samlokalisert team har uformell kommunikasjon hjulet med å hindre avhengigheter mellom deltakere og arbeidsoppgaver. Dette er til forskjell fra distribuerte team, der de ikke har muligheten til å gjennomføre uformell kommunikasjon i like stor grad på bakgrunn av fysisk avstand. Derfor har de vanskeligheter med å se hvilke deltakere som er tilgjengelig til enhver tid.

Både samlokalisert og distribuerte team viser til at fysisk nærhet gir fordeler i forhold til å kunne arbeide effektivt sammen, da de har tilgang til ansikt-til-ansikt kommunikasjon. I distribuerte team har deltakerne beskrevet at det er vanskeligheter med å kunne arbeide effektivt sammen når det oppstår avhengigheter på tvers av lokasjonene.

Dette viser som nevnt innledningsvis i dette kapitlet at ansikt-til-ansikt kommunikasjon er nødvendig for å skape bevissthet og forståelse rundt arbeidsoppgavene som gjennomføres. Dette har henger sammen med hovedflyten av forskningsstudier på fagfelt, men er en imotsetning i forhold til andre studier. Deriblant Gutwin et al. som understreker at ansikt-til-ansikt kommunikasjon er ikke den eneste måten for å skape bevissthet [52]. Derimot er det fellesnighet om at bevissthet er nødvendig for å ha fornuftig kommunikasjon.

Kapittel 8

Konklusjon og videre arbeid

I dette kapitlet beskrives konklusjonen for studiet hvor de viktigste funnene gjengis og tilslutt gis det forslag til videre arbeid.

8.1 Konklusjon

Det er gjennomført casestudier for å undersøke hvilke utfordringer systemutviklingsteam har i forhold til læring i retrospektivpraksisen og kommunikasjonsutfordringer i distribuerte team. Konklusjonen på disse utfordringene er beskrevet nedenfor.

8.1.1 Organisatorisk læring

Innledningsvis ble det nevnt at manglende fokus på læring var hovedårsaken til at prosjektene ble forsinket eller mislyktes. I dette studiet har vi sett at deltakerne lærer fra retrospektivmøtene, men læringen er begrenset. En av årsakene er at deltakerne ikke gjennomfører forbedringstiltakene og får dermed ikke forbedret eksisterende praksis. Det kan med andre ord beskrives som mangel på bruk av læringen i retrospektivmøtet.

Det ble anvendt Argyis og Schönens læringsmodeller for å undersøke læringseffekten i retrospektivmøter. Dobbeltløkkelæringer forekommer i retrospektivmøter, men det er primært enkeltløkkelæringer som er identifisert i retrospektivmøter. Den største hindringen for å danne dobbeltløkkelæring har vært gjennomføring av forbedringstiltakene. Dette har ført til at de samme utfordringene diskuteres gjentatte ganger og som følge av dette har motivasjonen for å gjennomføre retrospektivmøter blitt redusert. En annen hindringen har vært eksterne aktører hvor deltakerne har utarbeidet forbedringstiltak for aktører som ikke er en del av teamet, eksempelvis ledelsen eller kunden.

I de tilfellene dobbeltløkkelæring er identifisert har deltakerne diskutert utfordringen i små gruppediskusjoner, og deretter presentert resultatet for resten av teamet. Dette har vist seg å være en betydelig effekt for å danne dobbeltløkkelæring, da hver deltakerne har større rom for å ytre sine synpunkter i diskusjonen. Dette hjelper med å finne de underliggende årsakene til utfordringen som teamet står overfor. Adferden og holdningen til deltakerne har vært avgjørende for å danne dobbeltløkkelæring, der åpenhet og tillit blant deltakerne i teamet har vært en viktig faktor.

Det kan konkluderes med at dagens utfordring for systemutvikling i forhold til læring i retrospektivmøtet er interne forpliktelser og konstant evaluering av handling, nærmere bestemt øke gjennomføringsevnen av forbedringstiltak. For å øke gjennomføringsevnen, må forbedringstiltakene være målbar slik at teamet kan måle effekten av forbedringstiltaket og avgjøre hvordan forbedringstiltakene skal gjennomføres. Derfor er mitt bidrag for å løse denne utfordringen, en modell som er presentert for praktikere:

- Det er utarbeidet en modell for gjennomføring av retrospektivmøtet for å øke gjennomføringsevnen av forbedringstiltak og som er tilrettelagt for å danne dobbeltløkkelæringer.

8.1.2 Kommunikasjon

Når det gjelder kommunikasjon har det vist seg at ansikt-til-ansikt kommunikasjon er nødvendig for å kunne lettere formidle informasjonen og forstå hverandre. Denne muligheten har ikke distribuerte team, som har ført til misforståelser og forsinkelser av arbeidet. Disse funnene henger sammen med eksisterende forskning som forteller at ansikt-til-ansikt kommunikasjon er nødvendig for å skape bevissthet og forståelse rundt arbeidsoppgavene som skal gjennomføres. Dette er derimot en motsetning i forhold til Gutwin som har vist til at ansikt-til-ansikt kommunikasjon er ikke nødvendig for å skape bevissthet og forståelse.

Kommunikasjonsutfordringene i distribuerte team har i følge deltakerne i studiet ført til forsinkelser. Da det er enklere forklare og arbeide sammen når de sitter fysisk sammen i forhold til å kommunisere over videokonferanse. Dette er en motsetning til samlokaliserte team der dette ikke har vært en utfordring.

8.2 Forslag til videre arbeid

Det er to forslag til videre arbeid, der det ene er å bruke modellen som jeg har utarbeidet til å teste dette i andre casestudier i systemutviklingsprosjekter og det er andre forslaget er å undersøke hvordan bruk av verktøy kan påvirke læringen i retrospektivmøtet. Disse to forslagene beskrives nedenfor.

Dette studiet er hovedsakelig basert på ett casestudie, der hovedtyngden av datamaterialet foreligger. Derfor anbefales det å gjennomføre lignende studier hvor det brukes modellen som jeg har utarbeidet for gjennomføring av retrospektivmøtet og som forhåpentligvis kan bekrefte funnene i dette studiet. Et slikt studie må være empirisk for å bekrefte funnene, og i tillegg bør det gjennomføres i andre land eller kulturer i enn Norge for å se små gruppediskusjoner har effekt i forhold til å danne dobbeltløkkelæring.

Tidligere i dette studiet var ønsket å kunne sammenligne samlokalisert læringsutbytte med distribuert læringsutbytte, noe som ikke har skjedd. I forhold til dette formålet har jeg utarbeidet en liste med verktøy for å undersøke fordeler og ulemper ved gjennomføring av retrospektivmøtet i et distribuert team, der verktøyet Retrium tilfredsstilte flere av kravene. Det er samtidig gjennomført ett retrospektivmøtet, der dette verktøyet er brukt og gjennomføringen av denne er beskrevet i vedlegg A. Et interessant problemsstilling kan være i hvilken grad læringen blir påvirket ved bruk av verktøy. For å gjennomføre denne forskningen, bør det først gjennomføres ett distribuert retrospektivmøte, der deltakerne gjennomføre retrospektivmøte uten verktøy ved bruk av videokonferanse. Deretter anbefales det å bruke verktøyet Retrium til å gjennomføre retrospektivmøtet. For å måle effekten av læringen i retrospektivmøtet kan antall enkeltløkkelæring og dobbeltløkkelæring være ett forslag, selv om dette ikke gir det helhetlig bildet av læring som foregår, så det gir det en indikasjon på læringen som foregår med og uten verktøy.

Bibliografi

- [1] C.Argyris og D.Schön. *Organizational Learning II: Theory, Method and Practice*. Addison-Wesley Publishing Company, USA, 1996.
- [2] M.B. O’Leary og J.N Cummings. The spatial, temporal and configurational characteristics of geographic dispersion in teams. *MIS Quarterly*, 31(3):433–452, 2007.
- [3] R.E Kraut, R.S.Fish, R.W Root og B.L Chalfonte. Informal communication in organizations: Form, function and technology. In S. Oskamp og S.Spacapan, editor, *Human Reactions to Technology: The Claremont Symposium on Applied Social Psychology*, pages 145–199. USA: Sage Publications, 1990.
- [4] Winston W. Royce. Managing the development of large software systems. *In Proceedings of IEEE WESCON*, 1970.
- [5] Barry W. Boehm. A spiral model of software development and enhancement. *Computer*, 21(5):61–72, 1988.
- [6] P.Abramsson, N.Oza og M.T Siponen. Agile software development methods: A comparative review. In T.Dingsøy, T.Dybå og N.B.Moe, editor, *Agile Software Development*, pages 31–59. Berlin: Springer Verlag Berlin Heidelberg, 2010.
- [7] E.Derby og D.Larsen. *Agile Retrospectives: Making Good Team Great!* Pragmatic Bookshelf, USA, 2006.
- [8] M. Korkala, P.Abrahamsson og P.Kyllönen. A case study on the impact of customer communication on defects in agile software development. In *Agile Conference 2006*, pages 76–87. USA: IEEE Computer Society, 2006.
- [9] F.Kon og R.Conradi C.O Melo, D.S.Cruzesm. Interpretative case studies on agile team productivity and management. *Information and Software Technology*, 55(2):412–427, 2013.
- [10] Robert L. Glass. Project retrospectives, and why they never happen. *Software IEEE*, 19(5):111–112, 2002.
- [11] A.Birk , T.Dingsøy og T.Stålhane. Postmortem never leave a project without it. *Software IEEE*, 19(3):43–45, 2002.

- [12] M.Korkala, M.Pikkarainen og K.Conboy. A case study of customer communication in globally distributed software product development. In *Proceedings of the 11th International Conference on Product Focused Software*, Profes '10, pages 43–46. USA: ACM, 2010.
- [13] J.D. Herbsleb og A.Mockus. An empirical study of speed and communication in globally distributed software development. *IEEE Transactions on Software Engineering*, 29(6):481–494, 2003.
- [14] Torgeir Dingsøy. Postmortem reviews: Purpose and approaches in software engineering. *Information and Software Technology*, 47(5):293–303, 2004.
- [15] Patrick Kua. *The retrospective handbook - A guide for agile teams*. Leanpub book, USA, 2012.
- [16] *10th Annual State of Agile Survey*, VersionOne Inc, 2016, Besøkt 31.05.2016. <https://www.versionone.com/pdf/state-of-agile-development-survey-ninth.pdf>.
- [17] *3th Annual State of Agile Survey*, VersionOne Inc, 2009, Besøkt 28.09.2015. <https://www.versionone.com/pdf/state-of-agile-development-survey-ninth.pdf>.
- [18] D. Mishra, A.Mishra og S.Ostrowska. Impact of physical ambiance on communication, collaboration and coordination in agile software development: An empirical evaluation. *Information and Software Technology*, 54(10):1067–1078, 2012.
- [19] Outi Salo. Professional knowledge management. In K.D.Althoff , A.Dengel , R.Bergmann , M.Nick og T.R.Berghofer, editor, *Systematical Validation of Learning in Agile Software Development Environment*, pages 106–110. Berlin: Springer Verlag Berlin Heidelberg, 2005.
- [20] T. E. Bell og T.A Thayer. Software requirements: Are they really a problem? In *Proceedings of the 2nd International Conference on Software Engineering*, ICSE '76, pages 61–68. USA: IEEE Computer Society Press, 1976.
- [21] K. Petersen, C. Wohlin og D.Baca. The waterfall model in large-scale development. In F. Bomarius, editor, *Product-Focused Software Process Improvement: 10th International Conference*, PROFES 2009, pages 386–400. Tyskland: Springer-Verlag Berlin Heidelberg, 2009.
- [22] H.Gomaa og D.B.H Scott. Prototyping as a tool in the specification of user requirement. In *Proceedings. 5th IEEE international Conference in Software Engineering*, ICSE '81, pages 333–342. USA: IEEE, 1981.
- [23] Martin James. *Rapid Application Development*. Macmillan Publishing Co, New York, USA, 1991.

- [24] D. H. Mills , M.Dyer og R.C Linger. Cleanroom software engineering. *Software, IEEE*, 4(5):19–25, 1981.
- [25] L.Yvonne Bajwa. Lessons learned from the use of a spiral model for an ada development effort: The software life cycle support environment. In *Proceedings of the IEEE 1989 National Aerospace and Electronics Conference, NAECON 1989*, pages 1807–1812. USA: IEEE Computer Society, 1989.
- [26] P.Beynon-Davies, C.Carne, H. Mackay og D.Tudhope. Rapid application development (RAD): An empirical review. *European Journal of Information Systems*, 8(3):211–233, 1999.
- [27] R. W. Selby, V. R. Basili og F. Terry Baker. Cleanroom software development: An empirical evaluation. *IEEE Transactions on Software Engineering*, 10(5):488–494, 1987.
- [28] H. Takeushi og I. Nonaka. The new new product development game. *Harvard Business review*, 64(1):137–146, 1986.
- [29] H.Saiedian og R.Dale. Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology*, 42(6):419–428, 2000.
- [30] Ken Schwaber. Scrum development prosess. In J.Sutherland, editor, *Business Object Design and Implementation*, pages 117–134. England: Springer London, 1995.
- [31] B.Boehm og W.J Hansen. *Spiral development: Experience, principles and refinements*, 2000. Technical report, DTIC Document.
- [32] N.Yamamichi, T. Ozeki , K.Yokochi og T. Tanaka. The evaluation of new software developing process based on a spiral modeling. In *Global Telecommunications Conference, Communications: The Key to Global Prosperity, GLOBECOM '96*, pages 2007–2012. USA: IEEE, 1996.
- [33] B.W Boehm og W.J. Hansen. *Spiral development: Experience, principles and refinements*, 2000. DTIC Document.
- [34] A.Qumer og B.Henderson-Sellers. An evaluation of the degree of agility in six agile methods and its applicability for method engineering. *Information and Software Technology*, 50(4):280–295, 2008.
- [35] P.Abramsson, O.Salo, J.Ronkainen og J.Wasta. *Agile software development methods – Review and analysis*, 2002. VTT Publication 478.
- [36] D.Cohen, M.Lindvall og P.Costa. An introduction to agile methods. *Advances in Computers, Advances in Software Engineering*, 62(1):1–63, 2004.
- [37] Peter Middleton. Lean software development: Two case studies. *Software Quality Journal*, 9(4):241–252, 2001.

- [38] L.Layman, L.Williams, D.Damian og B.Bures. Essential communication practices for extreme programming in a global software development team. *Information and Software Technology*, 48(8):781–794, 2006.
- [39] M.Paasivaara, S.Durasiewicz og C. Lassenius. Distributed agile development: Using scrum in a large project. In *Proceedings 2008 3rd IEEE International Conference on Global Software Engineering, ICGSE '08*, pages 87–95. USA: IEEE Computer Society, 2008.
- [40] L.Pries-Heje og J.Pries-Hej. Why scrum works: A case study from an agile distributed project in Denmark and India. In *Agile 2011 Conference*, pages 20–28. USA: IEEE Computer Society, 2011.
- [41] *9th Annual State of Agile Survey, VersionOne Inc*, 2015, Besøkt 28.09.2015. <https://www.versionone.com/pdf/state-of-agile-development-survey-ninth.pdf>.
- [42] S.G. Cohen og D.E. Bailey. What makes teams work: Group effectiveness research from the shop floor to the executive suite. *Journal of Management*, 23(3):239–290, 1997.
- [43] Jeff Sutherland. *The Scrum Paper: Nuts, Bolts, and Origins of Agile Process*, 2012. Versjon 1.1, Scrum Inc.
- [44] K.Schwaber og J.Sutherland. *The Scrum guide 2013*, Besøkt 12.12.2015. <http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf>.
- [45] K.Schwaber og J.Sutherland. *The Scrum guide 2011*, Besøkt 12.12.2015. <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-NO.pdf>.
- [46] B.Ramesh, L.Cao og K.Mohan. Can distributed software development be agile? *Communication of ACM*, 49(10):41–46, 2006.
- [47] J.Sutherland, A.Viktorov, J.Blount og N.Puntikov. Distributed scrum: Agile project management with outsourced development teams. In *Proceedings of the 40th Hawaii International Conference on System Sciences 2007*, HICSS '07, page 274a. USA: IEEE Computer Society, 2007.
- [48] Laurie Williams. Agile software development methodologies and practices. *Advances in Computers*, Vol 80:1–44, 2010.
- [49] N.Kurapati, V.S.C Manyam og K.Petersen. Agile software development practice adoption survey. In C.Wohlin, editor, *Agile Processes in Software Engineering and Extreme Programming*, pages 16–30. Berlin: Springer Verlag Berlin Heidelberg, 2012.
- [50] *Manifestet for smidig programvareutvikling*, Besøkt 25.11.2015. <http://www.agilemanifesto.org/iso/no/manifesto.html>.

- [51] Ikujiro Nonaka. A dynamic theory of organizational knowledge creation. *Organization Science*, 5(1):14–37, 1994.
- [52] C. Gutwin, R. Penner og K.Schneider. Group awareness in distributed software development. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work, CSCW '04*, pages 72–81. USA: ACM, 2004.
- [53] T.O.A Lehtinen, R. Virtanen, J.O Viljanen, J.O Mantyla og C.Lassenius. A tool supporting root cause analysis for synchronous retrospectives in distributed software teams. *Information and Software Technology*, 56(4):408–437, 2014.
- [54] M.J Tidemann. Post-mortems – Methodology and Experiences. *IEEE Journal on Selected Areas in Communications*, 8(2):176–180, 1990.
- [55] T.Stålhane, T.Dingsøy, G.K Hanssen og N.B.Moe. Post mortem – an assessment of two approaches. In A. I. Wang og R.Conradi, editor, *Empirical Methods and Studies in Software Engineering*, pages 129–141. Berlin: Springer Verlag Berlin Heidelberg, 2003.
- [56] F.O Bjørnson, A.I Wang og E.Arisholm. Improving the effectiveness of root cause analysis in post mortem analysis: A controlled experiment. *Information and Software Technology*, 51(1):150–161, 2008.
- [57] T.Dingsøy, N.B.Moe og Ø.Nytrø. Arugumenting experience reports with light postmortem reviews. In F.Bomarius og S.Komi-Siriviö, editor, *Product Focused Software Prosess Improvement*, pages 167–181. Berlin: Springer Verlag Berlin Heidelberg, 2001.
- [58] Arnold Kransdorff. Using the benefits of hindsight – the role of post-project analysis. *The Learning Organization*, 3(1):11–15, 1996.
- [59] L.Baird, P. Holland og S.Deacon. Learning from action: Imbedding more learning into the performance fast enough to make a difference. *Organizational Dynamics*, 27(4):19–32, 1999.
- [60] M.V.Zedwitz. Organizational learning through post-project reviews in r & d. *R & D Management*, 32(3):255–268, 2002.
- [61] M.Myllyaho , O.Salo , J.Kaariainen , J.Hyysalo og J. Koskela. A review of small and large post-mortem analysis methods. *Proceedings of the ICSSEA, Paris*, pages 1–8, 2004.
- [62] M. Drury, K.Conboy og K.Power. Decision making in agile development: a focus group study of decisions & obstacles. In *Agile Conference 2011*, pages 39–47. USA: IEEE Computer Society, 2011.
- [63] Y.Dubinsky og O.Hazzan. Software governance using retrospectives: A case study. In *Proceedings 2012 IEEE International Conference on Software Science*, pages 40–45. USA: IEEE Computer Society, 2012.

- [64] J. Webster og W.K.P Wong. Comparing traditional and virtual group forms: Identity, communication and trust in naturally occurring project teams. *The International Journal of Human Resource Management*, 19(1):41–62, 2008.
- [65] Y. I. Alzoubi , A.Q.Gill og A. Al-Ani. Empirical studies of geographically distributed agile development. *Information and Management*, 53(1):22–37, 2016.
- [66] Filippo Lanubile. Collaboration in distributed software development. In A.D Lucia og F.Ferrucci, editor, *Software Engineering*, pages 174–193. Berlin: Springer Verlag Berlin Heidelberg, 2009.
- [67] J.A Espinosa, S.A Slaughter, J.D Herbsleb og R.E Kraut. Team knowlegde and coordination in geographically distributed software development. *Journal of Management Information Systems*, 24(1):135–169, 2007.
- [68] M.Korkala og P.Abrahamsson. Communication in distributed agile development: A case study. In *33rd Euromico Conference on Software Engineering and Advanced Applications*, Profes '10, pages 203–210. USA: IEEE Computer Society, 2007.
- [69] P.J.Ågerfalk, B.Fitzgerald, H.Holmström, B. Lings, B.Lundell, E.Ó.Conchúir. A framework for considering opportunities and threats in distributed software development. In *Proceedings of the International Workshop on Distributed Software Development*, pages 47–61. Austrian Computer Society, 2005.
- [70] S.Dorairaj, J.Noble og P.Mailk. Effective communication in distributed agile software development teams. In E.Bache X.Albaladejo A. Sillitti, O.Hazzan, editor, *Proceedings in 12th International Conference XP 2011 Madrid, Spain*, pages 102–116. Berlin: Springer Verlag Berlin Heidelberg, 2011.
- [71] A.French og P.Layzell. A study of communication and cooperation in distributed software team. In *Proceedings at International Conference on Software Maintenance 1998*, pages 146–154. USA: IEEE Computer Society, 1998.
- [72] J.D. Herbsles og R.E. Grinter. Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software*, 16(5):63–70, 1999.
- [73] S. Star og J.R. Griesemer. Institutional ecology, translations' and boundary objects: Amateurs and professionals in berkeley's museum of vertebrate zoology. *Social studies of science*, 19(3):387–420, 1989.
- [74] P.R Charlie. Transferring, translating, and transforming: An integrative framework for managing knowledge across boundaries. *Organization science*, 15(5):555–568, 2004.
- [75] M.C.Figueiredo , C.R.B.Souza , M.Z.Pereira , R.Prikladnicki og J.L.N.Audy. Knowledge transfer, translation and transformation in the work of information technology architects. *Information and Software Technology*, 56(10):1233–1252, 2014.

- [76] T.Dingsøy, T.E.Fægri, T.Dybå, B.Haugset og Y.Lindsjørn. Team performance in software development. *IEEE Computer Society*, pages 12–16, 2016. accepted for publication.
- [77] N.B.Moe, T.Dingsøy og T.Dyrå. A teamwork model for understanding an agile team: A case study of scrum project. *Information and Software Technology*, 52(5):480–491, 2010.
- [78] James D. Herbsleb. Global software engineering: The future of socio-technical coordination. In *Future of Software Engineering 2007*, pages 188–198. USA: IEEE Computer Society, 2007.
- [79] A.H Anderson, R.McEwan, J.Bal og J.Carletta. Virtual team meetings: An analysis of communication and context. *Computers in Human Behavior*, 23(5):2558–2580, 2007.
- [80] Frank Maurer. A case study on the impact of scrum on overtime and customer satisfaction. In *Agile Conference 2005, ADC '05*, pages 70–79. USA: IEEE Computer Society, 2005.
- [81] Briony J.Oates. *Researching information systems and computing*. pages: 141–144, 188, 198, 209 og 214. London: SAGE Publications, 2006.
- [82] T.Dingsøy, N.B.Moe, J.Schalken og Tor Stålhane. Organizational learning through project postmortem reviews – an explorative case study. In P.Abrahamsson, N.Baddoo, T.Margaria og R.Messnarz, editor, *Software Process Improvement*, pages 136–147. Berlin: Springer Verlag Berlin Heidelberg, 2007.
- [83] X. Yu og S.Petter. Understanding agile software development practices using shared mental models theory. *Information and Software Technology*, 56(8):911–921, 2014.
- [84] T.O.A. Lehtinen, M.V.Mantyla og J.Vanhanen. Development and evaluation of a lightweight root cause analysis method (arca method) – field studies at four software companies. *Information and Software Technology*, 53(10):1045–1061, 2011.
- [85] Ken Schwaber. *The enterprise and Scrum*. Microsoft Press, USA, 2007.

Tillegg A

Verktøy for distribuert retrospektiv

Hensikt

Det skal gjennomføres casestudier på distribuert retrospektiv mellom team innen samme land og tidssone. I dette studiet gjennomføres retrospektiv hovedsakelig mellom to byer. Det trengs et solid verktøy for å gjennomføre retrospektiv, slik at brukerne av verktøy får mest mulig utbytte av det.

Scenario

Det ble utarbeidet to scenarioer som var grunnlaget for å danne kravspesifikasjoner til verktøyene. Det første scenarioet handler om et team på hvert sted som har videokonferanse med hverandre. Det siste scenarioet handler om at alle har hver sin pc og sitter med hodetelefon og deltar på videokonferanse. Valget endte på det siste scenarioet, da dette gir større mulighet for at deltakerne involveres.

Utvelgelse av verktøy

Utvelgesprosessen for testing av verktøy ble gjort i samråd med Torgeir Dingsøyr. De fleste verktøyene som er testet er hentet fra Scrumalliance¹ og etter råd fra professor Torgeir Dingsøyr. Det ble i alt valgt 18 verktøy som skulle testes nærmere på 13 kategorier, se vedlegg A.1. Kategoriene er kravspesifikasjonene som stilles til verktøyene, og er nærmere beskrevet i vedlegg A.1.1. Disse kravspesifikasjonene ble redegjort under en idémyldringsprosessen hos SINTEF, der to forskere og jeg deltok. Hovedkategoriene ble idemyldring, strukturering og votering, der disse kategoriene veide tyngst ved vurdering av verktøyene.

¹Besøkt 20.09.2015 på <https://www.scrumalliance.org/community/articles/2014/august/distributed-retrospectives>

Testing av verktøy

Vurdering av verktøyene ble gjennomført i tre faser. I første fase vurderte jeg hver og enkelt verktøy i henhold til kategoriene, og deretter rangerte jeg verktøyene etter antall oppfylte krav. I neste fase ble verktøyene testet ut samlokalisert i den rangerte rekkefølgen sammen 1-2 personer. I denne fasen ble en nærmere analyse av hovedkategoriene gjennomført, samt ble problemene tilknyttet pålogging og flerbruker problematikk identifisert, se vedlegg A.2. I siste fase ble det valgt ut et verktøy som var aktuell til casestudiet, og dette verktøyet ble testet ut distribuert. Denne testingen foregikk med 1 person i en by og 2 personer i en annen by, der kommunikasjonen foregikk via Skype. Denne testen skulle bekrefte beskrivelsen av verktøyet som ble vurdert i fase 2.

Verktøy for case studiet

På bakgrunn av informasjonen som ble gjengitt i dette notatet, endte jeg med et verktøy som var aktuell å teste ut i case studiet. Verktøyet ble *Retrium* som var en av tre verktøy som oppfylte alle tre hovedkategoriene, en nærmere beskrivelse og vurdering av verktøyet er gitt i vedlegg A.3.

A.1 Fase 1: Resultat for vurdering av verktøy

Programvare:	Vurding mot mål	Enkel input	La lett å bruke	Idemøtning	Enkelt å kommentere input	Strukturere input	Votere	Lagre produsert data	Se egne kapper	Vise referat fra siste møte	Kostnadsfritt	Video	Brukerkonto
IdeaBoardz	X	✓	✓	✓	X	✓	✓	X	X	✓	✓	X	✓
Linoit.com	X	✓	✓	✓	X	✓	✓	X	⁵ ✓	✓	✓	X	X
Stormz Hangout	X	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	X
Trello	X	✓	✓	✓	✓	✓	✓	X	⁵ ✓	✓	✓	X	X
Hackerpad	X	✓	✓	✓	X	✓	✓	X	⁵ ✓	✓	✓	X	X
Google docs	X	✓	✓	✓	✓	✓	✓	X	⁵ ✓	✓	✓	X	✓
Noteapp	X	✓	✓	✓	X	✓	✓	X	⁵ ✓	X ²	✓	X	✓
Socco	X	X	✓	✓	X	✓	X	X	X	X	✓	X	X
MindMeister	X	✓	✓	✓	X	✓	✓	X	⁵ ✓	✓	✓	X	X
Collaboration cloud	X	X	X	✓	X	✓	X	X	X	X	✓	X	X
Retrobotapp	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	X	X
Luid Meetings	X	✓	✓	✓	X	✓	X	X	X	X	✓	X	X
Realtimeboard	X	✓	✓	✓	✓	✓	✓	X	⁵ ✓	X	✓ ³	X	X
Scribbler	X	✓	✓	✓	X	✓	✓	X	X	X	X ⁴	X	X
Cosketch	X	✓	✓	✓	X	✓	✓	X	X	X	✓	X	✓
Symphonical	X	✓	✓	✓	X	✓	✓	³ ✓	⁵ ✓	✓	✓	X	X
Padlet	X	✓	✓	✓	X	✓	✓	⁷ ✓	⁵ ✓	✓	✓	X	✓
Retrium	X	✓	✓	✓	X	✓	✓	✓	X	X	✓	X	✓

¹Begrenset ved å bruke gitte alternativer.

²For mer enn 2 brukere.

³Ved bruk av betalingskonto

⁴Mulighet for samtale uten video.

⁵Ved å vise tidligere produsert data

⁶ Teanefunksjonalitet.

⁷Hvis brukeren er pålogget.

Figur A.1: Resultatet av fase 1 for vurdering av verktøy

A.1.1 Beskrivelse av kategoriene

Kategori	Beskrivelse
Vurdering mot mål:	Mulighet til å vurdere gjennomført arbeid i løpet av sprinten mot målet. Eksempelvis så gir verktøyet mulighet til å vise målet og deltakerne gir tilbakemelding på hvor langt det er igjen og dette vises som en score. Scoren legges sammen fra tidligere retrospektiv og viser totalscoren så langt i prosjektet.
Lett å bruke:	Det skal være intuitivt å ta i bruk verktøyet. Det kreves ingen former for opplæring og lignende.
Enkel input:	Det skal være lett å skrive sine tanker og ideer på det aktuelle verktøyet.
Idémyldring:	Verktøyet har funksjoner som tilrettelegger for idémyldring (brainstorming).
Enkelt å kommentere input:	Mulighet for å kommentere andres og egne bidrag.
Strukturere input:	Det er tilrettelagt for at brukeren kan strukturere input etter sitt eget ønske.
Votering:	Mulighet til å stemme over ulike lapper (input) fra alle deltakerne. Slik at det kommer fra hvilke lapper som er de viktigste å ta opp i felleskapet.
Lagre produsert data:	Mulighet til å lagre informasjonen som produseres ved bruk av verktøyet, enten som pdf, excel eller på verktøyet som brukes.
Se egne lapper:	Enhver deltaker skal ha mulighet til å se sine egne lapper (bidrag).
Vise referat fra siste møte:	Det kreves ett referat eller en oppsummering som er produsert ved å benytte verktøyet fra siste møte. Det skal være mulig å vise fram referatet (eller tidligere lagrede data) ved å bruke verktøyet.
Kostnadsfritt:	Verktøyet er ikke et kommersielt produkt.
Video:	Muligheten til å gjennomføre videosamtaler.
Brukerkonto:	Deltakerne trenger ikke en brukerkonto for å benytte verktøyet. Kravet er også oppfylt ved tilfeller der det kreves en konto pr team og resten av teamet kan bruke verktøyet uten brukerkonto.

Tabell A.1: En beskrivelse av kategoriene som brukes i figur A.1

A.1.2 Argumentasjonen for kategoriene

I distribuerte team er det vanskelig å gjennomføre retrospektiv i forhold til samlokaliserte team [15]. Retrospektivmøtene gjennomføres annerledes på distribuerte team i forhold til tradisjonelle team, bruk av verktøy er ett av mulighetene. Mine kravene til bruk av verktøy i distribuerte retrospektiv er beskrevet i vedlegg A.1.1. Det er definert flere krav til verktøy og bakgrunnen for disse skal begrunnes i de neste avsnittene.

Vurdering mot mål er definert som et krav til verktøy på bakgrunn av det er viktig at brukerne kan måle sine prestasjoner mot målet. På denne måten kan teamet bevisstgjøres på progresjonen de har i forhold til målet.

Enkel input og lett å bruke er to krav som er formulert med hensyn til brukervennlighet. Lehtinen viser til at brukervennlighet er en avgjørende faktor for at verktøyet blir nyttig og at verktøyet brukes korrekt [53]. Dette innebærer at det skal være intuitivt å bruke verktøyet.

Det er viktig at verktøyet har en funksjonalitet som støtter *idemyldring*. Dette er fordi idemyldring er en prosess der deltakerne i teamet kan samle sine tanker og idéer fra forrige arbeidsperiode og synliggjøre det for andre deltakere i teamet. Derby og Larsen påpeker at uten felles mentalmodell i teamet, så vil de kun fokusere på sine egne utfordringer og behov [7]. Dette fører til at utfordringene og behovene til teamet ikke blir løst i like stor grad. Idemyldring er et utgangspunktet for å få felles mental modell for hele teamet i retrospektivmøte [7]. Det skal være *enkelt å kommentere input* er også ett av kravene som ble stilt til verktøyet. Dette kravet er stilt fordi det skal være mulig å kommentere på lappene underveis for å øke forståelse rundt konteksten i lappene.

Strukturere input innebærer at verktøyet har funksjonaliteten for å gruppere lappene. Uten denne muligheten vil lappene være uoversiktlige og det vil være vanskelig å stemme over disse lappene i etterkant. I tillegg det være tidskrevende å bruke verktøyet uten å ha muligheten til å gruppere lappene eller at det fører til teamet må stemme på en og en lapp. Derby og Larsen påpeker at det er viktig å disponere tidsbruken på et retrospektiv [7].

Votering er et krav til verktøyet da det hjelper med å prioritere lappene som har blitt samlet inn i løpet av idemyldringen, slik at teamet kan fokusere på de utfordringene og behovene som er viktigst for dem [7, 84].

Lagre produsert data er en funksjonalitet for å ta vare utbyttet av retrospektivmøtene. Dingsøyr foreslo at det fysiske resultatet av retrospektivmøtene kan være en liste med anbefalinger for å sikre læring eller en rapport som beskriver prosjektet avhengig om formålet er kunnskapsspredning i bedriften eller forbedringer [14]. *Vise referat fra sist møte* er et krav til verktøyet for å ha muligheten til å gjennom resultatet av forrige retrospektiv (forbedringstiltakene). Denne funksjo-

naliteten er nødvendig fordi menneske er ikke i stand til å huske all informasjon om foregående arbeidsperiode. Dette er på grunn av at hukommelsen avtar med tiden [59].

Muligheten til å *se egne lapper* er avgjørende når lappene skal presenteres til de andre deltakerne. Når ett distribuert team er stort, og skal gjennomføre retrospektiv så kan det være vanskeligheter med å skille mellom hvem som har skrevet hvilke lapper.

I et distribuert team har deltakerne ikke muligheten til å møte hverandre fysisk, derfor bør kommunikasjonen mellom teamene være mest mulig innholdsrik. *Video* er derfor definert som ett krav til verktøyene som de bør tilfredsstillere for å få mest mulig innholdsrik kommunikasjon. I situasjoner er dette ikke er mulig kan telefonkonferanse erstattes, men dette er betydelig mindre innholdsrik i forhold til videokonferanse [8].

Kostnadsfritt innebærer at det gratis å bruke verktøyet og er et krav til verktøyet da det viser seg at systemutviklere hopper over retrospektivmøtene på grunn av økonomiske årsaker [10]. *Brukerkonto* er et krav som innebærer at kun en eller ingen trenger brukerkonto for å benytte verktøyet. Dette kravet stilles på bakgrunn av det blir mye administrativt å holde orden på dersom alle deltakerne skal ha ett eget brukernavn og tilhørende passord. Det er ønskelig at gjennomføringen av retrospektivmøtet skal være så enkelt så mulig, derfor stilles dette kravene.

A.2 Fase 2: Resultat for vurdering av verktøy

Verktøy	Idémyldring	Votering	Strukturering	Annet
Retrium	-Det som er skrevet på lappene er kun synlig for brukeren selv, men alle kan bare se om lappene er blanke eller ikke.	-Hver bruker har 5 stemmer og brukerne kan ikke se hva de andre har stemt.	-Mulighet til å gruppere lappene før voting på tvers av kategoriene.	-Passord-beskyttelse. -Begrenset kategorier eller maler som kan benyttes.
Retrobotapp	-Vanskelig å flytte lapper. -Umulig å redigere lapper -Uoversiktlig med mange lapper	-Støtter votering -Lister ikke i prioritert rekkefølge.	- Kan gjøres ved å bruke begrenset alternativer. -Mulighet til å velge 8 forskjellige forhåndsdefinerte maler.	-Innloggingsutfordringer -Varierende øvelser.
IdeaBoardz	-Raskt å ta opp ideer -Anonymisert input	-Ingen maksimal grense for antall en kan gi. -Eksport gir rangering etter stemmer.	-Enkelt å gruppere. -Viser max 4 lapper i grupper. -Kan sortere etter antall stemmer.	-Kan fokusere på ett område. -Kan endre på kategoriene underveis.
Realtimeboard	-Knotete å lage lapper -Anonymisert input		-Fleksibelt	-Mer tegneverktøy enn idedugnad.
Stormz Hangout	-Litt knotete å skrive lapper -Enkelt å skrive lapper.	-Kan gi en eller 3 stemmer pr bruker.	-Umulig strukturere lappene. -Fleksibelt	-Vanskelig å bruke verktøyet. -Mer arbeidsverktøy enn en verktøy for retrospektiv.
Padlet	-Enkelt å skrive lapper. -Godt tilrettelagt for idémyldring		-Fleksibelt -Tungvint å se egne lapper, da du må trykke på hver lapp for å se det.	-Mer arbeidsverktøy enn en verktøy for retrospektiv. -Passordbeskyttelse og uten pålogging.

Tabell A.2: Vurdering av 7 beste verktøyene i de tre hovedkategoriene

Fase 2: Resultat for vurdering av verktøy

Verktøy	Idémyldring	Votering	Strukturering	Annet
Notteapp	-Svært enkelt å skrive lapper. -Anonymisert input		-Fleksibelt	-Mer arbeidsverktøy enn en verktøy for retrospektiv. -En vegg, der alle lapper postes fleksibelt.
Symphonical	-Raskt å skrive lapper -Kan endre og flytte lapper. -Anonymisert input		-Kan velge kategori og tilhørende deltakere. -En lapp settes på en deltaker i en gitt kategori.	-Mer arbeidsverktøy enn en verktøy for retrospektiv.
Google docs				-Synkronisert tekstbe-handlingsverktøy.
Limoi.com	-Enkelt å skrive lapper.		-Fleksibelt	-Mer arbeidsverktøy enn retrospektiv verktøy.
Mind Meister	-Litt knottete å skrive lapper.		-Begrenset	-Mer tankekart enn retrospektiv verktøy.
Hackerpad			-Fleksibelt, med mulighet for søkefunksjonalitet.	-Synkronisert tekstbe-handlingsverktøy.
Scribblar			-Fleksibelt, men tegne-verktøy.	-Mer tegneverktøy enn idedugnad.
Cosketech			-Fleksibelt, men tegne-verktøy.	-Mer tegneverktøy enn idedugnad.
Socco				-En programvare for virtuell hjemmekontor.
Luid meeting				-En programvare for å gjennomføre telefon konferanser.
Collaboration Cloud				-Ingen nyttig programvare, da funksjonaliteten ikke er på plass.

Tabell A.3: Vurdering av 11 dårligste verktøyene i de tre hovedkategoriene

A.3 Nærmere beskrivelse av verktøyet ”Retrium”

Dette verktøyet har 3 forskjellige forhåndsdefinerte maler som benyttes for å gjennomføre retrospektiv. De eneste forskjellene på malene, er navnet på kategoriene og antall kategorier. Det betyr at funksjonaliteten er den samme. Det er kun en bruker som trenger brukerkonto for å opprette et retrospektivmøte og deretter distribuere linken til deltakerne i teamet.

Malene som kan benyttes for å gjennomføre retrospektiv er:

1. Mad, Sad, Glad
2. Start, Stop, Continue
3. 4L: Liked, Learned, Lacked og Longed for.

Selve retrospektivmøtet foregår i 4 faser i følgende rekkefølge:

1. Tenking
2. Gruppering
3. Votering
4. Diskusjon

I første fase skal deltakerne i teamet gjennomføre idémyldring. I neste fase har kun deltakeren med brukerkonto muligheten til å gruppere lappene, mens resten av deltakerne ser på eller hjelpe til manuelt. I tredje fase skal deltakerne stemme over de grupperte lappene. I siste fase skal teamet gå gjennom grupperte lapper, der begynnes med gruppen med flest stemmer. Hensikten er å diskutere seg fram til forbedringstiltak. Det er forøvrig ikke mulig å prioritere og tildele ett forbedringstiltak.

Kravspesifikasjonene til verktøyet er nærmere beskrevet på tabellen på neste side, i forhold til kategoriene i tabellene A.4 og A.5.

Vurdering av Retrium i forhold til kravene

Kategori	Fordeler	Ulemper
Vurdering mot mål:		-Ingen støtte
Lett å bruke	-Trenger forholdsvis ingen/lite opplæring før verktøyet kan tas i bruk.	
Enkel input:	-Enkelt å skrive ned lapper i verktøyet. -Skiller mellom ulike brukere med forskjellige bakgrunnsfarger på lappene.	-En bruker kan skrive ubegrenset antall lapper, og dette kan ikke endres på.
Idémyldring:	-Verktøyet har funksjonaliteten for å gjennomføre idémyldring.	-Under idemyldringen er teksten på lappene kun tilgjengelig for brukeren selv og skjult for andre brukere. -Ingen mulighet til presentere lapper.
Enkelt å kommentere input:		-Ingen støtte
Strukturere input:	-Muligheten til å gruppere lapper (på tvers av kategorier). -Viser alle lapper i en gruppe.	-Kun flytte en lapp om gangen. -Gruppering av alt på en gang. -Uoversiktlig ved mange lapper. -Kan ikke sette navn på en gruppe med lapper. -Kun brukeren med brukerkonto kan gruppere lappene.
Votering:	-Hver bruker kan bruke inntil 5 stemmer. -Votering er anonymt gitt at bakgrunnsfargen til brukeren er ukjent for andre brukere.	-Ingen støtte for egendefinert grense for antall stemmer som kan brukes pr bruker.
Lagre produsert data:	-Det eneste som lagres fra møtet er en liste med tiltak som teamet har utarbeidet.	-Ingen mulighet til å eksportere tiltakene som pdf og lignende. -Ingen støtte for å lagre resultatet av voteringen av grupperte lapper i verktøyet eller eksternt.

Tabell A.4: Vudering av verktøyet Retrium i forhold til kategoriene på tabell A.1

Kategori	Fordeler	Ulemper
Se egne lapper:	-Kan se egne lapper.	
Vise referat fra siste møte:	Kun mulig å se tiltakene fra tidligere møtet.	Ingen støtte for å vise resultatet av grupperingen av lappene.
Kostnadsfritt:	Det kan gjennomføres inntil 3 retrospektivmøter gratis med all funksjonalitet tilgjengelig pr brukerkonto.	Tjenesten koster 5 amerikanske dollar per bruker.
Video:		-Ingen støtte
Brukerkonto:	-Kun en person trenger brukerkonto og vedkommende har administrator rettigheter. -Den som har brukerkonto sender URL-linken til andre deltakere slik de får tilgang til retrospektivmøtet. -Kan ha passordbeskyttelse for å ha få tilgang til retrospektivmøtet.	

Tabell A.5: Vudering av verktøyet Retrium i forhold til kategoriene på tabell A.1

A.4 Case D2: Bruk av verktøyet Retrium

Det er gjennomført ett retrospektivmøtet for case D2. I dette vedlegget beskrives hvordan deltakerne i case D2 brukte verktøyet, Retrium for å gjennomføre retrospektivmøtet. I dette retrospektivmøtet er det 3 deltakere, to fra by1 og en fra by2. Hver av deltakerne hadde med seg sin egen PC, og det var en videokonferanse mellom by1 og by2. Det ble benyttet en ekstern fasilitator for å gjennomføringen av retrospektivmøtet. Jeg og fasilitatorene er tilstede for møtet fra by1.



Figur A.2: Deltakeren har selv ikke skrevet noe, og innholdet i andres deltakerens lapper er skjult.

Retrospektivmøtet begynner med at fasilitatoren distribuerer url til verktøyet slik at deltakerne kan delta på den virtuelle retrospektivmøtet. Fasilitatoren velger malen ”start, stop og continue” for å gjennomføre retrospektivmøtet. Kun to av disse tre kolonnene ble brukt til gjennomføringen av retrospektivmøtet. Continue-kolonnen brukes til å skrive lapper om hva som har gått bra (positive lapper) i løpet av den siste måneden, mens stop-kolonnen brukes til å skrive om hva kunne ha gått bedre (forbedringslapper) i løpet av samme tidsperiode. Deltakerne kan kun se hva de selv skriver, men lappene som de andre deltakerne har skrevet er skjult for dem, se figur A.2. Når deltakerne har skrevet ferdig lappene, så kan de se hva de andre deltakerne har skrevet, se figur A.3

Det neste steget er grupperingen av virtuelle lappene både når det gjelder positive lapper og forbedringslapper, se figur A.4. Denne grupperingen kan kun gjennomføres av brukeren som administrator rettigheter, som i dette tilfellet er den eksterne fasilitatoren. Grupperingen av positive og forbedringslapper foregår hver for seg, selv om verktøyet støtter muligheten til å gruppere lappene samlet. Det er ikke mulig å navngi de grupperte lappene (temaene) i dette verktøyet.

I neste steg skal deltakerne votere på de grupperte lappene. Deltakerne har fem stemmer hver for alle grupperte lappene, som inkluderer både positive og forbedringslapper, se figur A.5. Det er ikke muligheten til å se hva de andre deltakerne stemmer før selve avstemmingen er over. Grupperte lapper refereres som temaer fra nå.

Utarbeidelse av forbedringstiltak er det neste steget. Verktøyet støtter ikke muligheten til å diskutere de ulike temaene, da den ikke har kommunikasjonsmekanisme som chat eller videokonferanse. Deltakerne gikk gjennom hvert tema og utarbeidet forbedringstiltak, se figur A.6. Der både positive og forbedringslapper presenteres samlet, der teamet med flest stemmer presenteres først og temaet som har færrest stemmer presenteres sist. Hver stjerne indikerer antall stemmer, og fargen på stjernen brukes for å skille mellom de ulike brukerne. Et eksempel på en utfordring som deltakerne i case D2 diskuterte var knyttet til duplikater av dokumenter og som førte til merarbeid. Det ble utarbeidet et forbedringstiltak som beskrives av U_{D2} : ”Ha ett tiltak på å fjerne duplikater innenfor installasjonsdokumentasjon”.

Siden verktøyet ikke støtter muligheten til å prioritere tiltak eller tildele forbedringstiltak til bestemte deltakere, så gjennomføres dette manuelt. Fasilitatoren skrev navnet på den personen som skal ha ansvaret for gjennomføring av hver av forbedringstiltakene. Deltakerne har tatt skjermklipp underveis da de mister lappene etter at retrospektivmøtet er gjennomført. Resultatet av retrospektivmøtet ved bruk av Retrium er et sett med forbedringstiltak som deltakerne kan krysse av når de har gjennomført det, se figur A.7.



Figur A.3: Lappene er synlige etter at alle har skrevet ferdig lappene sine.



Figur A.4: Gruppering av positive og forbedringslapper



Figur A.5: Votering av de grupperte lappene

discussing...

Completed 1 of 10 topics (10%)
Completed 4 of 15 votes (27%)

★★★

Jevnlige teammøter/retrospektiver Continue

Fortsette med faste retrospektiv Continue

Folk

☰ New action items (4)

Planlegge retrospektiver lenger fram i tid
Vurdere Retrium for hele team
Fjerne duplikater i installasjonsdokumentasjon
Be folk på heapchat om å legge dokumentasjon på confluence

Figur A.6: Utarbeider forbedringstiltak på temaene

Here are your new action items.

Any last minute changes?

☰ New action items (7)

Dele jira-skjerm under stå-opp-møter
Ukeplan som viser hva folk vil jobbe med
Fortsette å rullere på oppgaver
Planlegge retrospektiver lenger fram i tid
Vurdere Retrium for hele team
Fjerne duplikater i installasjonsdokumentasjon

Figur A.7: Viser oversikt over forbedingsiltakene fra gjennomført retrospektivmøte.

Tillegg B

Scrum rammeverket

Scrum er prosessrammeverk for å utvikle komplekse programvareprodukter. Dette rammeverket har blitt brukt siden begynnelsen av 1990-tallet og er den mest brukte metoden for å utvikle programvare på verdensbasis. Scrum består av et antall roller, artefakter og møter, og termologien knyttet til Scrum er presentert nedenfor. Schwaber og Sutherland har utgitt en Scrum guide som informasjonen i dette vedlegget er basert på [44, 85].

B.1 Sprint

En sprint varer mer eller mindre enn 4 uker (en måned) som velges selv av utviklingsteamet. Et prosjekt deles gjerne opp i flere sprinter med samme varighet gjennom et helt utviklingsforløp. Hver sprint har en definisjon på hva som skal lages, et design og en fleksibel plan for å lage det. Det skal ikke foretas endringer som hindrer at teamet når sprintmålet, og omfanget av arbeid som gjennomføres avklares og forhandles mellom produkteier og utviklingsteamet. I løpet av en sprint foregår de flere forskjellige møter tilknyttet rammeverket og er beskrevet nedenfor.

B.2 Roller

Scrum rammeverket består av tre forskjellige roller. Disse rollene er produkteier, scrum-master og utviklingsteamet.

Produkteier er en person som sørger for at alle interessentene blir fornøyde. Produkteieren har en forretningsperspektiv i forhold til gjennomføringen av prosjektet – skal sørge for at kunden for å maksimere verdien for produktet og arbeidet til utviklingsteamet.

I tillegg er produkteieren den eneste ansvarlige personen for produktkøen som innebærer følgende ansvarsområder:

- Å uttrykk produktkøelementene klart og forståelig.
- Å prioritere produktkøelementene slik at de ligger i en ordnet rekke for best mulig oppnå målsetninger.
- Sørge for at arbeidet til utviklingsteamet gir mest mulig verdi.
- Sørge for at produktkøen er godt synlig og sikre at den viser hva utviklingsteamet kommer til å jobbe med i neste omgang. I tillegg passe på at utviklingsteamet forstår elementene i produktkøen godt nok.

Scrum master er en person som sørger for utviklingsteamet etterlever Scrum-rammeverket. Scrum-master har flere arbeidsoppgaver for produkteieren, utviklingsteamet og organisasjonen. Scrum-master skal blant annet bistå produkteieren på oppfordring og finne gode teknikker for produktkøhåndtering. Når det gjelder utviklingsteamet skal scrum-masteren fjerne eventuelle hindringer som hindrer framdriften til utviklingsteamet. Det kan nevnes at scrum-master har ansvaret for å trene utviklingsteamet i selvorganisering og tverrfaglighet samt lære dem å lage produkter med høy verdi.

Utviklingsteamet er tverrfaglig, selvstyrte og selvorganiserte. Teamet består typisk av tre til ni medlemmer, og dette er avhengig av om teamet har nok kompetanse og kapasitet til å gjennomføre prosjektet. Oppgaven er å finne ut hvordan funksjonaliteten skal utvikles og gjennomføre det innen estimert tid og frist. Teamet har ingen leder eller roller, og derfor ligger ansvaret for fremdriften hos teamet som helhet. Teamet er selvorganiserende som betyr at ingen har lov til å fortelle hvordan utviklingsteamet gjør produktkøelementer om til potensielt leverbare inkremitter med funksjonalitet.

B.3 Møter

Scrum-rammeverket inneholder 4 forskjellige møter, som gjennomføres i løpet av en sprint. Disse møtene er sprint planlegging, daglig scrum, sprint review og sprint retrospektiv. Enkelte møter gjennomføres oftere enn andre. Dagligmøte gjennomføres hver arbeidsdag, mens de andre møtene gjennomføres enten ved begynnelsen av sprinten eller på slutten av sprinten. Møtene er presentert nedenfor.

Sprint planlegging er et møte som foregår i begynnelsen av sprinten. I dette møtet planlegges arbeidet for sprinten. Møte består av to deler, der den første delen går ut hva som skal bli levert i denne sprinten, mens den andre delen går ut hvilket arbeidet som er nødvendig for å lage dette inkrementet – tidsbruken er fordelt likt mellom begge delene.

I første delen forklarer produkteieren hvilke elementer som ligger høyest prioritert i produktkøen. Formålet er at både produkteier og teamet sammen velger ut hvilke oppgaver som skal gjennomføres i løpet av sprinten samt at hele utviklingsteamet får en fellesforståelse av arbeidet som skal gjennomføres. Det er kun utviklingsteamet som vurderer hva de er i stand til å fullføre, og etter at de har vurdert det, lagres det en sprintmål.

Etter å ha bestemt omfanget av arbeidet i første del, skal dette realiseres i den andre delen av møtet. Dette gjøres ved å velge elementer ("oppgaver") fra produktkøen. De valgte elementene som gjerne er høyest prioritet i produktkøen flyttes fra produktkøen til sprintkøen, der de nærmere beskrives. På slutten av møtet skal utviklingsteamet kunne forklare produkteier og scrum-master hvordan de skal selvorganisere seg for å oppnå sprintmålet og lagre produktinkrement.

Daglig scrum, gjøres kalt dagligmøte eller stand-up møte er et møte som varer maksimalt 15 minutter. I løpet av møtet skal hvert teammedlem svare på tre forskjellige spørsmål for å synkronisere arbeidet sitt og planlegge arbeidshverdagen. Møtene holdes ofte på samme sted og til samme tid, og spørsmålene som skal svares er:

- Hva har du gjort siden forrige møte?
- Hva skal du gjøre til neste møte?
- Hvilke hindringer har du?

Sprint review er et møte som holdes på slutten av sprinten for inspisere produktinkrementet og tilpasse produktkøen ved behov. Utviklingsteamet, kunden og interessentene samarbeider om det som ble gjort i løpet av sprinten og ut fra dette hva som skal gjøres i neste sprint. Det brukes 4 timer for en sprint som har vart i en måned, og mindre tid hvis sprinten var kortere. Resultatet av sprintreview er en revidert produktkø som definerer sannsynlig utvalgte produktkø for neste sprint. Sprintreviewet inkluderer følgende elementer:

- Produkteier identifiserer alt arbeid ble ferdig eller ikke ferdig.
- Det diskuteres hva som gikk bra i sprinten og hvilke utfordringer de møtte og hvordan det ble løst.
- En demonstrasjon av arbeidet som ble gjennomført og er ferdig. Deretter svare på spørsmål fra interessentene.
- Produkteier redegjør for sannsynlige leveransedatoer basert på progresjon hittil og hele gruppa samarbeider hva som skal gjøres neste gang

Sprint retrospektiv er det avsluttende møtet i sprinten som tar ett tilbakeblikk på det som har skjedd i løpet av sprinten. Møtet skal avholdes rett etter sprint review før neste sprint begynner med sprint planlegging. Møtet skal vare i 3 timer for en månedssprinter, og kortere dersom sprintene er kortere.

Hovedhensikten med retrospektiv:

- Undersøke hvordan siste sprint med fokus på mennesker, forholdet mellom mennesker, prosesser og verktøy.
- Identifisere viktigste positive og negative erfaringer.
- Lage en plan for å ta i bruk forbedringene.

B.4 Artefakter

Det er tre forskjellige artefakter i scrum-rammeverket, og disse er produktkø, sprintkø og burndown chart.

Produktkøen er en prioritert liste over det som behøves i produktet og er den eneste kilden som beskriver endringene som gjøres i produktet. Produkteieren er den ansvarlige for produktkøen. Produktkøen er aldri komplett og er stadig i endring, og den eksisterer så lenge produktet eksisterer. Elementene i produktkøen kan være alle funksjonene, kravene forbedringer og bug-fixer som produktet trenger i framtiden. Elementene i produktkøen har en beskrivelse i form av en estimat og en rekkefølge. De øverste elementene i produktkøen er mer detaljerte enn de lavere, og derfor er det enklere å presisere estimatene på øverste elementene. Det er viktig å legge merke til at det er kun utviklingsteamet som estimerer elementene og produkteier bistår for klargjøring og avveininger.

Sprintkøen er en liste over produktelementer fra produktkøen som skal gjennomføres i løpet av sprinten. Sprintkøen opprettes i sprint planleggingsmøtet, der utviklingsteamet henter ut produktelementer fra produktkøen til sprintkøen. Elementene i sprintkøen kan endres ettersom utviklingsteamet mener at det planlagte arbeid ikke er nødvendig for å oppnå sprintmålet, kan det legges tilbake til produktkøen. Det er kun utviklingsteamet som kan forvalte sprintkøen. Sprintkøen er en godt sanntidsbilde for hvordan utviklingsteamet av gjenstående arbeid.

Burndown chart er en grafisk fremstilling over gjenstående ”tid” på gjenværende oppgaver i nåværende sprint.

Tillegg C

Beskrivelse av temaene i retrospektivmøter

Tema	Beskrivelse
Koordinering av oppgaver	Beskriver håndteringen av oppgavene som skal gjennomføres i løpet av prosjektet. Eksempel på dette kan være utfordringer knyttet til bruk av arbeidsverktøyet Jira for å koordinere arbeidsoppgaver til teamet. Med andre ord prioritering av oppgaver, tydeligheten på oppgaven og strukturen for håndtering av arbeidsoppgaver i Jira.
Validering og verifisering	Dette innebærer at det diskuteres om fagområdet testing og kvalitetssikring på programkoden. Med andre ord inkluderer dette bugs, teknisk gjeld, mangel på kunnskap, teststrategi og code review.
Ressurser	En betegnelse om teamet har tilstrekkelig antall deltakere i teamet for å få gjennomført prosjektet relativt effektivt
Ansvarsområde	Omfatter at de enkelte deltakerne diskuterer ansvaret for bestemte oppgaver og roller. Eksempler for dette er hvem som har ansvaret for hva, og hva som var årsaken til en bestemt oppgaven ikke ble gjennomført.
Kunde	Teame tar opp temaer relatert til utfordringer knyttet til kunderelasjon, eksempelvis i hvilken grad kundene klager på utført arbeid, og teametsforhold til kunden.

Tabell C.1: En beskrivelse av temaene som er brukt i figur 6.5