

UNIVERSITETET I BERGEN

MASTEROPPGÅVE

Brukarhistorier med meining

Forfattar:

KRISTIAN VÅRDAL

Rettleiar:

BJØRNAR TESSEM

ved

Institutt for informasjons- og medievitenskap

29. mai 2014

Universitetet i Bergen

Samfunnsvitenskapelig fakultet
Institutt for informasjons- og medievitenskap

Mastergrad

Brukarhistorier med mening

av KRISTIAN VÅRDAL

Samandrag

Formålet med dette studiet var å undersøkje om bruken av semantiske teknologiar bidreg til auka domeneforståing for eit utviklarlag. Difor vart det utvikla ein applikasjon, *Semantizer*, som nyttar den semantiske teknologien *WordNet*, applikasjonen *SynsetTagger* og ein metode for automatisk orddisambiguering for å velje riktig synonym til ord. For å evaluere om *Semantizer* bidreg til auka domeneforståing, vart eit eksperiment gjennomført. Eksperimentet bestod av tjue deltakarar, og etter dette vart det gjennomført intervju med tre av deltakarane. Det vart ikkje funne at *Semantizer* har ein effekt på domeneforståinga, men etter intervjua verkar alle intervjuobjekta positive til *Semantizer* og måten *WordNet* kan nyttast til å auke domeneforståinga i eit utviklarlag.

Forord

Først og fremst vil eg gjerne takke min rettleiar Bjørnar Tessem for god ide, humør og tilbakemeldingar undervegs i både utvikling av prototype og oppgåve. Det herskar ingen tvil om at eg valte riktig rettleiar til dette studiet.

Takk til Rune Arntsen og administrasjonen i 5. etasje for rask og god hjelp med å skaffe rom til eksperiment og intervju.

Hjarteleg takk til mi kjære søster Karoline for gjennomlesing av oppgåva med sitt kritiske blikk, og for å alltid ha eit ledig augeblikk for diskusjon og latter!

Ellers må ei stor takk rettast til alle dei geniale medstudentane på rom 635. Verdifulle tilbakemeldingar, gullkorn, utbrot og interesser på dette rommet har bidrege til å forme denne masteroppgåva på alle måtar.

Til sist, takk til alle deltakarane i eksperiment og intervju.

Innhold

Samandrag	i
Forord	ii
Innholdsliste	vi
Figurar	vi
Tabellar	vii
1 Introduksjon	1
1.1 Mål for forskingsprosjektet	2
1.2 Forskingsspørsmål	2
1.3 Oppgåvestruktur	3
2 Teori	4
2.1 Smidig metodikk	4
2.2 Scrum	5
2.2.1 Scrum-master og utviklarlaget	7
2.2.2 Produkteigar	7
2.2.3 Scrum-artefaktar: brukarhistorier og Scrum-vegg	9
2.2.4 Viktige komponentar i utviklarlaget	10
2.3 Naturleg språk-krav	11
2.3.1 Uregelmessige krav	11
2.3.2 Fullstendige krav	12
2.4 Ontologi	12
2.4.1 WordNet	14
3 Problemstilling og metode	18
3.1 Problemstilling og motivasjon	18
3.2 Designvitskap	20
3.3 Eksperimentell design	22
3.3.1 Validitet	24
3.3.2 Pålitelegheit	29
3.4 Intervju	29
4 Semantizer-applikasjonen	31

4.1	Målet med Semantizer	31
4.1.1	Utviklingsmetode	31
4.1.2	Arkitektur	32
4.2	Automatisk merking med SR-AW	33
4.3	SynsetTagger	34
4.4	Endeleg utforming av Semantizer	36
4.4.1	Brukarhistorier og automatisk orddisambiguering	37
4.4.2	Søk etter ord i WordNet	38
4.4.3	Synonym definert av produkteigar	41
4.5	Utfordringar undervegs	43
4.6	Utviklingsmiljø	43
5	Forskningsdesign	45
5.1	Designforskning	45
5.2	Eksperimentdesign	47
5.2.1	Avhengige- og uavhengige variablar	48
5.2.2	Val av forskingssubjekt	49
5.2.3	Eksperimentoppsett	49
5.2.4	Intern validitet	52
5.2.5	Ekstern validitet	54
5.2.6	Konstruksjonsvaliditet	54
5.2.7	Konklusjonsvaliditet	55
5.3	Instrument	56
5.3.1	Datasett	56
5.3.2	Samtykkeskjema og spørjeskjema	58
5.3.3	Analyse	59
5.4	Pilotstudie	59
5.4.1	Funn i pilotstudiet	60
5.5	Intervju	61
6	Resultat og diskusjon	63
6.1	Analyse av kvantitative data	63
6.2	Diskusjon	65
6.2.1	Brukarhistorier	65
6.2.2	Spørjeskjema	66
6.2.3	Innspel frå intervju	68
6.3	Evaluering av eksperimentet	72
7	Konklusjon og vidare arbeid	75
7.1	Konklusjon	75
7.2	Refleksjonar over studiet	76
7.3	Vidare arbeid	77
	Referansar	77
A	Spørjeskjema for Command and Control System	81
B	Spørjeskjema for Health insurance	84

C Brukarhistorier for Command and control system	87
D Brukarhistorier for Health insurance	95
E Intervju	101

Figurar

2.1	Scrum-skjelett, som vist i Schwaber, 2004, s. 6	6
3.1	Korleis kontrollere viktige faktorar i eit eksperiment	23
4.1	Applikasjonen <i>SynsetTagger</i> av Veres, Johansen og Opdahl (2013).	35
4.2	Forklaring til “user” i popup-vindaue.	36
4.3	Skjermbilete av <i>Semantizer</i> -applikasjonen	37
4.4	Skjermbilete av brukarhistoriene og funksjonar i <i>Semantizer</i>	38
4.5	<i>WordNet</i> -representasjon av termen “user”, med tre ulike synonym.	39
4.6	Ekspandert JTree med hyponym og hypernym for termen “user”.	41
4.7	JTree med brukartilpassa synonym, “staff user”.	42
5.1	Helseforsikring modellert i eit UML-diagram ¹	57
5.2	Kartotek kort med brukarhistorier for kommando- og kontrollsystemet.	58
6.1	Relasjonar til “activity”-noden.	70

Tabellar

2.1	Grunnar til at prosjekt feilar, som vist i Aurum og Wohlin, 2005, s. 315	8
2.2	Leksikalsk matrise som syner relasjonane mellom ordform og meiningane til eit ord	15
3.1	Retningsliner for designvitskap	21
3.2	Truslar mot intern validitet	25
3.3	Truslar mot ekstern validitet	26
3.4	Truslar mot konstruksjonsvaliditet	27
3.5	Truslar mot konklusjonsvaliditet	28
4.1	SR-AW resultat i prosent Pedersen og Kolhatkar, 2009, s. 4	34
5.1	Eksperimentoppsett, mellom-gruppe design	50
5.2	Opphavleg tenkt eksperimentoppsett, innanfor-gruppe design	51
5.3	Fordelar og ulemper med mellom-gruppe og innanfor-gruppe design	52
6.1	Middelverdi og sum for domena	64
6.2	Kommando- og kontrollsystem: applikasjon vs. ingen applikasjon	64
6.3	Statistikk for kommando- og kontrollsystem	64
6.4	Helseforsikringsdomenet applikasjon vs. ingen applikasjon	65
6.5	Statistikk for helseforsikringsdomenet	65
6.6	Frekvenstabell for kommando- og kontrollsystem	67
6.7	Frekvenstabell for helseforsikringsdomenet	68

Kapittel 1

Introduksjon

Ei utfordring med smidige metodar i systemutvikling er at ein dokumenterer i mindre grad enn med tradisjonelle metodar. Ein leverer ikkje store krav-, domenemodell- eller designdokument fordi ein meiner dette er for kostbart i høve til nytte. Tradisjonelle metodar støttar seg derimot på lange og krevjande prosessar for å utarbeide kravspesifikasjonar som inneheld alle krava som vert stilt til eit produkt. I tillegg leverer ein gjerne domenemodellar i form av ordlister og UML-modellar. Ordlistene forklarar spesifikke termar, medan forskjellige UML-diagram kan representere ulike eigenskapar og objekt i eit informasjonssystem (Osada, Ozawa, Kaiya & Kaijiri, 2007).

Dette har blant anna den smidige metoden Scrum gått bort i frå, og lener seg heller mot korte og konsise krav i form av brukarhistorier. Ein av fordelane med brukarhistorier er at ein tvingar kunden til å utarbeide konsise og nøyaktige krav. Tilhengarar av Scrum vil hevde at dette medfører at dokumentasjonen blir kortare og enklare for utviklarlaget å forstå og arbeide med. Tilhengarar av dei tradisjonelle metodane vil kanskje påstå at brukarhistorier ikkje tilbyr nok informasjon til utviklarane, som av den grunn ikkje forstår eller oppfattar domenet like godt som dei burde. Med utgangspunkt i Scrum og brukarhistorier, vert det i dette prosjektet forska på korleis ein kan nytte semantiske teknologiar til å hjelpe utviklarlaget med å betre forstå domenet ein utviklar for.

1.1 Mål for forskingsprosjektet

Målet med forskingsprosjektet er å finne ut om ein ved å nytte semantiske teknologiar kan forsterke utviklarlaget sin domenekunnskap. For å forstå dette vart det i prosjektet utvikla ein applikasjon kalla *Semantizer* som spesifikt skal tileigne brukarhistorier meir informasjon. For å få dette til vart den semantiske teknologien *WordNet*¹ nytta. *WordNet* er ein leksikalsk database av ord, og eit hierarki av relasjonane mellom desse orda. Eit ord eller objekt kan ha fleire tydingar, såkalla synonym, og det er konteksten som avgjer kva synonym som er riktig. Ein produkteigar, slik det er definert i Scrum er ansvarleg for å representere kunden og sluttbrukarane sine interesser i produktet. Produkteigar er ansvarleg for å lage brukarhistorier for eit utviklingsprosjekt. Eit mål med *Semantizer* er at orda i brukarhistoriene automatisk skal tileignast eit eller fleire synonym i *WordNet*. Ein produkteigar kan deretter undersøkje ordet, og slå fast om det gjeve synonymet stemmer overeins med oppfatninga produkteigaren sjølv har i ein gjeven kontekst. Dersom synonymet ikkje er rett eller synonymet ikkje fins, skal produkteigar også kunne leggje til nye synonym. Desse sjølv laga synonyma skal ein kunne gje ei forklaring, og om ønskeleg relatere til andre kategoriar i *WordNet*.

Effekten av *Semantizer* har blitt testa eksperimentelt. Formålet med eksperimentet var å undersøkje om bruken av *Semantizer* leia til auka domeneforståing. Resultatet av eksperimentet var kvantitative data som vart evaluert etter standard statistiske metodar. I tillegg vart tre av eksperiment-deltakarane intervjuja med tanke på å få innspel til å forstå eksperimentresultata og vidare potensiale for *Semantizer*.

1.2 Forskingsspørsmål

Basert på måla som er definert over, er forskingsspørsmålet i dette prosjektet følgjande:

FS 1: Kan bruk av den semantiske teknologien *WordNet* bidra til auka domeneforståing for utviklarlaget i eit smidig utviklingsprosjekt?

¹<http://wordnet.princeton.edu/>

Sidan dette prosjektet søker å få eit svar på forskingsspørsmålet eksperimentelt, vart standard hypotesetestingsmetodikk nytta og ein formulerer difor ei null-hypotese og ei alternativ hypotese:

H_0 : Bruk av den semantiske teknologien **WordNet** bidreg ikkje til å auke domeneforståinga for utviklarlaget i eit smidig utviklingsprosjekt.

H_1 : Bruk av den semantiske teknologien **WordNet** bidreg til å auke domeneforståinga for utviklarlaget i eit smidig utviklingsprosjekt.

1.3 Oppgåvestruktur

Kapittel 1 er ein introduksjon til masteroppgåva og forskingsspørsmåla. Kapittel 2 presenterer relevant teori omkring smidig metodikk, Scrum, naturleg språk-krav, ontologi og **WordNet**. Kapittel 3 presenterer problemstillinga og kva forskingsmetodar som er nytta med fokus på designvitskap og eksperimentell metode. Kapittel 4 er ein presentasjon av **Semantizer**-applikasjonen. Kapittel 5 skildrar korleis forskingsdesignet er i dette prosjektet, og korleis eksperimentdesignet er sett opp. Kapittel 6 er ein presentasjon av resultat og diskusjon omkring resultatet. Kapittel 7 konkluderer funna i prosjektet og vidare arbeid.

Kapittel 2

Teori

I dette kapitlet er smidig metodikk skildra, med spesielt fokus på Scrum. Vidare vert rollene i Scrum presentert, deretter nokre viktige artefaktar i Scrum som brukarhistorier og Scrum-veggen og nokre viktige komponentar for samarbeid i utviklarlaget. Avslutningsvis vert naturleg språk-krav og ontologiar skildra.

2.1 Smidig metodikk

Dei siste 15 åra har det utvikla seg ei sterk rørsle innan programutvikling under namnet smidig utvikling (“agile software development”). Dingsøy, Dybå og Moe (2010) hevdar at smidig utvikling har hatt stor påverknad på korleis programutvikling har endra seg frå tradisjonelle utviklingsmetodologiar i denne perioden. Dei smidige ideane har påverka små, mellomstore og store bedrifter frå blant anna telekommunikasjon og helsesektor til spelindustrien. Dingsøy mfl. (2010) utdjupar at smidig metodikk har endra korleis utviklarlaget planlegg, kommuniserer og koordinerer arbeidet sitt. Smidig utvikling blir sett på som avløysaren til dei tradisjonelle metodane som vektlegg ei form for rasjonell og ingeniør-basert tilnærming. Dei tradisjonelle metodane legg stor vekt på omfattande planlegging, kodifisert prosess og streng gjenbruk av kode. Blant fleire er fossefallsmodellen ein av dei mest omtalte metodane innan tradisjonell systemutvikling.

Smidige metodar godtek at verda ikkje er føreseieleg og fokuserer heller på verdien kompetente menneske og deira relasjonar fører meg seg i programutvikling. Nokon av dei mest kjende smidige metodikkane er Adaptive software development (ASD), Crystal family,

Extreme programming (XP), Pragmatic programming og Scrum. Innan dei vitskaplege miljøa er XP og Scrum mykje forska på, og desse har også fått stor merksemd i industrien.

Ifølgje Dingsøy mfl. (2010) så dreiar smidig programutvikling seg om “tilbakemelding og endring”, og presiserer at programutvikling er ein empirisk og ikkje-lineær prosess. Tilbakemeldingar mellom utviklarlaget og kunde skal skje hyppig, slik at ein kan utvikle eit produkt som er føreseieleg og som oppfyller kunden sine krav. At ein metodologi er smidig forklarar Dingsøy mfl. (2010) slik:

“strip away as much of the heaviness, commonly associated with the traditional software-development methodologies, as possible to promote quick response to changing environments, changes in user requirements, accelerated project dead-lines and the like” (Dingsøy mfl., 2010, s. 2).

Det vert argumentert for at ein smidig metodikk må medverke til minst eitt av dei neste punkta. Desse er å

- skape endring
- reagere i forkant av endring
- reagere ved endring
- lære av endring

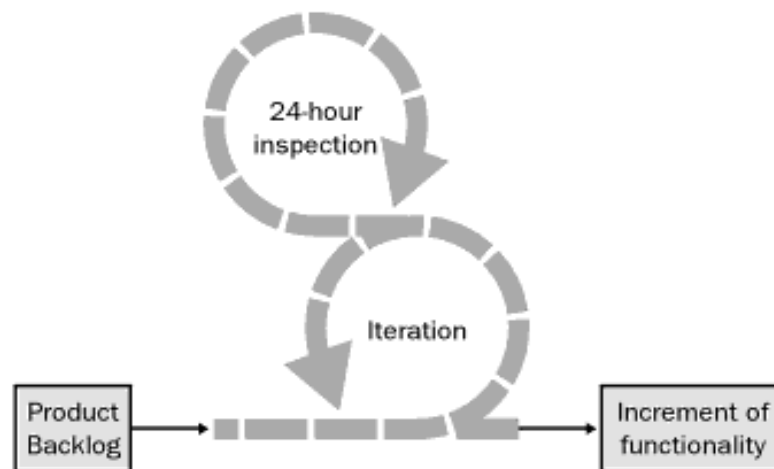
Vidare så må smidige metodikkar bidra til og ikkje avleie merksemd frå økonomi, kvalitet eller enkelheit. Det siste kravet er å vere klar til og sette ein komponent i bruk. Smidige metodikkar skil seg frå tradisjonelle metodikkar ved at ein fokuserer på å vere smidig og tilpasse seg forandring, og i tillegg levere produkt av høg kvalitet med ein enkel arbeidsprosess (Dingsøy mfl., 2010).

2.2 Scrum

Scrum er kanskje den mest populære av dei smidige utviklingsmetodane. Alle aktivitetar som skjer i Scrum baserer seg på eit iterativt og inkrementelt prosess-skjelett, som vist

i figur 2.1. Den nedste sirkelen representerer ein iterasjon som skjer etter kvarandre der resultatet er eit inkrement av eit produkt. Den øvste sirkelen syner det daglege inkrementet som oppstår i ein iterasjon, kor utviklarlaget møtes og tar del i ulike aktivitetar. Desse aktivitetane er blant anna å validere andre medlemmar av utviklarlaget sitt arbeid og tilpasse seg på dei områda det trengst. Det som driv ein iterasjon er ei liste med krav, gjerne kjent som brukarhistorier (“user stories”). Når det ikkje er fleire brukarhistorier å implementere eller prosjektet ikkje lenger vert finansiert, er prosjektet over (Schwaber, 2004).

Utviklarlaget byrjar med å sjå over kva brukarhistorier som må utviklast, og vel ut eit gjeve antal brukarhistorier som laget skal utvikle til eit produkt. Laget vurderer sine egne evner, tilgjengeleg teknologi og til sist korleis produktet skal lagast. Problem som laget støytar på undervegs vert diskutert kollektivt ved behov. Deretter skal det sjølvorganiserande laget jobbe med å produsere kode resten av iterasjonen. Mot slutten av iterasjonen presenterer laget produktet som er utvikla så langt, slik at kunden kan inspisere funksjonar som er utvikla. Kunden sine ønskjer om endring kan også formidlast til utviklarlaget i denne presentasjonen, slik at desse ønskja kan bli implementert i det ferdige produktet (Schwaber, 2004).



FIGUR 2.1: Scrum-skjelett, som vist i Schwaber, 2004, s. 6

2.2.1 Scrum-master og utviklarlaget

I Scrum finst det tre roller: Scrum-master, laget og kunden. Scrum-masteren er ansvarleg for Scrum-prosessen. Dette inneber å lære laget Scrum og implementere Scrum som ein del av organisasjonen sin kultur. Samstundes må Scrum-masteren også sikre at laget leverer forventa kvalitet og at laget følgjer reglar og praksisar.

Utviklarlaget, eller laget (“teamet”) er ansvarlege for å utvikle dei nødvendige funksjonane som produktet skal tilby. Laget skal vere sjølvorganiserande, sjølvleiande og kryssfunksjonelle. Oppgåva til utviklarlaget er å finne ut korleis produktloggen kan bli til funksjonar i eit informasjonssystem og samtidig leie dette arbeidet (Schwaber, 2004).

2.2.2 Produkteigar

Produkteigar (“product owner”) er ansvarleg for å representere kunden og sluttbrukarane sine interesser i prosjektet. Det er produkteigar som lagar krava til produktloggen og som finansierer prosjektet. For å sikre at dei mest verdifulle krava vert utvikla først, er det også produkteigar sitt ansvar å prioritere produktloggen. Dette blir gjort ved å stadig prioritere lista slik at dei mest verdifulle krava vert utvikla på neste sprint (Schwaber, 2004).

Produkteigar er ein viktig ressurs som utviklarlaget er avhengig av å ha kontakt med. I Scrum er det éin eller fleire personar som tek på seg rolla som produkteigar, og produkteigar skal vere på same lokalitet som utviklarlaget. Dersom utviklarlaget ikkje er konsulentar som reiser ut til kunden, så må kunden kome til utviklarlaget. Dette er fordi ein ønskjer å løyse problem som oppstår så raskt som mogleg. Aurum og Wohlin (2005) argumenterer for at produkteigar skal vere tilgjengeleg, ha fullstendig kjennskap til problemdomenet, og ha makt til å ta avgjersler. Dersom produkteigar manglar nokon av desse eigenskapane, vil dette hindre utviklinga av produktet. Vidare seier Aurum og Wohlin (2005) at nærvær til produkteigar er svært nødvendig sidan dokumentasjon er redusert til eit minimum, og at laget spør produkteigar dersom noko er uklart. Ei misforståing mellom laget og produkteigar kan føre til at det vert utvikla noko unødvendig som produkteigar og dei interesserte må betale for. Med ein produkteigar hos utviklarlaget så kan ein stille spørsmål og diskutere dersom noko er usikkert (Martin, Biddle & Noble, 2010).

I dei tilfella kor det ikkje er mogleg å ha ein produkteigar på arbeidsplassen til utviklarlaget, så kan dette bli ei barriere for utviklarlaget. Tida det tek å kome i kontakt med produkteigar er tid som i utviklingsperspektiv er bortkasta og som burde gått med til utvikling av produktet i staden for. Korleis ein skal takle ei slik barriere er opp til kvart utviklingslag, men ein kan nytte seg av videosamtalar via til dømes Lync og Skype. Dette er ein god måte å kommunisere på over store geografiske avstandar, men ein er like langt dersom kunden ikkje har tid til å prioritere rolla som produkteigar.

I eit studie av Aurum og Wohlin (2005) vart det undersøkt kva faktorar som leiar til at eit systemutviklingsprosjekt feilar. Som synt i tabell 2.1 ser ein at fem av åtte faktorar som fører til at eit prosjekt feilar er på grunn av krav. Desse er nummer 1, 2, 4, 6 og 8. Den største feilkjelda er ufullstendige krav (13,1%). Ufullstendige krav er utdjupa i seksjon 2.3.1. Den nest største feilkjelda er låg deltaking frå kunden si side (12,4%). Dette syner kor viktig det er i alle systemutviklingsprosjekt å definere krav og involvere produkteigar.

No	Problem	%
1	Incomplete requirements	13.1
2	Low customer involvement	12.4
3	Lack of resources	10.6
4	Unrealistic expectations	9.9
5	Lack of management support	9.3
6	Changes in the requirements	8.7
7	Lack of planning	8.1
8	Useless requirements	7.5

TABELL 2.1: Grunnar til at prosjekt feilar, som vist i Aurum og Wohlin, 2005, s. 315

Både i Scrum og Extreme Programming (XP) er det viktig at ein produkteigar er til stade med utviklarlaget. Ifølgje Wake (2001) er det viktig at produkteigaren er hos utviklarlaget for at utviklinga av produktet skal skje raskt. Dersom produkteigar får eit spørsmål frå utviklarlaget skal produkteigar gje eit svar med ein gong, eller finne eit svar i løpet av kort tid. Mange av desse spørsmåla baserer seg på å ta avgjersler for vidare utvikling av produktet, og desse avgjerslene er difor viktige for funksjonar og design til det endelege produktet. I tillegg er det viktig at ein produkteigar er til stade og bestemmer kva som

skal gjerast og prioriterer brukarhistorier (Wake, 2001). Farell, Narang, Kapitan og Webber (2002) seier at realiteten for mange smidige utviklingsprosjekt er at ein produkteigar ikkje er tilgjengeleg eller ikkje villeg til å delta i systemutviklingsprosessen. Dette er eit viktig prinsipp i smidig utvikling, som punkt nummer to i tabell 2.1 understrekar. Å involvere kunden er svært viktig for å lykkast med prosjektet og produktet ein utviklar.

2.2.3 Scrum-artefaktar: brukarhistorier og Scrum-vegg

Brucarhistorier er ei konsis skildring av eit krav frå produkteigar, som dreiar seg om kva funksjonar programmet skal ha. Eit døme på ei brukarhistorie er “As a student I would like to search for a book title in a library database”. Produkteigar lagar brukarhistoriene som blir plassert på ein Scrum-vegg. Scrum-veggen er delt i fem kategoriar, desse er “produktlogg”, “sprintlogg”, “utvikling”, “testing” og “ferdig”. Det er produkteigar si oppgåve å prioritere desse brukarhistoriene. For å finne ut kor krevjande det er å gjennomføre ei brukarhistorie så estimerer ein alle brukarhistoriene individuelt. Dette er ein prosess som utviklarlaget gjer i byrjinga av ein sprint. Til dømes planleggingspoker (“planning poker”) kan nyttast til dette formålet, utan at det blir nærare forklart her. Når prosessen er over vil ein ha eit estimat på kor krevjande kvar brukarhistorie er (Haugen, 2006).

Alle brukarhistoriene er i utgangspunktet plassert i produktloggen, men når ein planlegg neste sprint i eit planleggingsmøte (“planning meeting”) vel utviklarlaget dei høgst prioriterte brukarhistoriene som vert flytta til sprintloggen. Basert på estimering av brukarhistorier frå tidlegare prosjekt, har utviklarlaget tal som syner kor mange estimerte poeng som er reelt at laget kan fullføre i løpet av ein sprint. Til dømes kan eit Scrum-utviklarlag fullføre 30 estimerte poeng i gjennomsnitt kvar veke. Difor kan utviklarlaget leggje til brukarhistorier verdt 30 estimerte poeng til sprintloggen, og deretter vel kvar enkelt utviklar ei brukarhistorie. Desse valde brukarhistoriene blir flytta til neste kategori som er utvikling, markert med namnet til den som jobbar med denne. Vanlegvis arbeider éin utviklar med éi brukarhistorie om gongen, men dersom ein nyttar seg av parprogrammering kan det vere aktuelt å setje begge utviklarane på denne brukarhistoria.

Når utviklar meiner at utviklinga av brukarhistoria er ferdig, vert brukarhistoria flytta til neste kategori som er testing. Kven som tek på seg ansvaret for å drive testing på ei brukarhistorie er ikkje nødvendigvis den same som utvikla brukarhistoria. Dette er eit av prinsippa ved å vere smidig, at ein har som mål at alle på utviklarlaget skal ha

kompetanse til å jobbe vidare med ei brukarhistorie. I testfasen kan ein avdekkje feil eller manglar ved produktet ein har utvikla, noko som vil føre til at ein må setje brukarhistoria tilbake til kategorien utvikling. Her legg testaren gjerne ved ein liten kommentar om kvifor brukarhistoria ikkje kunne markerast som ferdig, og så må feilen rettast av neste ledige utviklar. Om alt er i orden og produktet som er utvikla oppfyller brukarhistoria, så kan brukarhistoria flyttast til siste kategori som er ferdig. Utviklaren kan deretter byrje på neste brukarhistorie (Sutherland & Schwaber, 2010).

2.2.4 Viktige komponentar i utviklarlaget

Ifølgje Moe og Dingsøy (2008) så er det fem viktige komponentar som er nødvendige for at samarbeidet i utviklarlaget skal fungere optimalt. Desse er lagleiarskap (“team leadership”), gjensidig resultatoppfølging (“mutual performance monitoring”), backup av arbeidsoppgåver (“backup behaviour”), tilpassingsevne (“adaptability”) og lagfokus (“team orientation”). Desse fem er avhengige av tre koordinerande mekanismar: delte mentale modellar (“shared mental models”), kontinuerleg kommunikasjon (“closed-loop communication”) og gjensidig tillit (“mutual trust”). Dersom ein av komponentane eller koordinerande mekanismane feilar, så vil dette gå ut over laget si evne til å samarbeide. Laget skal vere sjølvorganisert, det vil seie at det er laget beståande av produkteigar, utviklarar og SCRUM-master som saman skal planlegge og ta avgjersler. Det skal vere mogleg å sjå og gje tilbakemeldingar på eige og andre sitt arbeid, noko som er mogleg gjennom produktloggen. Backup av arbeidsoppgåver og tilpassingsevne dreiar seg om at alle på laget skal kunne arbeide med kvarandre sitt arbeid og hjelpe til med arbeidsoppgåver når det til dømes er veldig mykje arbeid. Med andre ord skal ein arbeide for at alle utviklarar kan gjere dei oppgåvene som må gjerast, slik at arbeidet ikkje stansar fordi ein utviklar av ein grunn ikkje kan gjere ei oppgåve. Laget si orientering er også ein viktig komponent, og inneber i Scrum at ein set mål for laget slik at ein blant anna jobbar for eit felles mål og at ein vel oppgåver som passar utviklar sin kompetanse og ønskjer.

Delte mentale modellar er nødvendig for å stadig orientere laget mot målet, nemleg å lage eit produkt som kunden ønskjer. Dette gjer ein i Scrum ved å arrangere møter, til dels daglege Scrum-møter (“daily Scrum”), planleggingsmøter (“planning meeting”) og liknande aktivitetar kor produkteigar, Scrum-master og utviklarar er involvert. Kontinuerleg kommunikasjon dreiar seg om at det er kommunikasjon i laget, både på dagleg

basis og når ein sprint eller eit prosjekt er fullført. I Scrum så har ein eit oppsummerande møte (“review meeting”) etter kvar sprint kor ein kan få og gje tilbakemelding til laget. Gjensidig tillit er heilt nødvendig for at laget skal fungere og stole på kvarandre. Det handlar om å stole på at ein annan utviklar kan produsere den nødvendige koden til ein sett frist (Moe & Dingsøy, 2008).

2.3 Naturleg språk-krav

Naturleg språk er eit viktig verkemiddel når ein skal definere krav for eit system ein ønskjer å utvikle. Ambriola og Gervasi (1997) forklarar at naturleg språk er noko alle i eit systemutviklingsprosjekt har kjennskap til, og difor er det naturleg språk som blir nytta til å definere krav. I nokre situasjonar har produkteigar kjennskap til eit meir formelt språk for å definere krav, men i desse tilfella er dette språket gjerne ikkje kjent for personen som skal analysere krava. Difor er i stor grad naturleg språk dominerande i utforminga av krav. Dette har fleire fordelar, blant anna at dei aller fleste produkteigarar kan definere sine krav og ønskjer, og utviklarar og designarar kan forstå og kome med tilbakemelding på desse krava. Dette er spesielt viktig med tanke på brukarhistorier som er ein viktig Scrum-artefakt. Desse vert oppretta av ein produkteigar og nytta av utviklarlaget. Gervasi (2005) argumenterer at produkteigarar brukar naturleg språk når dei føreslår, diskuterer og lagar nye krav. Av alle kravdokument som finst hevdar Gervasi (2005) at 95% av alle kravdokument er skriva på naturleg språk. Ein kan difor seie at naturleg språk er noko som er felles for alle produkteigarar, og med naturleg språk så minkar ein terskelen for deltaking i systemutvikling. Med krav definert med naturleg språk må ein likevel vere oppmerksom på at det er eit språk med visse svakheiter.

2.3.1 Uregelmessige krav

Uregelmessig framstilling av krav beskriv Gervasi (2005) som eit av dei vanlegaste feila som oppstår ved naturleg språk-krav. Dersom eit krav fører til konflikt med andre krav, motseier korleis systemet skal oppføre seg eller forandrar domenet, så er dette kravet uregelmessig. Vidare meiner Gervasi (2005) at dersom ein skal oppretthalde ein konsistent kravspesifikasjon, er det nødvendig at ord og termar beheld same meining i heile

kravspesifikasjonen. Dette betyr at ein må unngå å bruke tilsynelatande like termar og ord som har forskjellig meining.

Det finst mange grunnar til at uregelmessige krav oppstår i eit systemutviklingsprosjekt. Zowghi, Gervasi og McRae (2001) forklarar at dette kan oppstå fordi eit prosjekt har fleire produkteigarar som har ulikt syn på både systemutviklingsprosessen og målsettinga til det ferdige produktet. I tillegg er også varierende språkbruk, utviklingsstrategiar, syn og grad av overlapping i dei områda produkteigarane er interesserte i faktorar som spelar inn. Gervasi (2005) legg til at dersom utviklinga av eksisterande eller nye krav ikkje er riktig koordinert av produkteigar, endar ein opp med eit krav som ikkje samsvarar med det produktet som var planlagt.

2.3.2 Fullstendige krav

Eit anna viktig attributt som ifølgje Gervasi (2005) også er blitt forska på i ei rekke studiar, er fullstendigheit. At ein komponent er fullstendig er avgjerande når ein utviklar programvare, både for sikkerheitskritiske informasjonssystem og for delkomponentar av eit større informasjonssystem. Gervasi (2005) hevdar at ein av grunnane til at maskinrelaterte feil og ulykker oppstår er fordi krav eller kravspesifikasjonen er ufullstendig. Vidare argumenterer han for at ufullstendige krav er nokon av det vanskelegaste feila å oppdage i kravspesifikasjonar.

For å oppnå fullstendige krav meiner Gervasi (2005) at krava må oppfylle tre fundamentale karakteristikkar. Den første er at “ingen informasjon er uavklart og ingen informasjon ventar avklaring”. Det andre punktet er at “informasjonen ikkje inneheld uidentifiserte objekt eller entitetar”. Det siste punktet er at “kravspesifikasjonen manglar ingen informasjon”.

2.4 Ontologi

Termen ontologi stammar ifølgje Studer, Benjamins og Fensel (1998) frå filosofien kor Aristoteles forsøkte å klassifisere ting i verda og kva desse vart nytta til. Forfattarane brukar denne definisjonen om ontologi: “An ontology is a formal, explicit specification of a shared conceptualisation”. Ei konseptualisering (“conceptualisation”) referer til ein

abstrakt modell som beskriv eit fenomen i verda ved å identifisere dei relevante konsepta som finst i dette fenomenet. Eksplisitt (“explicit”) betyr at typen konsept som er nytta, og avgrensingane på desse konsepta, er eksplisitt definert. Formelt (“formal”) betyr at ontologien skal vere maskinleseleg som ekskluderer naturleg språk. At ein ontologi er delt (“shared”) reflekter eit viktig standpunkt med ontologiar, nemleg at det er samtykke frå ei gruppe om kunnskapen som finst i ontologien.

Ifølgje Studer mfl. (1998) er motivasjonen for å nytte ontologiar at ein kan dele ei felles forståing av eit domene med andre menneske og datamaskiner, men den aller viktigaste motivasjonsfaktoren er deling og gjenbruk av ein eksisterande kunnskapskropp i eit format som maskiner forstår.

Studer mfl. (1998) hevdar at dei aller fleste ontologiar i dag forsøker å modellere statistisk domenekunnskap, som til dels domene modellar innan medisin, elektriske komponentar eller mekaniske maskiner. Med ein ontologi forsøker ein å strukturere og lagre universell akseptert kunnskap, uavhengig av korleis ontologien skal nyttast. Ulike typar forskning har forskjellige syn på korleis ein skal representere og nytte ontologiar, og eit av desse syna er korleis ein kan lage og bruke informasjonen i ontologiar basert på domenekunnskap.

Innan eit gjeve domene så har ein ontologi, i tillegg til å vere ein representasjon av domenet, konsens om at kunnskapen i ontologien er gyldig. Kva personar, organisasjonar eller offentlege institusjonar som samtykkjer til kunnskapen i ein ontologi er ifølgje Studer mfl. (1998) avhengig av kontekst. For eksempel vil eit forskingsprosjekt på eit sjukehus som byggjer ein kunnskapsontologi for ein gjeven sjukdom ha konsens på dette sjukehuset, og kunnskapen vil difor vere gyldig innanfor dette forskingsprosjektet og sjukehuset. I ein offentlig utvikla ontologi til dels for søk i eit biblioteksystem, så er nasjonal konsens nødvendig, som betyr at alle samtykkjer til at ontologien er brukande.

Studer mfl. (1998) hevdar at ontologiar kan fremje kommunikasjon mellom menneske i ein organisasjon. Dette er fordi ontologiar består av termar, forklaringar, relasjonar og avgrensingar, og desse definisjonane bør alle medlemmane i organisasjonen akseptere. Denne kunnskapen skal vere tilgjengeleg for dei rette medlemmane og haldast ved like av organisasjonen.

Det blir nemnt at ontologiar kan klassifiserast i fire nivå av generalisering:

- Domeneontologiar

- Generiske ontologiar
- Applikasjonsontologiar
- Representasjonsontologiar

Domeneontologiar har kunnskap som er gyldig for eit spesifikt domene. Desse kan for eksempel vere domene innan medisin, elektronikk eller mekaniske motorar. Generiske ontologiar er gyldige over fleire domene. Desse ontologiane blir også referert til som superteoriar eller kjerne-ontologiar (“super theories”, “core ontologies”). Applikasjonsontologiar består av nødvendig kunnskap for å modellere eit spesifikt domene. Til sist er representasjonsontologiar som ikkje forpliktar seg til eit spesifikt domene. Desse representerer entitetar som ikkje spesifikt seier kva som skal bli representert.

2.4.1 WordNet

I denne seksjonen vert **WordNet** nærare forklart, som er ein viktig ontologi nytta i dette prosjektet.

Ei gruppe psykologar og lingvistar ved Princeton University bestemte seg for å utvikle ein leksikalsk database ved å følge nokre teoriar som stammar frå psykolingvistikken. Motivasjonen for utviklinga av denne leksikalske databasen var å lage eit verkty kor ein kunne søkje konseptuelt, ikkje berre alfabetisk som i ei tradisjonell ordliste. I utgangspunktet skulle denne leksikalske databasen nyttast i kombinasjon med ei tradisjonell ordliste på veven, men undervegs i utviklinga vart det klart for forskarane at ein måtte definere nye mål med **WordNet**. Etersom **WordNet**, den leksikalske databasen, bygger på hypoteser basert på psykolingvistisk forskning, så kan **WordNet** seiast å vere ei ordbok basert på psykolingvistiske prinsipp (Miller, Beckwith, Fellbaum, Gross & Miller, 1990).

Den mest openbare forskjellen mellom **WordNet** og ei standard ordbok er ifølgje Miller mfl. (1990) at **WordNet** deler leksikonet inn i fem kategoriar. Desse er substantiv, verb, adjektiv, adverb og funksjonelle ord. I **WordNet** er alle kategoriane bortsett frå funksjonelle ord inkludert. Miller mfl. (1990) definerer **WordNet** slik:

WordNet is an on-line lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns,

verbs, and adjectives are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets.

Ved å innføre kategoriar basert på syntaks i *WordNet*, vart problemet med redundans introdusert. Dette førte til at for eksempel ordet “back” er med i meir enn ein kategori. Fordelen med kategoriar er at forskjellane i den semantiske organiseringa av desse kategoriane vert tydeleg framheva og kan bli systematisk nytta. I dette prosjektet er spesielt substantiv nytta i *SynsetTagger*-applikasjonen, jamfør seksjon 4.3.

I følge Miller mfl. (1990) så er den mest ambisiøse funksjonen i *WordNet* at ein forsøker å organisere leksikalsk informasjon basert på meininga til ord. Dette er ein klar kontrast til å sortere ord etter ordform, slik tradisjonelle ordlister fungerer. På grunn av dette liknar *WordNet* meir på ei synonymordbok (“thesaurus”) enn ei ordliste.

Ordet “word” er tvtydig og kan bli brukt for å forklare representasjonen til eit ord eller den underliggjande semantikken. Difor kan ordform nyttast for å forklare representasjonen til eit ord, medan meininga til eit ord er forklart som den tingen eller ideen ordet representerer. I tabell 2.2 er M_y meiningane eit ord kan ha, medan F_x er ordformene til dette ordet. Tabellen syner at ei ordform kan ha ei eller fleire meiningar, og ei meining kan vere gyldig for ei eller fleire ordformer. Eit synonym er presentert ved $E_{(1,1)}$ og $E_{(1,2)}$, og betyr at meininga til eit ord M_1 kan forklarast med to synonym gitt ved ordformene F_1 og F_2 . Eitt er polysem gitt ved $E_{(1,2)}$ og $E_{(2,2)}$, og vil sei at ei ordform har to meiningar.

Word Meanings	Word Forms			
	F_1	F_2	F_3	F_n
M_1	$E_{1,1}$	$E_{1,2}$		
M_2		$E_{2,2}$		
M_3			$E_{3,3}$	
.				.
.				.
.				.
M_m				$E_{m,n}$

TABELL 2.2: Leksikalsk matrise som syner relasjonane mellom ordform og meiningane til eit ord

For å organisere orda i ein kategori nyttar ein meininga til ord i staden for ordforma. Desse orda vert samla i sett av synonym, og er kalla synset. Eit søk med WordNet

Search - 3.1¹ på ordet “user”, returnerer eit synset med tre synonym. Settet av synonym er markert ved å nytte krøllparentesar: {user, exploiter, drug user}. Når ein referer til eit spesifikt synset, for eksempel første synonym i synsettet returnert for “user”, skriv ein dette på forma n#1 user. “n” tyder “noun” (substantiv), #1 indikerer posisjonen i synsettet og til sist kjem synonymet.

Ifølgje Miller mfl. (1990) kan ein definere synonym som “[...] two expressions are synonymous if the substitution of one for the other never changes the truth value of a sentence in which the substitution is made” (Miller mfl., 1990, s. 6). Med denne definisjonen er ekte synonym sjeldne, og ein mildare definisjon er difor foreslått:

A weakened version of this definition would make synonymy relative to a context: two expressions are synonymous in a linguistic context C if the substitution of one for the other in C does not alter the truth value (Miller mfl., 1990, s. 6).

Med denne definisjonen er det nødvendig å klassifisere ord anten som substantiv, verb, adjektiv og adverb. Sidan ord skal vere utskiftbare betyr det at ord i forskjellige kategoriar ikkje kan vere synonym og kan difor heller ikkje skape synset. Bruken av synset grunnar i psykolingvistikken som hevdar at substantiv, adjektiv, verb og adverb er uavhengig organisert, og at ein difor lett kan skilje mellom ord som er synonym og ord som ikkje er synonym.

Hypernym og hyponym

Ifølgje Miller mfl. (1990) så skil hypernym og hyponym seg frå synonym ved å vere ein semantisk relasjon mellom meiningane til ord. For eksempel så er {user} eit hyponym av {person}, og {person} eit hyponym av {being}. Eit konsept representert som synsettet {x, x', ...} er hyponymet til eit konsept representert som synsettet {y, y', ...}, dersom det er aksept blant engelsktalende at det finst ein relasjon *An x is a (kind of) y*.

Eit hyponym er transitivt og asymmetrisk, og sidan det vanlegvis er ein enkelt overordna term så dannar dette ein semantisk hierarkisk struktur kor ein hevdar at hyponymet er plassert under eit overordna hypernym. Denne typen hierarkisk struktur er mykje nytta i system som nyttar arv, kor eit hyponym arvar alle funksjonar frå eit eller fleire overordna

¹<http://wordnetweb.princeton.edu/perl/webwn>

hypernym. For eksempel vil {user} arve eigenskapar frå {person}, og {person} vil arve eigenskapar frå {being}.

Ein annan måte å skrive dette leksikalske treet på er $user @ \rightarrow person @ \rightarrow being$, kor “@ \rightarrow ” er den transitive, asymmetriske semantiske relasjonen “is a” eller “is a kind of”. Dette designet skapar eit hierarki som strekk seg frå mange spesifikke termar på lågare nivå til nokre få generiske termar på toppen (Miller mfl., 1990).

Kapittel 3

Problemstilling og metode

I denne seksjonen vert først problemstillinga diskutert, saman med motivasjonen for prosjektet. I dette prosjektet vart designvitskap nytta, som er skildra nærare i seksjon [3.2](#). Til sist er eksperimentell design forklart, som vart nytta i evalueringa av prosjektet, jamfør seksjon [3.3](#).

3.1 Problemstilling og motivasjon

I smidig systemutvikling er fokuset å produsere og levere eit produkt til kunden, og undervegs vere tilpassingsdyktig slik at ein kan takle endringar på ein god måte. Desse endringane kan for eksempel vere nye krav definert av produkteigaren eller ny teknologi som mogleggjer utvikling av nye produkt raskare og meir effektivt. Brukarhistorier er kravspesifikasjonen for smidige utviklingsmetodar som Scrum, og desse nyttar eit naturleg språk. Dette er positivt i den forstand at ein som produkteigar kan nytte naturleg språk for å gje brukarhistoriene mening. Deretter er det opp til utviklarlaget å føre desse brukarhistoriene frå tekst til programvare.

Eit problem med naturleg språk er at dette ikkje er eintydig, som kan skape problem for utviklarlaget. Desse problema kan løysast ved samtale med produkteigar, som forklarar kva som er meint og kva som er ønskja resultat av ei eller fleire brukarhistorier. Dette krev at produkteigar er til stades på arbeidsplassen eller tilgjengeleg for utviklarlaget på andre måtar, noko ikkje alle produkteigarar har tid eller moglegheit til. Dette kan vere fordi eksisterande arbeidsoppgåver er viktige slik at ein prioriterer desse høgare enn rolla

som produkteigar. Ein annan grunn kan vere at det i ein pressa økonomisk situasjon ikkje er forsvarleg for bedrifta å la denne personen arbeide som produkteigar. Det kan føre til at utviklarlaget må ta avgjersler for å halde tidsfristar, som til slutt påverkar produktet i negativ forstand. Dette kan i verste fall føre til at produktet ikkje er slik produkteigar hadde ønska.

Ein grunn til at produktet ikkje vert slik produkteigaren ønsker, kan vere på grunn av brukarhistorier som ikkje skapar noko verdi i sluttproduktet. Desse brukarhistoriene kan vere uregelmessige krav, som skildra i seksjon 2.3.1. Dette kan leie til at utviklarlaget implementerer noko som er feil eller unødvendig. Når dette problemet blir oppdaga, kan det ifølgje Gervasi (2005) vere svært dyrt å rette opp i denne feilen sidan produktet allereie kan vere i produksjon. For å unngå at utviklarlaget implementerer desse brukarhistoriene, er det viktig at utviklarlaget forstår domenet dei utviklar for, og får tilgang på informasjon som er nødvendig for å utvikle produktet.

I dette prosjektet er målet at *Semantizer* skal kunne betre produkteigar sine moglegheiter til å definere brukarhistorier, slik at desse vert mest mogleg eintydige. Spesielt fagtermar som er ukjende for utviklarlaget skal produkteigar kunne gje ei forklaring, som skal gjere det enklare for utviklarlaget å forstå brukarhistoria og domenet. For å finne synonym eller meiningar eit ord har, vert semantiske teknologiar teke i bruk. Ved å trekkje ut informasjon frå brukarhistoriene, kan ein identifisere både ord og kontekst ved å nytte semantisk teknologi. Identifikasjonen kan skje anten manuelt ved at eit menneske utfører denne oppgåva, eller automatisk. Eit mål med *Semantizer* er at dette skal skje automatisk, ved å nytte semantiske teknologiar tilgjengeleg via opne lisensar.

I tillegg er det ønskeleg å modellere domenet for brukarhistoriene, for eksempel i ein tre-representasjon. Dette kan kanskje gje produkteigar og utviklarlaget eit oversyn over brukarhistoriene, og også opne moglegheita for å utforske termar og relasjonane til termarne i den semantiske veven. Både identifisering av ord og kontekst, i tillegg til modellering av brukarhistorier, vil vere viktige faktorar for å auke domeneforståinga til eit utviklarlag.

Neste seksjon presenterer metodane som er nytta i prosjektet. Desse er designvitskap og eksperimentell design, sistnemnte vart nytta i eit evalueringsperspektiv.

3.2 Designvitskap

Hevner, March, Park og Ram (2004) skildrar to ulike paradigme for forskning i informasjonssystem. Desse to er åtferdsvitskap (“behavioral science”) og designvitskap (“design science”). Designvitskap kjem opphavleg frå ingeniørfaga og er ifølgje forfattarane eit paradigme for problemløysing. Karakteristikkar ved dette paradigmet er at ein ønskjer å drive idéutvikling som er med på å skape idear, praksisar, tekniske evner og produkt ved hjelp av analyse, design, implementering, handtering og effektiv bruk av informasjonssystem. Hevner og Chatterjee (2010) definerer designvitskapleg forskning (“Design science research”) som eit forskingsparadigme kor ein forskar søkjer å svare på relevante problemstillingar som menneske har. Dette gjer forskaren ved å skape ein artefakt som bidreg til å styrke vitskapen med ny kunnskap. Denne artefakten er både nyttig og avgjerande for forståinga av problemstillinga. Hevner mfl. (2004) presenterer sju retningsliner for designvitskapleg forskning.

Retningslinjer for designvitenskap
1. <i>Design i form av ein artefakt</i>
Designvitenskapleg forskning må produsere ein artefakt i form av ein konstruksjon, modell, metode eller ved eksemplifisering.
2. <i>Problemrelevanse</i>
Målet med designvitenskapleg forskning er å utvikle teknologiske løysingar til problem som både er viktige og relevante i forretnings samband.
3. <i>Evaluering av design</i>
Nytten, kvaliteten og effekten av ein artefakt må bli demonstrert ved hjelp av velprøvde evalueringsmetodar.
4. <i>Forskningsbidrag</i>
Effektiv designvitenskapleg forskning må vise til og verifisere klare bidrag til eit eller fleire områder, i høve til designfasen, designfundamentet og/eller designmetodologiar.
5. <i>Forskningsstyrke</i>
Designvitenskapleg forskning byggjer på eit fundament av strenge metodar for konstruksjon og evaluering av artefakten som vert utvikla.
6. <i>Design som ein søkeprosess</i>
Søket etter ein effektiv artefakt krev at ein nyttar seg av tilgjengelege verkty for å nå eit ønskja mål samtidig som ein respekterer dei lovene og reglane som finst i problemdomenet.
7. <i>Kommunikasjon av forskning</i>
Designvitenskapleg forskning skal presenterast effektivt både til teknologi- og forretningsretta publikum.

TABELL 3.1: Retningslinjer for designvitenskap

I det følgjande avsnittet er det presentert korleis prosjektet følgjer retningslinjene for designvitenskap. Første punkt i retningslinjene for designvitenskap er støtta ved å utvikle ein artefakt i form av ein Java-applikasjon. Denne applikasjonen er kalla *Semantizer*

og nyttar den semantiske teknologien *WordNet* for å undersøkje om denne semantiske teknologien leiar til auka domenekunnskap blant utviklarlaget, jamfør seksjon 1.1 og forskingsspørsmålet i 1.2. Punkt nummer to er støtta ved å velje eit relevant og viktig problem i forretnings samband, som skildra i seksjon 3.1. Neste punkt er evaluering av design, som er støtta og demonstrert ved å gjennomføre eit eksperiment, jamfør seksjon 3.3. Punkt fire er støtta ved å gje vitskapelege bidrag i form av masteroppgåva og artefakten som er utvikla. Artefakten er basert på teknologiar som nyttar opne lisensar, og kan difor nyttast til andre vitskapelege prosjekt. Punkt nummer fem er støtta ved å nytte eit eksperiment, som er ein velkjend og mykje nytta vitskapeleg metode. Punkt nummer seks er støtta ved at artefakten er utvikla i ein iterasjon og deretter evaluert. Det siste punktet er kommunikasjon av forskning. Dette punktet er støtta ved å kommunisere forskinga til både eit teknisk orientert publikum og eit forretningsretta publikum. Artefakten og programkoden er tilgjengeleg for utviklarar som representerer eit teknisk orientert publikum. Artefakten og masteroppgåva gjev produkteigarar forklaring på korleis artefakten fungerer og kva teknologiske avgrensingar som er gjeldande for prosjektet.

I dette prosjektet vert artefakten evaluert ved eit eksperiment, seksjon 3.3 utdjupar dette nærare.

3.3 Eksperimentell design

Ein av fordelane med eit eksperiment er ifølgje Wohlin mfl. (2000) kontrollen ein har over deltakarar, objekt og instrumentering. Dette sikrar at ein er i stand til å trekkje meir generelle konklusjonar. Ein annan fordel med eksperiment er at ein kan utføre statistisk analyse ved å nytte hypotesetestingsmetodikkar, og i tillegg kan ein gjenta eksperimentet i andre prosjekt. Utgangspunktet for å gjennomføre eit eksperiment er ein ide kor ein ønskjer å syne at det finst ein relasjon mellom årsak og synt effekt. Denne ideen vert formulert i ei hypotese.

Ifølgje Oates (2006) er ei hypotese definert som ei utsegn som ikkje er blitt empirisk testa endå, men som det er mogleg å lage empiriske testar til slik at ein kan samle data som anten vil bekrefte eller avkrefte hypotesen. Ein alternativ definisjon er “A hypothesis means that we have an idea of, for example, a relationship, which we are able to state

formally in a hypothesis” (Wohlin mfl., 2000, s. 31). Eit kriterium er at hypotesen må vere mogleg å falsifisere eller avkrefte.

Vidare seier Oates (2006) at det er viktig å skilje mellom avhengige og uavhengige variablar i eksperiment. Manipulasjon av ein uavhengig variabel vil føre til ein effekt som vert spegla av ein avhengig variabel. Til dømes kan ein manipulere ein uavhengig variabel i skulesamanheng ved å gje lærarar ein ny læringsmetodikk som kan føre til at studentar får betre eksamensresultat. Eksamensresultatet er den avhengige variabelen som blir endra på grunn av ytre påverknad. I dette prosjektet er *Semantizer* den uavhengige variabelen og domeneforståing blant utviklarar er den avhengige variabelen ein ønskjer å observere.

Eit eksperiment studerer effekten som oppstår ved å forandre ein eller fleire uavhengige variablar, kalla faktorar. Andre uavhengige variablar vert kontrollert under heile eksperimentet, slik at ein kan sei at manipulasjon av ein faktor fører til ein gjeven effekt. Målet med dette eksperimentet er å syne at manipulering av éin faktor fører til ei spesifikk endring som ein kan observere. Det er difor viktig å kontrollere alle variablar som kan påverke eksperimentet, anten ved å kontrollere alle variablane samtidig eller ha fleire eksperiment (Wohlin mfl., 2000). Det finst fleire måtar å få kontroll over faktorar utover dei uavhengige variablane som kan påverke eksperimentresultatet:

- Fjerne visse faktorar frå eksperimentet
- Halde ein faktor konstant gjennom eksperimentet
- Velje forskingssubjekt tilfeldig
- Nytte kontrollgrupper
- Fjerne forskarar og forskingssubjekt sine forventningar til eksperimentet

FIGUR 3.1: Korleis kontrollere viktige faktorar i eit eksperiment

Å observere og måle resultat er avgjerande i eit eksperiment, og i dette eksperimentet vil det bli lagt ved eit spørjeskjema til forskingssubjekta. På dette skjemaet skal forskingssubjekta vurdere nokre utsegn på ein skala, som gjev eit grunnlag for å seinare analysere data kvantitativt. For å kunne måle om det er forskjell på bruken av programvaren kontra ingen programvare, er det nødvendig å gjennomføre eksperiment med og utan *Semantizer*. Ei kontrollgruppe gjennomfører eksperimentet utan bruk av programvaren, kor

forskingssubjekta får tilgang til penn og papir. Deretter vert same eksperiment utført av eksperimentgruppa, men denne gongen nyttar deltakarane *Semantizer*. Jamfør kapittel 5.

For å kunne kontrollere variablane som er i eit eksperiment er det ein fordel å ha eit eige rom som blokkerer påverknad frå miljøet rundt. Eit laboratorium er eit døme på eit slikt rom kor ein er i stand til å kontrollere ulike typar variablar. På den andre sida vil det for mange forkingssubjekt kjennes kunstig å vere i eit laboratorium sidan dette sjeldan er nytta i det daglege. Dette kan vere eit problem for forskarar som ønskjer å teste eit informasjonssystem i det miljøet det er nytta i til dagleg, til dels på skular eller ein arbeidsstad. Ved å flytte eksperimentet til eit meir naturleg miljø misser forskarane kontroll over variablane og det blir vanskelegare for forskarane å manipulere ein uavhengig variabel (Oates, 2006).

3.3.1 Validitet

I denne seksjonen vert det skildra korleis fire typar truslar kan påverke validiteten til eit eksperiment. Først er truslane for intern og ekstern validitet forklart, deretter truslar mot konklusjons- og konstruksjonsvaliditet.

Intern validitet

Truslar mot intern validitet kan påverke den avhengige variabelen med tanke på kausalitet, utan at forskaren har kjennskap til det. Dette kan true konklusjonen i eksperimentet, sidan det er mogleg at det finst kausale relasjonar mellom ein uavhengig faktor og målt effekt (Wohlin mfl., 2000). Nokre truslar mot intern validitet er synt i tabell 3.2.

Truslar mot intern validitet
<p><i>1. Historiske hendingar kan påverke individ og gruppa sitt resultat</i></p> <p>Kan vere gjeldande dersom ein deltakar er testa med fleire ulike uavhengige faktorar på forskjellige tider. Dersom ein test vart utført ein dag med for eksempel spesielle arrangement, og ein annan test vart gjennomført på ein normal dag, finst det ein risiko for at dette påverkar resultatet til eksperimentet.</p>
<p><i>2. Å gjenta ein test fleire gongar og forskingssubjekt lærer testen</i></p> <p>Dersom ein deltakar gjennomfører testinga fleire gongar og testinga er lik, så vil deltakaren lære korleis testinga fungerer.</p>
<p><i>3. Dårlig instrumentering kan påverke målinga av den avhengige variabelen</i></p> <p>Denne effekten kan oppstå på grunn av instrumenta ein nyttar i eksperimentet, som spørjeskjema, datasett eller liknande. Dårlig designa dokument kan påverke eksperimentet i negativ forstand.</p>
<p><i>4. Kompenserande rivalisering</i></p> <p>Denne effekten kan oppstå dersom for eksempel kontrollgruppa ikkje får nytte same verkty som i eksperimentgruppa. På grunn av dette kan vedkomande syne at den tradisjonelle metoden å løyse eit problem på er konkurransedyktig.</p>
<p><i>5. Demoraliserande effekt på forskingssubjekt</i></p> <p>For eksempel kan ein deltakar i kontrollgruppa bli demoralisert og lei dersom deltakaren ikkje får lov å nytte for eksempel <i>Semantizer</i> som ein del av testinga.</p>

TABELL 3.2: Truslar mot intern validitet

Ekstern validitet

Eksperimentet sin eksterne validitet er god dersom resultatata ikkje er unike under visse forhold, men er generaliserbare. At resultatata er generaliserbare vil seie at dei same resultatata kan forventast i seinare anledningar. Den beste måten å demonstrere generaliserbarheit på er å gjennomføre eksperimentet mange gongar i mange situasjonar (Oates,

2006). Truslar mot ekstern validitet er ifølgje Wohlin mfl. (2000) tilstandar som hindrar evna til å generalisere resultatet frå eit eksperiment til ein gjeven industripraksis. Wohlin mfl. (2000) hevdar at menneske, stad og tid er former for interaksjon som påverkar den uavhengige faktoren i eit eksperiment. Nokre truslar mot ekstern validitet er synt i tabell 3.3.

Truslar mot ekstern validitet
<i>1. Samanheng mellom utval og uavhengig faktor</i>
Samanheng mellom utval og uavhengig faktor er for eksempel å nytte studentar i eit eksperiment. Ifølgje Oates (2006) er studentar gjerne yngre og betre utdanna enn den generelle befolkninga, og har anna motivasjon og verdiar enn den generelle befolkninga.
<i>2. Samanheng mellom omgivnad og uavhengig faktor</i>
Samanheng mellom omgivnad og uavhengig faktor betyr at ein for eksempel gjennomfører eit eksperiment i eit miljø som ikkje samsvarar med industripraksis. Eit problem kan vere at ein nyttar gamle verkty når det er nyare verkty tilgjengeleg, og difor kan denne trusselen påverke effekten av prosjektet.

TABELL 3.3: Truslar mot ekstern validitet

Konstruksjonsvaliditet

Ifølgje Wohlin mfl. (2000) så gjeld konstruksjonsvaliditet å kunne overføre resultatet i eit eksperiment til teori eller konsept som eksperimentet handlar om. Tabell 3.4 syner nokre av truslane mot konstruksjonsvaliditet.

Truslar mot konstruksjonsvaliditet
1. Å nytte for få operasjonar (mono operation bias)
Dersom ein nyttar ein enkel operasjonalisering, anten det er éin uavhengig variabel eller éitt forskingssubjekt i eit eksperiment, så kan konstruksjonen bli underrepresentert og ikkje gi det fulle bilete av teorien bak. For eksempel <i>Semantizer</i> kor éin versjon av applikasjonen er testa i eksperimentet. Fleire versjonar kan vere nødvendig for å forstå om <i>Semantizer</i> representerer teorien det er byggja på.
2. Interaksjon mellom testing og uavhengige faktorar kan påverke resultatet
Dette kan føre til at forskingssubjekt blir påverka av testinga, og det oppstår ein effekt mellom forskingssubjekt og testing når ein er ute etter å måle effekt mellom forskingssubjekt og uavhengig variabel.

TABELL 3.4: Truslar mot konstruksjonsvaliditet

Konklusjonsvaliditet

Truslar mot konklusjonsvaliditeten til eit eksperiment er oppteken av problema som kan påverke evna til å trekkje rett konklusjon om relasjonar mellom uavhengige faktorar og resultatet i eksperiment (Wohlin mfl., 2000). Desse er synt i tabell 3.5.

Truslar mot konklusjonsvaliditet
<p>1. Svak statistisk styrke</p> <p>Dette seier noko om kor sannsynleg det er at ein test vil avdekkje eit gyldig mønster dersom H_0 er falsk. Statistisk styrke er avhengig av utvalet, og betyr at jo større utval dess sterkare er den statistiske styrken.</p>
<p>2. Brot på føresetnader til statistiske testar</p> <p>For <i>Mann-Whitney</i>-testen, jamfør seksjon 5.3.3, vert føresetnadane oppfylt ved å nytte Likert-skalaen (1-5), ha to uavhengige grupper som måler den uavhengige variabelen (kontroll- og eksperimentgruppe), sikre at det ikkje finst relasjonar mellom observasjonar i kvar gruppe eller mellom gruppene, og det er ikkje nødvendig å normalfordele variabelen.</p>
<p>3. Å leite etter resultat og feilrate</p> <p>Å leite etter spesifikke resultat som fører til at analysen ikkje lenger er uavhengig, er ein trussel mot konklusjonsvaliditet. Feilrate er for eksempel å gjennomføre tre ulike analyser med eit signifikansnivå lik $p = .05$, kor totalen vert $(1 - 0,05)^3$ som er lik 0,14.</p>
<p>4. Pålitelegheita til måla</p> <p>Konklusjonsvaliditet er avhengig av pålitelege mål, og faktorar som korleis ein ordlegg spørjeskjema, brukarhistorier og instrument vil påverke pålitelegheita til eit mål. For å finne ut om ein har pålitelege mål, kan ein utføre ein test to gongar og forvente å få same resultat på begge testane.</p>
<p>5. Tilfeldig ulikskap blant forskingssubjekta</p> <p>Denne trusselen handlar om at dersom det er stor ulikskap i ei gruppe kan dette påverke resultatet i større grad enn den uavhengige variabelen. Å velje ei meir homogen gruppe vil påverke ekstern validitet.</p>

TABELL 3.5: Truslar mot konklusjonsvaliditet

3.3.2 Pålitelegheit

Pålitelegheita (“reliability”) eller konsistensen til ein metode gjev ein indikasjon på kor like resultat metoden gjev i andre liknande situasjonar. Ifølgje Bryman (2012) er det tre faktorar som har innverknad når ein måler ein variabel, stabilitet (“stability”), intern pålitelegheit (“intern reliability”) og mellom-observatør konsistens (“inter-observer consistency”).

For å teste om ein variabel er stabil, nyttar ein gjerne to testar (“test-retest”). Felles for desse testane er at ein nyttar same utval av deltakarar. Ved to ulike høve med same utval deltakarar skal ein kunne måle høg korrelasjon mellom testane. Korrelasjon er definert som eit mål for styrken på relasjonen mellom to variablar (Bryman, 2012). Intern pålitelegheit vil i hovudsak sei at indikatorane eller svaralternativa deltakarane kan velje er konsistente gjennom heile testen eller eksperimentet. Til sist er mellom-observatør konsistens som handlar om at når fleire subjektive meiningar er involvert i innsamling av data, kan dette føre til mangel på konsistens fordi ulike subjekt kan kategorisere data forskjellig. Dersom to observatørar gjev forskjellig poengskår for ei oppgåve, indikerer dette at det ikkje er mellom-observatør konsistens i eksperimentet.

Bryman (2012) seier at sjølv om validitet og pålitelegheit er analytisk forskjellige, er dei relaterte fordi validitet krev pålitelegheit. Dersom eit mål ikkje er påliteleg, er dette målet heller ikkje valid. Dette gjeld for dei tre faktorane for pålitelegheit, som betyr at dersom eit mål ikkje er konsistent over tid er det heller ikkje gyldig. Dersom eit mål manglar intern pålitelegheit, betyr det at ein målar to eller fleire ting, og er difor ikkje gyldig. Til slutt, dersom det manglar mellom-observatør konsistens er ikkje observatørane einig om kva som vert målt og målet kan difor ikkje vere gyldig.

3.4 Intervju

For å betre kunne forklare resultatet etter eksperimentet, vert kvalitative data samla inn ved hjelp av intervju.

Eit intervju er ifølgje Sharp, Robinson og Petre (2009) “ein samtale med eit mål”. Forfattarane skil mellom fire ulike former for intervju: opne-, ikkje-strukturerte-, semi-strukturerte- og gruppeintervju. Dei tre første formene har fått namna sine etter kor

mykje kontroll intervjuaren krev av samtalanen når det blir stilt spørsmål frå eit definert sett spørsmål. Gruppeintervju er små grupper med ein gruppeleiar som fungerer som rettleiar eller guide.

Ikkje-strukturerte intervju kan sjåast på som ein uformell samtale om eit spesifikt emne. Det blir ikkje stilt spesifikke krav til brukaren om kva respons som skal bli returnert, i motsetnad til strukturerte intervju. Sjølv om det er ein svært open og fri form for intervjueteknikk, er det viktig at ein finn svar på det ein er ute etter og forsøker å etterforske nye spørsmål etter kvart som dei oppstår. I eit strukturert intervju er spørsmåla definert på forhand, og kan samanliknast med eit spørjeskjema (“questionnaire”). Denne intervjuemetoden er nyttig når ein har tydelege mål og spesifikke spørsmål kan bli identifisert. Spørsmåla blir formulert kort og konsist, og svaralternativ vert gjevne.

Semi-strukturerte intervju kombinerer karakteristikkar frå både strukturerte og ikkje-strukturerte intervju, og brukar difor både opne og lukka spørsmål. Personen som intervjuar har eit enkelt manus han følgjer, som gjev ein viss konsistens på alle intervju. Semi-strukturerte intervju oppfordrar intervjuobjektet til å fortelje meir om til dømes ein prototype, til personen ikkje har meir informasjon å kome med. Samtidig er det viktig å halde ein nøytral tone når ein kommuniserer med deltakaren, slik at ein ikkje leiar deltakaren til å svare på ein positiv eller negativ måte (Sharp mfl., 2009). Eit semi-strukturert intervju er nytta i dette prosjektet, og deltakarane er intervjuet kvar for seg. Jamfør vedlegg E for intervjuemanuset nytta i dette prosjektet.

Kapittel 4

Semantizer-applikasjonen

Dette kapitlet beskriv *Semantizer*- og *SynsetTagger*-applikasjonen. Først blir arkitekturen til *Semantizer* skildra, deretter SenseRelate-prosjektet og *SynsetTagger*. Deretter vert den endelege utforminga av *Semantizer* presentert, og avslutningsvis blir problem undervegs og utviklingsmiljøet skildra.

4.1 Målet med Semantizer

Målet med *Semantizer* er å utvikle eit verkty som kan fungere i eit eksperiment kor ein undersøker om bruk av semantiske teknologiar i eit systemutviklingsprosjekt bidreg til auka domeneforståing blant brukarane. Sidan Scrum ønskjer å minimere ressursar brukt til dokumentering, er det ein motiverande faktor å halde dette prinsippet ved like samtidig som ny informasjon vert tilknytt brukarhistoriene for å gjere dei endå rikare på meining.

4.1.1 Utviklingsmetode

I utviklinga av *Semantizer* vart det nytta ein personleg versjon av Scrum. Sidan utviklarlaget i dette prosjektet består av éin person og denne personen også er produkteigar, er berre nokre av Scrum-metodane relevante. Ein Scrum-vegg er oppretta på Trello¹, kor alle krav er definert i form av brukarhistorier. Trello gjer brukarhistoriene tilgjengeleg

¹<https://trello.com/>

på verdsveven. Scrum-veggen er delt inn i kategoriar som produktlogg, under utvikling, testing og ferdig. I tillegg er kategoriar for ulike formål lagt til, hovudsakleg slik at ein ikkje gløymer noko viktig.

Dersom eit problem eller ein feil er oppdaga, vert dette notert på Scrum-veggen. For å vite kva brukarhistorier som er viktigast, vert produktloggen prioritert og dei viktigaste brukarhistoriene utvikla først. *Semantizer* vart testa undervegs, og dette gjorde at ein oppdaga feil som måtte fiksast før neste brukarhistorie kunne utviklast. Denne personlege Scrum-metoden bidrog til å halde fokus på dei rette brukarhistoriene i utviklinga, og gav god oversikt av prosjektet.

4.1.2 Arkitektur

Arkitekturen til *Semantizer* er blitt utvikla med eit designmønster kalla MVC-mønsteret (“Model-View-Controller”), på norsk modell-brukargrensesnitt-kontroller-mønster. Model-delen består av applikasjonsdata og logikk, View-delen som på norsk vert kalla brukargrensesnittet eller GUI-et (“graphical user interface”) er presentasjonslaget for sluttbrukaren og Controller-delen tek i mot input frå brukaren som vert konvertert og sendt vidare til riktig modell. Resultatet er at ein oppnår lause koplingar mellom klasser, som i prinsippet betyr at klasser ikkje er avhengige av kvarandre. Dette medfører at endringar i ei klasse ikkje påverkar oppgåvene til andre klasser, og dette gjer det enklare å drive vedlikehald samt utvikling av nye funksjonar.

I dette prosjektet har primært Java vore nytta fordi det finst mange opne rammeverk og bibliotek tilgjengeleg for utvikling i Java. Søket mot *WordNet* databasen blir utført av eit rammeverk som heiter *RiTa.WordNet*². Dette er eit rammeverk utvikla for både Java, Processing og JavaScript. *RiTa.WordNet* tilbyr ei rekkje metodar for å hente ut data frå *WordNet*, og metoden *performSearch(String input)* søkjer etter termar i *WordNet*.

Ein applikasjonen som heiter *SynsetTagger* utvikla av Veres, Johansen og Opdahl (2013) vart nytta til å generere lettvekt-ontologiar frå synset og for å syne relasjonane som finst mellom synset i *WordNet*. *SynsetTagger* er skildra i seksjon 4.3.

²<http://www.rednoise.org/rita/wordnet/documentation/>

I tillegg til nemnte rammeverk og applikasjonar, vart eit open rammeverk utvikla i Perl nytta, kalla **WordNet::SenseRelate::AllWords**. Dette vart integrert i Java ved hjelp av *Process()*-metodane som Java tilbyr. Dette rammeverket tek seg av automatisk orddisambiguering og er skildra i seksjon 4.2.

4.2 Automatisk merking med SR-AW

I *Semantizer* vart det nytta ein prosess som automatisk eksekverer merking av tekst. Ted Pedersen har i prosjektet SenseRelate³ utvikla Perl-pakkar som nyttar semantisk likskap og slektskap for å utføre *automatisk orddisambiguering* (“word sense disambiguation; WSD”), omsetjing henta frå Lyse (2003). Automatisk orddisambiguering handlar om å tileigne meining til eit ord basert på konteksten ordet er i. **SenseRelate::AllWords** (SR-AW)⁴ er ein metode utvikla av Pedersen som nyttar kunnskap som allereie finst i **WordNet** for å tileigne ord eit synonym. Metoden nyttar ei algoritme som finn synonymet for eit gjeve ord (“target word”) som er mest relatert eller liknar mest på nabo-orda i setninga.

I masteroppgåva til Michelizzi (2005) vart ulike mål undersøkt på tre tekstsamlingar. Alle desse måla kan nyttast i **SR-AW**. Michelizzi vektlegg kor god presisjon (“precision”) og gjenfinning (“recall”) algoritmane har, i tillegg til det harmoniske gjennomsnittet (“F-measure”). Ifølgje forfattaren er det standardalgoritmen til **SR-AW**, “The adapted lesk algorithm”, som er best eigna om ein ønskjer ein balanse mellom presisjon og gjenfinning. I tillegg er denne algoritmen best på ord som ikkje er substantiv.

SR-AW-algoritmen fungerer slik at den utfører operasjonar på teksten frå venstre mot høgre, setning for setning, ord for ord. Kvar ord vert sentrert i kontekst-vindauget, og sett til “target word”. Kontekst-vindauget er eit tal på kor mange ord som skal samanliknast i algoritmen. Standardstorleiken på kontekst-vindauget er fire, som betyr at eitt ord på venstre side, det aktuelle ordet (“target word”) samt to ord på høgre side utgjer konteksten. Storleiken på kontekst-vindauget er ifølgje Pedersen og Kolhatkar (2009) ein avgjerande faktor for kor lang tid det tek å fullføre prosessen. Tabell 4.1 syner resultatet etter eksperimentet med tre tekstsamlingar, Sem Cor (“SC”), SENSEVAL-2 (“S2”) og SENSEVAL-3 (“S3”).

³<http://www.d.umn.edu/~tpederse/senserelate.html>

⁴<http://search.cpan.org/~tpederse/WordNet-SenseRelate-AllWords-0.19/>

	2			5			15		
SC	P	R	F	P	R	F	P	R	F
lch	56	13	21	54	29	36	52	35	42
jcn	65	15	24	64	31	42	62	41	49
lesk	58	49	53	62	60	61	62	61	61
S2	P	R	F	P	R	F	P	R	F
lch	48	10	16	50	24	32	48	31	38
jcn	55	9	15	55	21	31	55	31	39
lesk	54	44	48	58	56	57	59	59	59
S3	P	R	F	P	R	F	P	R	F
lch	48	13	20	49	29	37	48	35	41
jcn	55	14	22	55	31	40	53	38	46
lesk	51	43	47	54	52	53	54	53	54

TABELL 4.1: SR-AW resultat i prosent Pedersen og Kolhatkar, 2009, s. 4

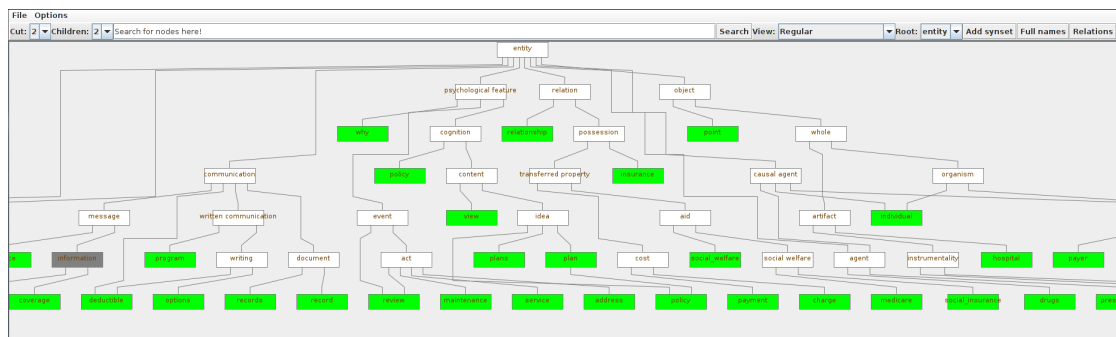
Tre mål for likskap er testa, det er “jcn” av Jiang og Conrath (1997), “lch” av Leacock, Miller og Chodorow (1998) og “lesk” av Banerjee og Pedersen (2002). Storleiken på kontekst-vindauget var 2, 5 og 15. Presisjon P , gjenfinning R og det harmoniske gjennomsnittet F syner at ein storleik på 5 gjev nesten like gode resultat som eit på 15. Presisjon og gjenfinning aukar litt med ein vindaugestorleik på 15, men på grunn av lenger køyretid er vindaugestorleiken i dette prosjektet sett til 5. Dette gjev god presisjon, gjenfinning og harmonisk gjennomsnitt i tillegg til relativt rask køyretid.

I neste seksjon er applikasjonen *SynsetTagger* skildra. *SynsetTagger* vart nytta for å visualisere substantiv i brukarhistoriene i ein trestruktur.

4.3 SynsetTagger

Applikasjonen *SynsetTagger* av Veres mfl. (2013) vart nytta til å generere lettvekt-ontologiar frå synset og syne relasjonane som finst mellom synset i *WordNet*. Ontologien vert visualisert i ein trestruktur, og ein kan velje mellom fire ulike perspektiv. Desse er “Regular”, “SuperTags”, “With inferred synsets” og “With inferred synsets+”.

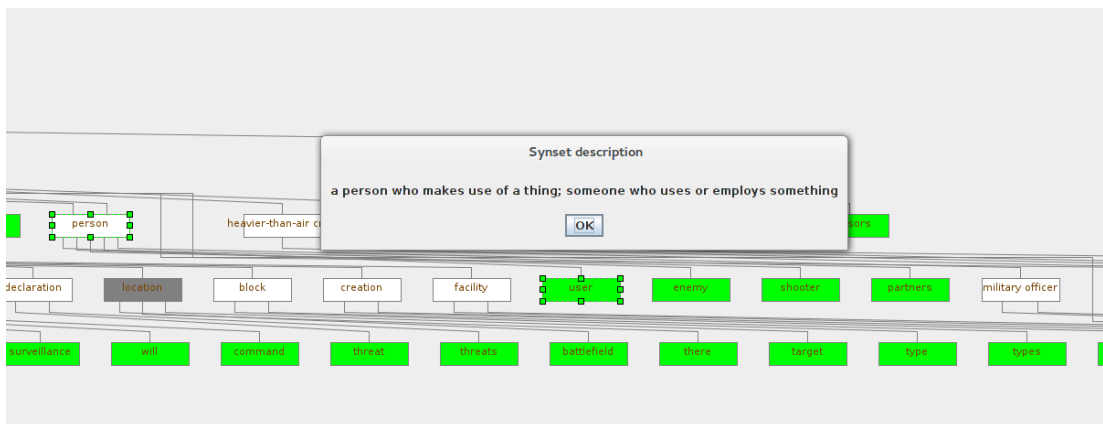
Det mest interessante perspektivet for dette prosjektet er SuperTags, sidan dette perspektivet berre syner blad-nodar og forelder-nodar som har meir enn eitt barn. Sidan antal substantiv i eit datasett med brukarhistorier kan vere meir enn hundre substantiv, vert dei andre perspektiva veldig uoversiktlege i forhold til Supertags-perspektivet. Når brukaren trykkjer på “Visualize”-knappen vil SynsetTagger-perspektivet alltid opnast i SuperTags-perspektivet, som er eit val gjort for denne applikasjonen.



FIGUR 4.1: Applikasjonen *SynsetTagger* av Veres, Johansen og Opdahl (2013).

I SuperTags-perspektivet er blad-nodane synt med grøn bakgrunnsfarge. Foreldrenodar med meir enn eitt barn har kvit bakgrunnsfarge, medan alle andre nodar er skjult. I “Regular”-perspektivet vil alle termene frå rotnoden til den blad-noden vere synleg i treet. Dette perspektivet er valt vekk sidan det gjev veldig mykje informasjon til brukaren med omtrentleg dobbelt så mange ekstra nodar i treet som kan skape forvirring i staden for oversyn. Dersom brukaren vil kan han skifte perspektiv via ein nedtrekksmeny.

I *Semantizer* vert alle brukarhistoriene sendt til ei klasse som opprettar ei XML-fil. Denne klassa sorterer ut ordklassene, og inkluderer berre substantiv med valt synonym. Ein metode lagt til for *Semantizer* opnar XML-fila når *SynsetTagger* startar. Denne metoden traverserer fila og legg deretter til nodar i treet, som synt i figur 4.1. Ved å høgreklikke på ein node får ein ulike alternativ. Nokre alternativ er blitt skjult sidan dei ikkje tilbyr nokon viktig funksjon, medan ein funksjon er lagt til. Dette er å hente forklaring av noden frå *WordNet*, som skjer via *RiTa*. *WordNet*-rammeverket. Forklaringa dukkar opp som eit vindauge, og forsvinn når brukaren trykkjer “OK”, som synt i figur 4.2. Denne funksjonen er nyttig for å finne ut, på same måte som i *Semantizer*, kva synonym som er valt for eit ord. I figur 4.2 er det valde ordet “user”, med forklaringa “a person who makes use of a thing; someone who uses or employs something”.

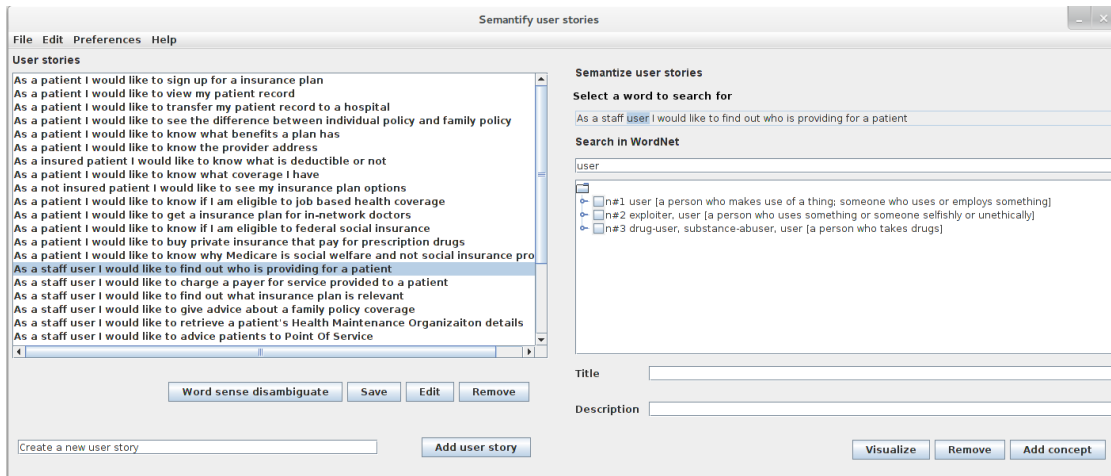


FIGUR 4.2: Forklaring til “user” i popup-vindauge.

4.4 Endeleg utforming av Semantizer

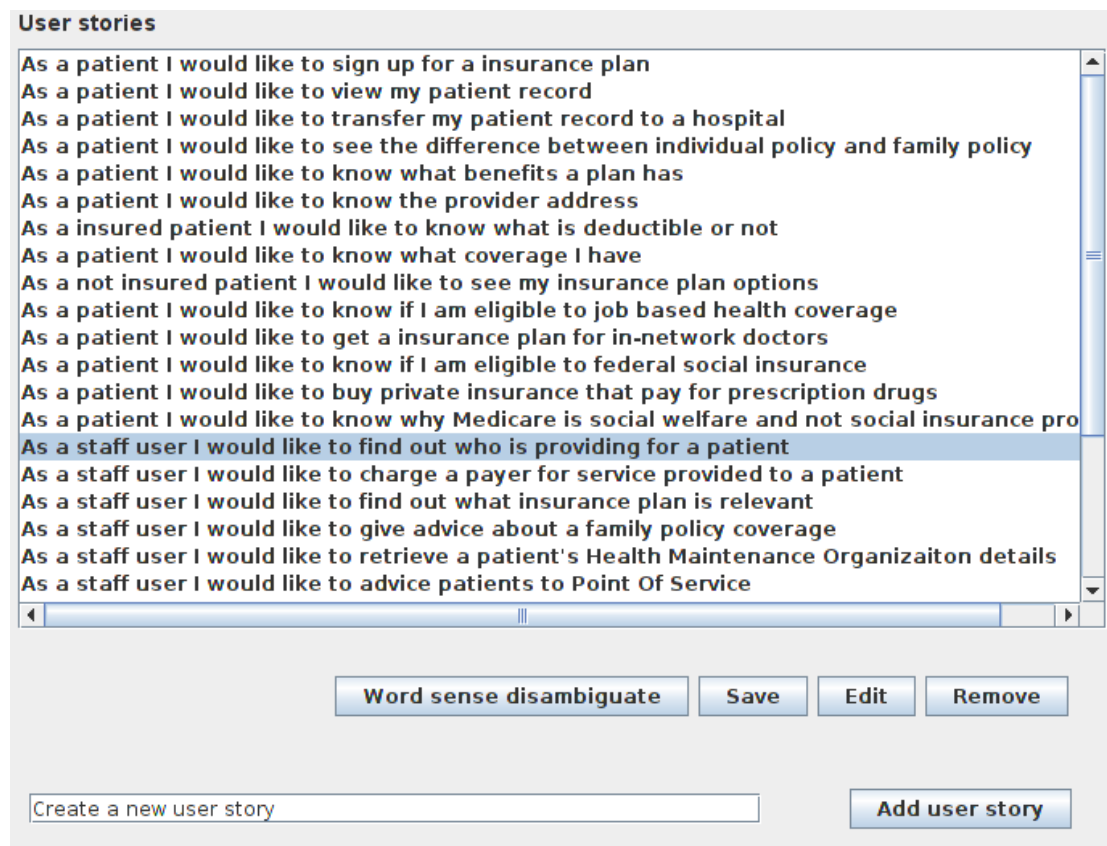
All kode for brukargrensesnittet, som for *Semantizer* i hovudsak ser ut slik som vist i figur 4.3, er plassert i ei eiga klasse. Dette er hovudsakleg designa med å bruke eit programtillegg (“plugin”) for Eclipse som heiter *WindowBuilder*⁵. *WindowBuilder* nyttar dra-og-slepp funksjonar for ulike grafiske komponentar, som kan gjere det enklare for ein utviklar å forme GUI-et. GUI-et er i hovudsak delt i to, venstre og høgre, og desse vert styrt av forskjellige kontrollere-klasser. Til venstre vert lista av brukarhistorier, knappar og menyen styrt av ei kontrollere-klasse, medan til høgre vert tekstfelta, treet og knappane under treet styrt av ei anna kontrollere-klasse. For å hente brukarhistoriene vart ein fillesar nytta for å lese ei vanleg tekstfil (.txt-format). Einaste krav til ei brukarhistorie er at den er skriva på engelsk, og at ein separerer kvar brukarhistorie i tekstfila med linjeskift. Brukarhistoriene må vere engelsk sidan *WordNet* er på engelsk, men dersom *WordNet* ein gong vert lansert med norske konsept og relasjonar vil *WordNet*-hierarkiet framleis fungere.

⁵<https://www.eclipse.org/windowbuilder/>

FIGUR 4.3: Skjermbilete av *Semantizer*-applikasjonen

4.4.1 Brukarhistorier og automatisk orddisambiguering

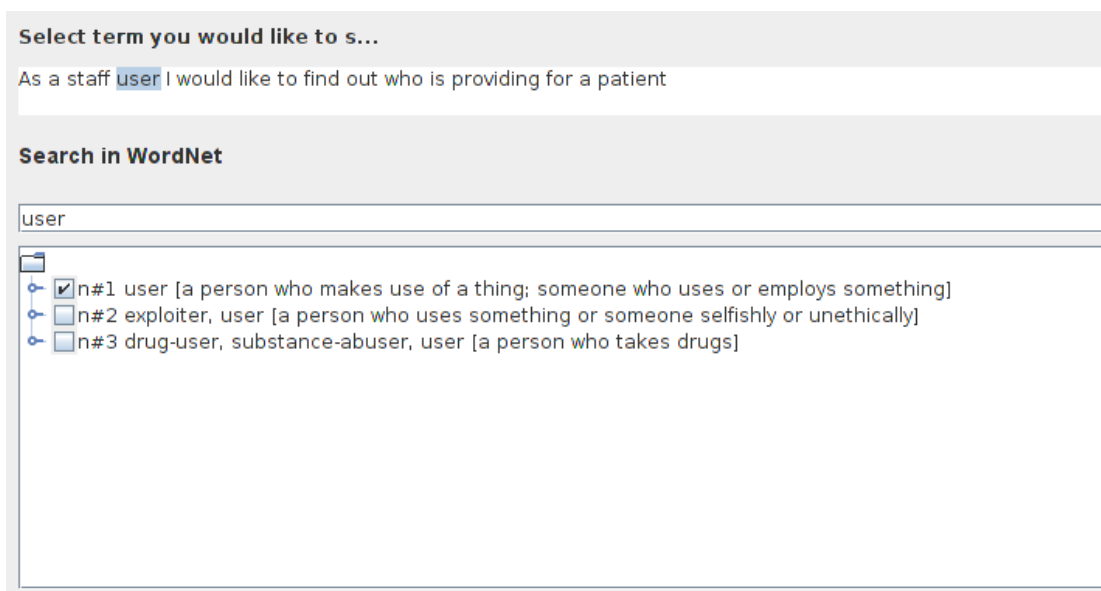
Nokre av funksjonane i *Semantizer* er vist i figur 4.4. Automatisk orddisambiguering (“word sense disambiguate”) er ein prosess beskrive i seksjon 4.2. Denne prosessen køyrer i bakgrunnen når den vert aktivert og er nødvendig for å automatisk velje det synonymet som passar best til eit gjeve ord basert på konteksten. “Save”-knappen lagrar brukarhistoriene til ei tekstfil ved hjelp av ein filskrivar. Lagre-funksjonen er ein gammal funksjon som kunne vore forbetra med til dels automatisk lagring, men dette har ikkje vore prioritet i prosjektet. Difor må ein trykkje på knappen dersom ein ønskjer å lagre endringar i brukarhistorielista. “Edit”-knappen opnar eit popup-vindauge som lar brukaren endre teksten i ei spesifikk brukarhistorie. “Remove”-knappen fjernar ein valt brukarhistorie, men først vert eit popup-vindauge opna som ber brukaren bekrefte valet. Dette er for å unngå at ei brukarhistorie vert fjerna ved eit feilklikk. Nedst i figur 4.4 er ei tekstline med teksten “Create a new user story” og ein knapp “Add user story”. I tekstlinja kan brukaren skrive inn nye brukarhistorier, og leggje desse til i lista ved å trykkje på knappen. Dersom ein har mange brukarhistorier i eit eksisterande dokument, kan ein leggje desse direkte inn i tekstfila som inneheld brukarhistoriene. Ein funksjon for å leggje til fleire brukarhistorier ved hjelp av “kopier og lim inn” var aktuell å leggje til, men vart ikkje prioritert.

FIGUR 4.4: Skjermbilete av brukarhistoriene og funksjonar i *Semantizer*.

4.4.2 Søk etter ord i WordNet

Når brukarhistoriene er lagt inn i *Semantizer*, kan brukaren velje ei brukarhistorie som deretter vert lagt til på høgre side i applikasjonen, som synt i figur 4.5. I dette tekstfeltet kan ein brukar markere enkeltord eller samansette ord, som deretter vart nytta som input for søket i *WordNet*. Søket blir utført av *RiTa.WordNet*-rammeverket. *RiTa.WordNet* har ein metode, `performSearch(String input)`, som tek i mot ein tekststreng "input" som er éin eller fleire termar. Først sjekkar ein om termen finst i ei stoppliste, før eventuelle "synset" blir henta ut. Eit synset i *WordNet* er eit sett av synonym, og settet kan vere tomt eller innehalde fleire synonym. Dersom settet er tomt, betyr dette at termen som er søkt etter ikkje fins i *WordNet* eller på grunn av ein feil i *RiTa.WordNet*-rammeverket. Denne feilen er at dersom eit synonym ikkje har både hyper- og hyponym, vert det returnert eit tomt synset. Dette problemet er rapportert til utviklaren av *RiTa.WordNet*.

Dersom synsettet ikkje er tomt, vert synonym med tilhøyrande hypernym og hyponym henta frå *WordNet*. Kontroller-klassa opprettar eit objekt frå ei klasse i Modell-delen kalla *RitaObject*, og legg til data returnert frå *performSearch(String input)*. *RitaObject* blir nytta av kontroller-klassa til *Semantizer* sitt JTree (“Java tree”) for å popularisere dette treet med data. Eit eksempel på dette er “user” i figur 4.5. Data er presentert slik det er skildra i seksjon 2.4.1, på dette formatet: *ordklasse # nummer synonym [forklaring]*.



FIGUR 4.5: *WordNet*-representasjon av termen “user”, med tre ulike synonym.

Eit mål med *Semantizer* er å gje produkteigaren og utviklarlaget meir informasjon om eit ord i form av synonym og forklaring. Dette fører til at produkteigarane sjølv kan velje det synonymet som passar best i ein gjeven kontekst, ved å klikke i avkryssingsboksen som synt i figur 4.5. Den valde brukarhistoria er

As a staff user I would like to find out who is providing for a patient

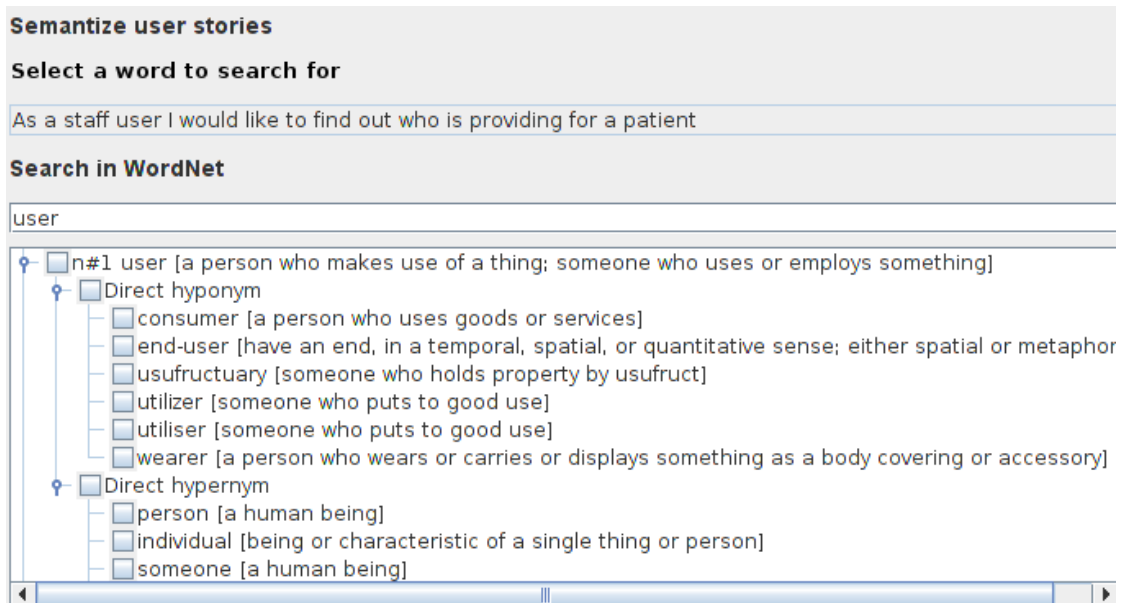
og brukaren markere eit eller fleire ord som deretter vert søkt etter i *WordNet*. I dette eksempelet er det valde ordet “user”, som har eit synset med tre synonym i *WordNet*. Desse er:

1. n#1 user [a person who makes use of a thing, someone who uses or employes something]

2. n#2 exploiter, user [a person who use something or someone selfishly or unethically]
3. n#3 drug-user, substance-abuser, user [a person who takes drugs]

I figur 4.5 har allereie prosessen for automatisk orddisambiguering blitt utført, og det synonymet som er mest rett ifølgje denne prosessen er det første synonymet. Dette markerer avkryssingsboksen i nemnte figur, men dersom det er ønskeleg kan produkteigar endre det valte synonymet ved å fjerne markeringa og velje eit anna synonym.

Dei tre alternative synonyma oversett til norsk er “brukar”, “utnyttar” og “stoffmisbrukar”. Gjeven brukarhistoria, er det relativt enkelt å velje rett synonym, nemleg “user”. Dersom brukaren ønskjer kan han klikke på ikonet til venstre for avkryssingsboksen for å få meir informasjon om hierarkiet til ordet, som synt i figur 4.6. “Direct hyponym”-kategorien inneheld konsept av ei finare inndeling, det vil sei eit konsept med meir spesifikk meining. Dette er eit nivå ned i **WordNet**-strukturen. Som figuren syner er blant anna “consumer” og “end-user” ei finare inndeling av “user”. Motsett er eit nivå opp i **WordNet**-strukturen kalla for “direct hypernym”. Hypernym er ein kategori for ord med breiare tyding. For eksempel er konseptet “user” ei finare inndeling av “person [a human being]” eller “individual [being or characteristic of a single thing or person]”. Med moglegheit til å navigere i **WordNet**-strukturen kan forhåpentleg dette hjelpe brukarane med å forstå konsept som er uklare. I tillegg kan brukaren velje det synonymet som han meiner er rett, og valet kan bli spesialisert eller generalisert ved å velje eit alternativ i hypernym eller hyponym strukturen. Dersom brukaren meiner at ingen av synonyma er rett, kan han opprette eit nytt synonym, som vert nærare forklart i neste avsnitt.



FIGUR 4.6: Ekspandert JTree med hyponym og hypernym for termen “user”.

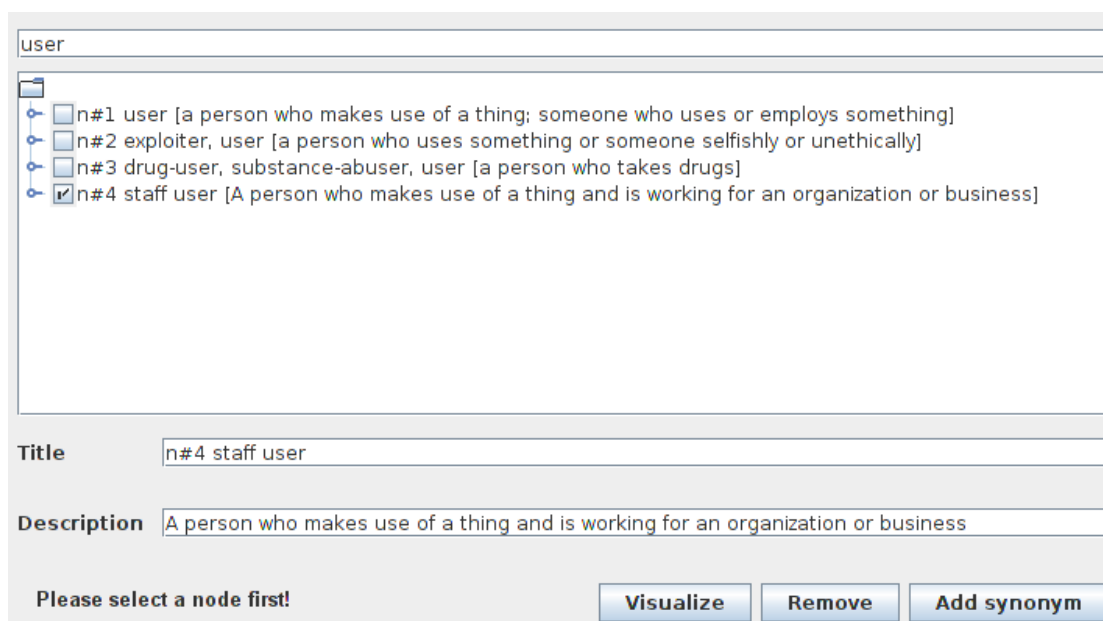
4.4.3 Synonym definert av produkteigar

Eit mål med *Semantizer* er at produkteigaren skal kunne leggje til egne synonym for eit gitt konsept. Dette er aktuelt av fleire grunnar, men den mest openbare er at *WordNet* ikkje har full støtte for veldig spesifikke termar, som for eksempel “staff user”. Det er mogleg å leggje til egne synonym via GUI-et, men funksjonane for å lagre desse synonyma i ein struktur tilsvarande *WordNet* er ikkje blitt utvikla. Ein kan anta at for å få dette til å fungere, ville brukarspesifikke synonym bli lagra i ein separat trestruktur. På same måte som *RiTa.WordNet* hentar ut informasjon frå *WordNet*, skal same rammeverk brukast for å hente ut informasjon frå den eksterne trestrukturen.

Dersom ein brukar vil leggje til eit nytt synonym, er dette det same som å leggje til ein ny node i treet og tileigne denne noden data. “Add synonym”-knappen, som synt i figur 4.7, legg til ein ny tom node, som har standardteksten “Set title” og “Set description” for høveleg tittel og forklaring. “Remove concept”-knappen fjernar eit synonym frå GUI-et, men det er ikkje lagt til funksjonar for å fjerne data frå ein separat trestruktur. Å fjerna data frå *WordNet* er ikkje eit alternativ, sidan denne databasen er haldt ved like av Princeton University. “Visualize”-knappen parsar alle substantiva som finst i brukarhistoriene til ei XML-fil. Dette blir gjort ved hjelp av ein XML-filskriver, som skriv til ei fil som *SynsetTagger* applikasjonen nyttar, jamfør seksjon 4.3.

I figur 4.7 har ein produkteigar lagt til eit nytt synonym, og gjeve dette tittelen “n#4 staff user” og forklaringa “A person who makes use of a thing and is working for an organization or business”. For å gjere det enklare å velje riktig ordklasse, vil ei forbetring vere å ha fire val i ein nedtrekksmeny, kor alternativa er substantiv (s), verb (v), adverb (av) og adjektiv (a). Nummeret vil automatisk bli det neste naturlege talet. Om eit synonym blir fjerna skal nummeret bli oppdatert for alle synonym som er involvert i endringa, utan at brukaren treng tenke på dette.

Funksjonen med å leggje til nye synonym fungerer som ein peikepinn på korleis dette kunne blitt implementert i GUI-et. Med meir tid ville det vere aktuelt å implementere ein database lik *WordNet* for å lagre brukargenererte synonym og relasjonar.



FIGUR 4.7: JTree med brukartilpassa synonym, “staff user”.

Med menneskeleg forståing vil ein produkteigar kunne både velje det riktige synonymet og leggje til nye synonym dersom dei eksisterande synonyma ikkje beskriv ordet godt nok. I figur 4.7 valde brukaren å leggje til eit nytt synonym, og merka dette synonymet som valt ved å klikke i avkryssingsboksen. Når brukaren vel det riktige synonymet manuelt, er det produkteigar sin forståing for domenet som avgjer kva som er riktig alternativ. I desse tilfella utfører produkteigaren manuell merking (“manual annotation”), for å velje det riktige synonymet (Erdmann, Maedche, Schnurr & Staab, 2000). Dette er ein tidkrevjande prosess dersom produkteigar må endre og leggje til mange synonym, men resultatet er ord og brukarhistorier som vert meir presise.

Dette summerer funksjonane som er implementert i *Semantizer*. I neste avsnitt vert nokre av utfordringane med utviklinga av *Semantizer* presentert.

4.5 Utfordringar undervegs

For å kunne modellere resultat frå *WordNet* i *Semantizer*, var det nødvendig å nytte ein oversiktleg måte og syne syno-, hyper- og hyponym på. Ein trestruktur vart nytta til dette formålet, men å implementere strukturen var tidkrevjande. Seinare i utviklinga vart det også aktuelt å syne kva synonym som var valt, og strukturen måtte utvidast med avkryssingsboksar.

Ei anna utfordring var å få dei aktuelle pakkane frå SenseRelate-prosjektet til å fungere i Windows. Valet fall til sist på å flytte utviklinga frå Windows 8 til Ubuntu 13.10, kor SenseRelate-prosjektet let seg installere smertefritt. Deretter måtte utviklingsmiljøet i Ubuntu settast opp.

Nokre mindre problem med *RiTaWordNet*-rammeverket medførte ein del sondering, og det vart utveksla e-postar med utviklaren av rammeverket, i tillegg vart ein feil rapportert. Denne feilen vart delvis retta av utviklaren av nemnte rammeverk.

Brukardefinerte synonym er ikkje lagra, og vil forsvinne når ein oppdaterer applikasjonen. Det vart vurdert å implementere lagringsfunksjonar, men å finpusse på eksisterande funksjonar i *Semantizer* hadde høgare prioritet. Dette var for å kunne gå i gong med eksperimentet.

4.6 Utviklingsmiljø

Windows 8 og Ubuntu 13.10

I byrjinga skjedde utviklinga av *Semantizer* på Microsoft Windows 8. Problem med å installere Perl-pakkar frå SenseRelate-prosjektet til Ted Pedersen i Windows 8, førte til at Linux og distribusjonen Ubuntu 13.10⁶ vart nytta i staden for Windows 8.

⁶<http://www.ubuntu.com/>

Eclipse IDE og IntelliJ IDEA

I byrjinga av utviklinga på Windows-plattformen vart Eclipse IDE⁷ og WindowBuilder⁸ nytta. Eclipse er eit populært og gratis utviklingsverktøy. WindowBuilder er ein tilleggspakke til Eclipse som gjer det enklare å lage brukargrensesnittet, blant anna med dra-og-slepp funksjonar. I overgangen til Ubuntu 13.10 vart Eclipse IDE bytta ut med IntelliJ IDEA⁹. Dette var hovudsakleg på grunn av anbefalingar frå medstudentar med erfaring frå IntelliJ IDEA og Eclipse IDE.

GitHub

For versjonkontroll og moglegheit for gjenoppretting av kode samt tekst, vart GitHub¹⁰ nytta. Studentar får i ei avgrensa periode lov til å opprette gratis og lukka pakkebrønner til studieoppgåver.

Sublime Text 2 og Latex

Masteroppgåva er skrive i Latex med Sublime Text 2¹¹ som teksteditor.

RiTaWordNet og WordNet

*WordNet*¹² er ein engelsk leksikalsk database som har substantiv, adverb, verb og adjektiv gruppert i sett av kognitive synonym, kalla synset, som beskriv eit spesifikt konsept.

*RiTa. WordNet*¹³ er eit rammeverk for å hente ut data frå *WordNet*.

SenseRelate

SenseRelate¹⁴ er eit prosjekt kor ein brukar mål for semantisk likskap og slektskap for å utføre automatisk orddisambiguering (“word sense disambiguation”).

⁷<http://www.eclipse.org/downloads/moreinfo/java.php>

⁸<https://www.eclipse.org/windowbuilder/>

⁹<http://www.jetbrains.com/idea/>

¹⁰<https://github.com/>

¹¹<http://www.sublimetext.com/>

¹²<http://wordnet.princeton.edu/>

¹³<http://www.rednoise.org/rita/wordnet/documentation/>

¹⁴<http://www.d.umn.edu/~tpederse/senserelate.html>

Kapittel 5

Forskningsdesign

I dette kapitlet blir det presentert korleis designforskning og eksperimentell design vart følgt for å svare på forskingsspørsmåla. Først er det ein gjennomgang av korleis ein har følgt retningslinene i designforskning gjennom studiet. Deretter ei detaljert framstilling av eksperimentdesignet som vart nytta. Avslutningsvis er instrumenta, pilotstudie og intervju nytta i eksperimentet skildra.

5.1 Designforskning

I denne seksjonen vert prosjektet definert med utgangspunkt i retningslinene til designforskning, som er forklart i seksjon 3.2.

1. Design i form av ein artefakt

Artefakten som vart utvikla er ein applikasjon som skal kople saman brukarhistorier med *WordNet*, kor målet er å auke domeneforståinga til utviklarlaget. *Semantizer* kan klassifiserast som ein konstruksjon, og oppfyller første retningsline for designvitskap. Konstruksjonen syner med eksempel henta frå den verkelege verda at funksjonane programmet tilbyr fungerer. Det vil seie at brukarhistorier gjeve av ein produkteigar skal analyserast og ord tileignast eit synonym. Ved hjelp av automatisk merking skal alle ord i setninga tildelast riktig synonym basert på kontekst. (Meir om automatisk merking i seksjon 4.2.) Dei valde orda og synonyma vart deretter nytta til å visuelt syne relasjonar mellom ord og korleis hierarkiet i *WordNet* klassifiserer desse orda. Dette blir gjort

med ein applikasjon som heiter *SynsetTagger* av Veres mfl. (2013), meir om *SynsetTagger* i seksjon 4.3. Målet med å nytte *SynsetTagger* er at det skal bli enklare for produkteigarar som opprettar brukarhistorier å forstå relasjonane og objekta som finst i brukarhistoriene. Samtidig skal utviklarlaget kunne lese og tolke desse brukarhistoriene og betre forstå domenet som vert utvikla. Dette kan bidra til å auke domeneforståinga blant medlemmane i utviklarlaget, som kan leie til færre spørsmål tilbake til produkteigar. Det medfører meir tid nytta til utvikling av produktet.

2. Problemrelevans

Som nemnt i seksjon 3.1 så medfører Scrum og smidig systemutvikling nokre problem, som til dels at ein ønskjer å ha ein produkteigar til stades hos utviklarlaget. Det er bedrifter som ikkje har ressursar til dette, og difor må ein kommunisere på andre måtar, som kan stogge utviklinga av produktet. Grunnane til at ein produkteigar ikkje kan vere til stades kan vere at denne personen ikkje kan ta seg tid frå eksisterande arbeidsoppgåver eller av økonomiske årsaker. Jamfør seksjon 2.2.2 for produkteigar sitt ansvar i smidig systemutvikling.

Eit mål med applikasjonen som skal utviklast er at den skal gje meir informasjon om brukarhistoriene tilbake til utviklarlaget. På denne måten kan ein forhåpentlegvis avklare nokre av spørsmåla som dukkar opp undervegs i utviklinga, slik at ein kan redusere behovet for kommunikasjon med produkteigaren.

3. Evaluering av design

For å undersøkje om bruken av den semantiske teknologien *WordNet* i eit systemutviklingsprosjekt aukar domeneforståinga til utviklarlaget, vart eit eksperiment nytta som evaluering. Eksperiment er ein velprøvd evalueringmetode, nærare forklart i seksjon 3.3. Meir om eksperimentdesignet i seksjon 5.2.

4. Forskningsbidrag

Forskningsbidraget er denne masteroppgåva, og eit resultat som vil gje svar på om *WordNet* kan auke domeneforståinga til eit utviklarlag i eit smidig systemutviklingsprosjekt. *Semantizer* er utvikla med teknologiar som er tilgjengeleg for alle, og rammeverka er lisensiert under frie og opne lisensar. Dette opnar for gjenbruk, modifikasjon og vidare utvikling av *Semantizer*, og difor er *Semantizer* også ein del av forskningsbidraget.

5. Forskningsstyrke

For å sikre kvalitet vart det nytta metodar som designvitskap og eksperimentelt design for utvikling og evaluering. Desse er velkjende og mykje nytta vitskaplege metodar, jamfør seksjon 5.2 for skildring av eksperimentdesignet.

6. Design som ein søkje-prosess

Sidan designforskning ifølgje Hevner og Chatterjee (2010) er ein iterativ metode, er det i dette prosjektet valt å gjennomføre éin iterasjon. Artefakten må difor sjåast på som ein prototype under utvikling, kor ein iterasjon med utvikling og evaluering er gjennomført. For å utvikle ein applikasjon som er formålstenleg er det nødvendig å nytte seg av verkty som har opne lisensar og difor er tilgjengeleg for alle som ønskjer å bruke verktya. Når første iterasjon i utviklinga av *Semantizer* er over, vart eit eksperiment nytta for å evaluere applikasjonen. Deretter vart det gjennomført intervju for å samle kvalitative data for å utdjupe resultatet av eksperimentet.

7. Kommunikasjon av forskning

Hevner og Chatterjee (2010) differensierer publikum i to grupper: teknologi- og forretningsretta publikum. Utviklarlaget som består av utviklarar, designerar, testarar og fleire utgjør den teknologiretta gruppa. Denne gruppa vil potensielt kunne nytte og vidareutvikle *Semantizer*, og treng difor eit detaljert innsyn i kva teknologi som ligg bak og korleis teknologiane vart nytta. Difor er all kode produsert i løpet av prosjektet lisensiert under frie og opne lisensar. Eit døme på forretningsretta publikum er leiarane i eit firma. Ein representant som fungerer som produkteigar frå denne bedrifta kan nytte programmet for å lage brukarhistoriene, og masteroppgåva gjev dokumentasjon på korleis *Semantizer* fungerer. Fokuset i dette prosjektet rettar seg først og fremst mot eit teknologiretta publikum som driv med utvikling av programvare, og som kan nytte *Semantizer* eller ideane applikasjonen byggjer på for vidare forskning og utvikling.

I neste seksjon er eksperimentet forklart. Eksperimentet vart nytta som evalueringsmetode, og utfyller punkt 3 i denne seksjonen som er definert i seksjon 3.2.

5.2 Eksperimentdesign

For å definere målet med eksperimentet vart det nytta ein mal henta frå Wohlin mfl. (2000, s. 42), og for dette eksperimentet ser malen slik ut:

Analyse av *Semantizer* <objektet>
med formålet å evaluere <formålet>
med tanke på domeneforståing <kvalitetsfokus>
frå eit forskarperspektiv <perspektiv>
med studentar i eit kontrollert eksperiment <kontekst>

Objektet som vert studert i dette eksperimentet er *Semantizer*. *Formålet* er å evaluere om bruken av *Semantizer* aukar domeneforståinga til utviklarlaget, difor er *kvalitetsfokus* i eksperimentet domeneforståing. Det er nytta eit *forskarperspektiv* i dette eksperimentet. *Forskingssubjekta* er studentar på mastergrad i informasjonvitskap ved Universitetet i Bergen, og det vert nytta eit seminarrom for å kontrollere variablar. Det er nytta eitt objekt i eksperimentet, *Semantizer*, og studiet er gjennomført med eit eksperimentoppsett kalla mellom-gruppe design, jamfør seksjon 5.2.3. I eksperimentet søker ein å verifisere eller falsifisere hypotesen som er definert i seksjon 1.2. Når ein nyttar eit eksperiment samlar ein inn kvantitative data og analyserer resultatet. Dette vil gje svar på om null-hypotesen (H_0) må forkastast eller ikkje.

Først vert avhengige og uavhengige variablar presentert, deretter val av forskingssubjekt eller deltakarar. Så vert eksperimentoppsettet skildra, kor fordelar og ulemper med to ulike eksperimentoppsett vert diskutert og kvifor eitt av desse vart nytta i dette eksperimentet. Etter dette vert truslar mot eksperimentet sin interne og eksterne validitet diskutert, i tillegg til truslane mot konstruksjons- og konklusjonsvaliditet.

5.2.1 Avhengige- og uavhengige variablar

Det er to typar variablar som er viktige å skilje, nemleg uavhengige- og avhengige variablar, noko som er forklart i seksjon 3.3. I dette eksperimentet er den uavhengige variabelen *Semantizer*. Det vert testa om bruken av *Semantizer* har påverknad på den avhengige variabelen, nemleg domeneforståing. Dersom resultatata viser auka forståing for eit domene ved bruk av *Semantizer* er den alternative hypotesen (H_1) bekrefta.

For å kunne sikre at det er ein spesiell uavhengig variabel som fører til endring i den avhengige variabelen, er det viktig å kunne kontrollere dei uavhengige variablane. For å få til dette er eit kontrollert eksperiment nødvendig. Eit seminarrom eller auditorium

ved Universitetet i Bergen vart nytta til dette formålet. I eit seminarrom er det enklare for forskaren å kontrollere dei uavhengige variablane som kan påverke ein deltakar, til dels andre deltakarar, studentar, mobiltelefonar, lyd eller tilgang til internett. Det kan kjennes kunstig for ein deltakar i eit slikt miljø, men dette miljøet gjev eit betre fokus retta mot eksperimentet og ikkje andre forstyrrende element.

Figur 3.1 syner nokre måtar å kontrollere faktorar på i eit eksperiment. Desse er blant anna å halde ein faktor konstant under eksperimentet. For eksempel var *Semantizer* og datasetta som vart nytta faktorar som var konstante, desse vart ikkje endra på når eksperimentet hadde byrja. Det same gjaldt for spørsmåla i spørjeskjema.

5.2.2 Val av forskingssubjekt

I dette prosjektet er *Semantizer* tiltenkt personar som driv med utvikling av programvare og produkteigarar. Produkteigarar er spesialistar innanfor eit eller fleire spesifikke domene dei arbeider med, og har ofte betre domenekunnskap i desse domena enn utviklarlaget. Det er difor interessant å forske på om utviklarlaget tileignar seg betre domeneforståing ved å nytte *Semantizer*.

Utviklarlaget er som nemnt i seksjon 2.2.1 personane som tek seg av utviklinga av produktet, og desse har kompetanse på systemutvikling. I tillegg har utviklarlaget kjennskap til Scrum og brukarhistorier, og kan visualisere og forstå kva som er ønska av det ferdige produktet. Difor er punktet å “velje forskingssubjekt tilfeldig” i figur 3.1 vanskeleg å oppfylle, sidan det vil vere stor forskjell på å spørje nokon med og nokon utan kjennskap til systemutviklingsdomenet. Difor er forskingssubjekta tilfeldig valde studentar på mastergrad i informasjonsvitskap ved Universitetet i Bergen. Desse tek på seg rolla som utviklarar i utviklarlaget, og har allereie kunnskap om smidig systemutvikling og Scrum.

Rekruttering av deltakarar til eksperimentet skjedde hovudsakleg muntleg, og dei aller fleste studentane gav tilbakemelding anten muntleg eller via sosiale media.

5.2.3 Eksperimentoppsett

I dette eksperimentet er det fire grupper og totalt 20 deltakarar. Deltakarane er delt i to eksperimentgrupper og to kontrollgrupper, med fem deltakarar i kvar gruppe. Som

tabell 5.1 syner, nyttar ein to domene. Desse er helseforsikring og militære kommando- og kontrollsystem, skildra i seksjon 5.3.1. Basert på desse domena vart det laga brukarhistorier som deltakarane las igjennom. Deltakarane i kontrollgruppene har totalt 10 minutt kor dei kan lese og studere brukarhistoriene, medan eksperimentgruppene har 15 minutt. Dette er for at deltakarane i eksperimentgruppene skal kunne bruke funksjonane i *Semantizer* og ha tid nok til å lese alle brukarhistoriene. Etter dette har deltakarane totalt nye 10 minutt til å svare på eit spørjeskjema. Kontrollgruppene brukar ikkje *Semantizer*, desse les brukarhistoriene på papir. KG 1 (kontrollgruppe 1) gjennomfører først testen for helseforsikring, deretter testen for kommando- og kontrollsystemet. KG 2 (kontrollgruppe 2) gjennomfører først testen for kommando- og kontrollsystemet, deretter testen for helseforsikring.

	Utan applikasjon	Applikasjon
Helseforsikring	KG 1 / KG 2	EG 1 / EG 2
Kommando-og kontrollsystem	KG 2 / KG 1	EG 2 / EG 1

TABELL 5.1: Eksperimentoppsett, mellom-gruppe design

Eksperimentgruppene EG 1 og EG 2 (eksperimentgruppe 1 og eksperimentgruppe 2) nyttar same rekkjefølgje som kontrollgruppene, men eksperimentgruppene nyttar *Semantizer* til å lese og undersøkje brukarhistoriene. Deltakarane kan også velje spesifikke ord i ei brukarhistorie for å sjå dei ulike synonyma ordet har. Ein fiktiv produkteigar har allereie nytta *Semantizer* og kjørt prosessen for *automatisk orddisambiguering* for alle brukarhistoriene. Dette er gjort for å spare tid, sidan testing viser at det tek mellom tre og fire minutt å gjennomføre prosessen for *automatisk orddisambiguering*. Resultatet av denne prosessen var at eitt eller fleire synonym vart foreslått som det riktige synonymet for eit ord, basert på konteksten. I tillegg kan brukaren trykkje på knappen “Visualiser” for å opne *SynsetTagger*-applikasjonen og importere alle substantiva frå brukarhistoriene.

Eksperimentoppsettet har blitt endra frå det opphavleg tenkte oppsettet. I utgangspunktet var eit eksperimentoppsett som synt i tabell 5.2 det aktuelle for eksperimentet. Dette vert ifølgje Rogers, Sharp og Preece (2011) kalla for *innanfor-gruppe design* (“within-subjects design”, “same-participant design”). To grupper med ti deltakarar i kvar gruppe, kor begge gruppene tek på seg rolla som både kontrollgruppe og eksperimentgruppe. I

dette oppsettet er Gruppe 1 først kontrollgruppe utan *Semantizer* for helseforsikringsdomenet, deretter eksperimentgruppe med *Semantizer* for kommando- og kontrollsystemet. Motsett er Gruppe 2 først kontrollgruppe for kommando- og kontrollsystemet, deretter eksperimentgruppe med *Semantizer* for helseforsikringsdomenet.

	Utan applikasjon	Applikasjon
Helseforsikring	Gruppe 1	Gruppe 2
Kommando-og kontrollsystem	Gruppe 2	Gruppe 1

TABELL 5.2: Opphavleg tenkt eksperimentoppsett, innanfor-gruppe design

Fordelane med innanfor-gruppe design er at ein eliminerer individuelle forskjellar sidan ein deltakar er ein representant for både kontroll- og eksperimentgruppa. Det trengs også færre deltakarar enn alternativet, *mellom-gruppe design* (“different-participant design”, “between-subject design”). Ei ulempe med innanfor-gruppe design er at ein må planleggje rekkjefølgja på deltakarane som utfører eksperimentet, slik at ein balanserer kor mange deltakarar som utfører eksperimentet med og utan applikasjon (“counterbalancing”). Dette nøytraliserer læringseffekten frå første oppgåve, kjent som rekkjefølgje-effekten (“order effect”)(Rogers mfl., 2011). I tillegg påpeiker Lazar, Feng og Hochheiser (2010) at innanfor-gruppe design tek lenger tid enn mellom-gruppe design, sidan deltakarane må fullføre eksperimentet både med og utan *Semantizer*.

I staden for innanfor-gruppe design vart mellom-gruppe design nytta (Rogers mfl., 2011). Dette eksperimentoppsettet medfører at deltakarane berre testar med eller utan *Semantizer*. Det vil sei at eksperimentgruppene som synt i tabell 5.1 utfører eksperimentet med *Semantizer* og kontrollgruppene utfører eksperimentet utan *Semantizer*. Ifølgje Rogers mfl. (2011) er fordelane med mellom-gruppe design at rekkjefølgja ikkje har nokon effekt på resultatet. Lazar mfl. (2010) nemner at dette designet er ryddigare, ein unngår at deltakarar lærer undervegs og ein har betre kontroll over andre faktorar slik som utmatting og frustrasjon. Ulempene med mellom-gruppe design er at ein treng mange deltakarar og at individuelle forskjellar mellom deltakarar kan vere eit problem. I tillegg seier Lazar mfl. (2010) at det kan vere vanskelegare å oppnå statistisk signifikante resultat med eit mellom-gruppe design. Eit samandrag av fordelane og ulempene med desse to eksperimentoppsetta er presentert i tabell 5.3.

	Fordelar	Ulemper
Innanfor-gruppe design	<ul style="list-style-type: none"> • Individuelle forskjellar vert eliminert på tvers av grupper • Treng færre deltakarar 	<ul style="list-style-type: none"> • For å unngå læringseffekten må oppgåver balanserast • Stor innverknad på utmatting og frustrasjon • Tar lang tid
Mellom-gruppe design	<ul style="list-style-type: none"> • Rekkjefølgje-effekten eliminert • Ryddig • Unngår læring undervegs • Betre kontroll på andre faktorar 	<ul style="list-style-type: none"> • Treng mange deltakarar • Individuelle forskjellar • Vanskeleg å oppnå statistisk signifikante resultat

TABELL 5.3: Fordelar og ulemper med mellom-gruppe og innanfor-gruppe design

Grunnen til at innanfor-gruppe designet synt i figur 5.2 vart bytta ut med mellom-gruppe designet, var fordi i innanfor-gruppe designet kunne deltakarane lære korleis verktøyet og oppgåvene fungerer. Dette kunne ha gjeve utslag på resultatet på spørjeskjemaet som deltakarane svarte på til sist, sidan deltakarane kan ha lært kva intensjonen med eksperimentet ear. I tillegg måtte deltakarane i eksperimentgruppa utført eksperimentet ein og ein, på grunn av at *Semantizer* berre var tilgjengeleg på ei datamaskin. På grunn av desse faktorane vart eksperimentdesignet endra til mellom-gruppe design. Dette oppsettet eliminerer faren med deltakarar som lærer undervegs i eksperimentet, sidan kontrollgruppene utfører eksperimentet utan applikasjon og eksperimentgruppa med applikasjon.

5.2.4 Intern validitet

For at eksperimentet skulle få ein god intern validitet vart truslane mot intern validitet presentert i tabell 3.2 diskutert nærare.

Første trussel er historiske hendingar som kan påverke individ og gruppa sitt resultat. Dette kan vere ein trussel dersom ein deltakar testar fleire ulike uavhengige faktorar

på forskjellige tider. Dersom ein test vart utført ein dag med for eksempel spesielle arrangement, og ein annan test vart gjennomført på ein normal dag, er det ein risiko for at dette kan påverke resultatet til eksperimentet. I dette eksperimentet vart testinga utført i løpet av ei veke, men det vart ikkje undersøkt om deltakarane deltok på nokon spesielle arrangement den veka testinga finn stad. Deltakarane i eksperimentgruppene gjennomfører eksperimentet ein og ein, i motsetnad til deltakarane i kontrollgruppene som gjennomfører i grupper på fire til seks personar. Kontrollgruppa er difor meir utsett for første trussel, historiske hendingar, enn eksperimentgruppa.

Trussel nummer to er å gjenta ein test fleire gongar som kan føre til at forskingssubjekt lærer testen. Dette kan vere ein trussel, og er aktuell sidan testinga vart utført to gongar for kvar deltakar. Sidan begge testane skal utførast i løpet av kort tid (< 1 time), vil ikkje historie påverke resultatata i nokon stor grad, men ein kan ikkje utelukke at ein deltakar ikkje endrar svaralternativ på spørjeskjemaet dersom deltakaren kjenner til testprosessen.

Trussel nummer tre handlar om dårleg instrumentering og korleis ein målar den avhengige variabelen. Deltakarane kan bli dyktigare til å løyse oppgåvene i eit eksperiment etter som deltakarane får meir erfaring med oppgåvene. Dette kan påverke resultatet i eit eksperiment i negativ forstand. I dette eksperimentet vart kontrollgruppene og eksperimentgruppene sine resultat målt opp mot kvarandre, og ved hjelp av statistisk analyse kan ein undersøkje om gruppene har forskjellig resultat på dei to datasetta. I tillegg kan ein undersøkje om deltakarane lærer eksperimentet undervegs, ved å sjå etter betre resultat på datasettet som vart testa sist.

Trussel nummer fire er kompensierende rivalisering, kor ein deltakar i kontrollgruppa ikkje får nytte same verkty som i eksperimentgruppa. På grunn av dette kan vedkomande syne at den tradisjonelle metoden å løyse eit problem på er konkurransedyktig. Dette kan vere ein trussel mot intern validitet i dette eksperimentet.

Punkt nummer fem er ein trussel mot intern validitet dersom ein deltakar i kontrollgruppa blir demoralisert og lei av å ikkje få nytte for eksempel *Semantizer* som ein del av testinga. Dersom dette er tilfelle for eksperimentet, kan det påverke resultatata til deltakaren og kontrollgruppa i negativ forstand.

5.2.5 Ekstern validitet

Like viktig som god intern validitet er god ekstern validitet. For å oppnå god ekstern validitet vert også truslane mot ekstern validitet diskutert, som synt i tabell 3.3.

Den første trusselen mot ekstern validitet er samanheng mellom utval og uavhengig faktor, som for eksempel å nytte studentar i eksperimentet. Studentar er gjerne yngre og betre utdanna enn den generelle befolkninga, og har anna motivasjon og verdiar enn den generelle befolkninga, seier Oates (2006). Dette kan medføre at resultatet av eksperimentet er generaliserbart for andre resultat av eksperiment utført med studentar som forskingssubjekt, men ikkje for den generelle befolkninga. I dette eksperimentet tek studentar på seg rolla som utviklarar i eit utviklarlag, sjølv om dei aller fleste studentar ikkje har same erfaring, forventing og press frå leiinga til å levere eit produkt som ein sysselsett utviklar har. Om *Semantizer* ein gong skal nyttast i eit profesjonelt utviklingsmiljø, kan resultat frå eit eksperiment med utviklarar vere nyttig. Denne masteroppgåva har avgrensa tid og ressursar, og difor vart studentar nytta som forskingssubjekt.

Den andre trusselen er samanheng mellom omgivnad og uavhengig faktor, som betyr at ein gjennomfører eksperimentet i eit miljø som ikkje samsvarar med industripraksis. Eit problem kan vere at ein nyttar gamle verkty når det er nyare verkty tilgjengeleg. *Semantizer* er eit heilt nytt verkty for å hente ut og syne meir informasjon om brukarhistorier, og tilbyr meir informasjon enn dagens Scrum-tavle med kartotek kort som representerer brukarhistorier. Eksperimentet vart ikkje utført i ei naturleg ramme for eit utviklarlag, så dette kan hatt innverknad på deltakarane. Sidan kontrollerte miljø er det beste for å kontrollere faktorar som spelar inn på eit eksperiment, er denne trusselen akseptabel i dette eksperimentet.

5.2.6 Konstruksjonsvaliditet

Den første trusselen er å bruke éin enkel operasjonalisering, anten det er éin uavhengig variabel eller éitt forskingssubjekt i eit eksperiment. Dette kan føre til at den teoretiske konstruksjonen blir underrepresentert og ikkje gir det fulle bilete av teorien bak. I dette eksperimentet vart det nytta éin versjon av *Semantizer*, og resultatet frå testinga med denne versjonen vil ifølgje Wohlin mfl. (2000) reflektere denne versjonen av *Semantizer*. For å reflektere den teoretiske konstruksjonen slik den er tenkt, må ein teste med fleire

versjonar av *Semantizer*. I dette studiet vart det gjennomført ein iterasjon med utvikling og evaluering, og fleire iterasjonar i tråd med designvitskapleg metode vil eliminere denne trusselen.

Den andre trusselen er interaksjon mellom testing og uavhengig variabel. Dette kan føre til at forskingssubjekt blir påverka av testinga, og det oppstår ein effekt mellom forskingssubjekt og testing når ein er ute etter å måle effekt mellom forskingssubjekt og uavhengig variabel. I dette eksperimentet kan denne trusselen bli observert i anten eksperiment- eller kontrollgruppene, dersom ein deltakar tydeleg forsøker å anten gjere det bra eller dårleg på eksperimentet. Dette kan difor påverke den målte effekten i eksperimentet.

5.2.7 Konklusjonsvaliditet

Første trussel mot konklusjonsvaliditet, som synt i tabell 3.5, er svak statistisk styrke. Dette seier noko om kor sannsynleg det er at ein test vil avdekkje eit gyldig mønster dersom H_0 er falsk. Statistisk styrke er avhengig av utvalet, og betyr at jo høgare utval dess sterkare er den statistiske styrken. I dette eksperimentet er ein ikkje-parametrisk test kalla Mann-Whitney nytta, jamfør seksjon 5.3.3. For å betre den statiske styrken vil det vere nødvendig med fleire deltakarar i eksperimentet.

Trussel nummer to er å krenke føresetnadane til statistiske testar. Det vil sei at ein ikkje oppfyller dei føresetnadane som er nødvendige for å gje rett resultat. For Mann-Whitney testen så oppfyller ein desse føresetnadane ved å nytte Likert-skalaen (1-5), to uavhengige grupper som måler den uavhengige variabelen (kontroll- og eksperimentgruppe), eliminerer relasjonar mellom observasjonar i kvar gruppe eller mellom gruppene, og til sist så normalfordelar ein ikkje variablar.

Trussel nummer tre er å leite etter resultat, kor ein leitar etter spesifikke resultat som fører til at analysen ikkje lenger er uavhengig. Feilrate er for eksempel å gjennomføre tre ulike analyser med eit signifikansnivå lik $p = .05$, kor totalen vert $(1 - 0,05)^3$ som er lik 0,14. I dette eksperimentet er ikkje målet å finne spesifikke resultat for å avkrefte H_0 , men å gjennomføre eit valid eksperiment. Difor er analysen ikkje manipulert på nokon måte, og er framleis uavhengig.

Trussel nummer fire er pålitelegheita til målet ein nyttar i eksperimentet. I dette eksperimentet spelar faktorar som korleis ein ordlegg instrumenta ei viktig rolle. Dette vil sei

korleis spørjeskjema, brukarhistorier og intervju vert ordlagt. I eksperimentet var dette ein trussel, jamfør kapittel 6 for diskusjon omkring instrumenta.

Den siste trusselen er tilfeldig ulikskap blant forskingssubjekta. Denne trusselen handlar om at dersom det er stor ulikskap i ei gruppe kan dette påverke resultatet i større grad enn den uavhengige variabelen. Å velje ei meir homogen gruppe vil på den andre sida påverke ekstern validitet. I dette eksperimentet er deltakarane studentar som studerer mastergrad i informasjonvitskap, og dette svekkar den eksterne validiteten. Svekkja ekstern validitet minkar moglegheita til å generalisere eksperimentet sidan deltakarane ikkje var valt tilfeldig frå befolkninga.

5.3 Instrument

I denne seksjonen vert instrumenta ein nyttar i eksperimentet presentert. Først skildrar ein datasetta, deretter samtykkeskjema og spørjeskjema. Avslutningsvis presenterer ein metoden for å analysere kvantitative data i korte trekk.

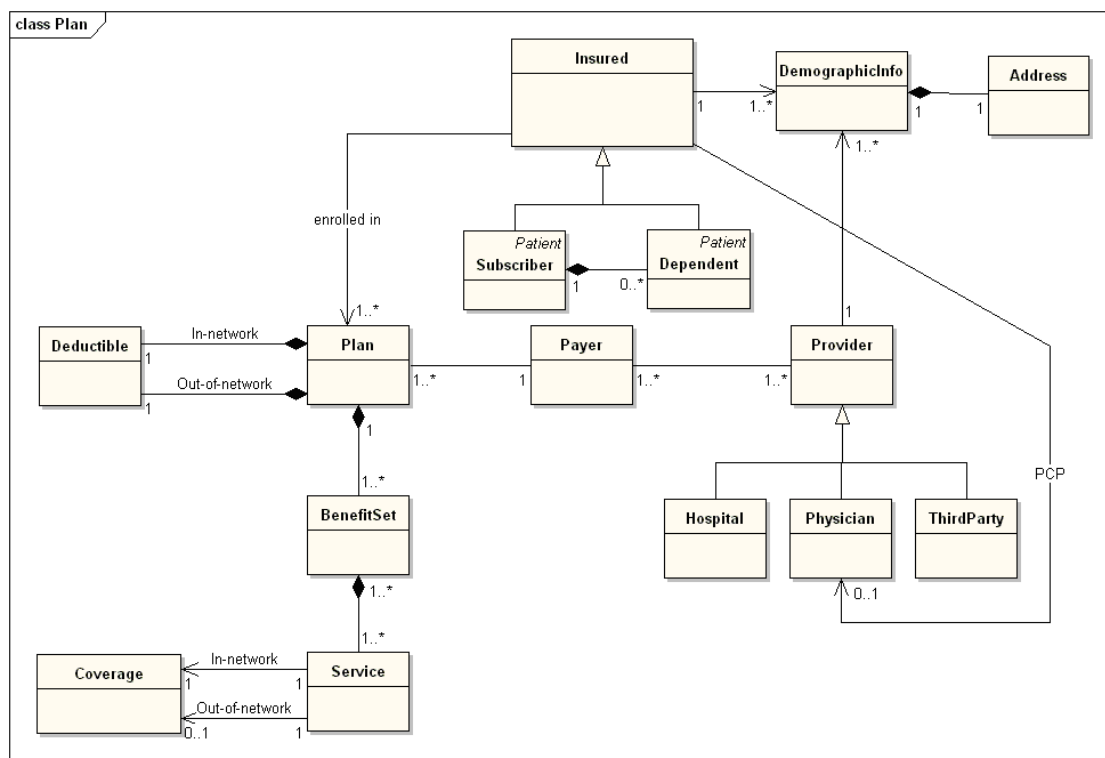
5.3.1 Datasett

Datasetta som vart nytta i eksperimentet var brukarhistorier frå to ulike domene. Det første domenet er helseforsikring (“health insurance plan”). Fleire eksterne kjelder vart nytta for å konstruere brukarhistorier til dette domenet. Blant anna vart det nytta eit UML-diagram som synt i figur 5.1. Denne figuren syner dei viktigaste objekta og relasjonane i helseforsikringsdomenet, og med artiklar som supplement som til dels frå *TLC*¹ vart det laga 61 brukarhistorier.

Det andre domenet er eit kommando- og kontrollsystem (“command and control system”), som var spesielt utvikla for militæret. Dette domenet er nøye modellert og forklart av AL-Aqrabawi (2001) som er hovudkjelda for informasjon. I tillegg til denne masteroppgåva vart ein ekspert i dette domenet kontakta for å gjere brukarhistoriene så realistiske som mogleg. Ei kort skildring av kommando- og kontrollsystemet vart lagt fram til eksperten, som deretter blei bedt om å lage brukarhistorier som matchar og utfyller dette systemet.

¹<http://tlc.howstuffworks.com/family/health-insurance.htm>

²http://en.wikipedia.org/wiki/File:Domain_model.png

FIGUR 5.1: Helseforsikring modellert i eit UML-diagram²

Eksperten vart forklart korleis brukarhistorier er bygd opp og kva målet med brukarhistorier er, og får deretter kartotek kort og penn for å lage brukarhistoriene. Med samtale og diskusjon vart det laga brukarhistorier med høg forståing for domenet. Eksperten kjente raskt igjen systemet modellert av AL-Aqrabawi (2001), som er ein modellering av “Aegis Combat System”³, eit kjent system som er nytta i fleire krigsskip i verda. Til saman vart 64 brukarhistorier oppretta, figur 5.2 syner nokre av desse på kartotek kort.

³http://en.wikipedia.org/wiki/Aegis_Combat_System

AS CO I NEED TO KNOW	① MIRC SENSOR AS WARPRE OFFICER I NEED TO KNOW THE DIMENSIONS OF MY ENEMY
WHAT SENSORS I HAVE	② AS CO I NEED TO KNOW IR-SIGNATURE ③ ④ ⑤ ⑥
	40%
As a CO I want need to be able to detect if a sensor is being spoofed by checking other sensors for verification	As a CO I want that when a threat is detected on only 1 sensor send message to operator

FIGUR 5.2: Kartotekkort med brukarhistorier for kommando- og kontrollsystemet.

5.3.2 Samtykkeskjema og spørjeskjema

Eit kombinert samtykkeskjema og spørjeskjema vart gjeve til deltakarane i eksperimentet. Samtykkeskjemaet er for å formidle til deltakarane at eksperimentet er anonymt og at deltakaren når som helst kan trekkje seg frå eksperimentet. Deretter vart formålet med datainnsamlinga forklart, og det vart opplyst at tidsfristen for å lese brukarhistoriene er ti minutt. Etter dette følgde spørsmåla til det spesifikke domenet.

Sidan deltakarane skulle svare på to spørjeskjema og dette samtykkeskjemaet var ein del av begge, vart det duplisering av informasjon og deltakarane måtte skrive under to gongar. Tanken bak dette var å sleppe eit ekstra ark berre for samtykkeskjemaet. I tillegg vil ulik rekkjefølgje på eksperimentet ikkje spele nokon rolle for kven som fekk samtykkeskjemaet først eller sist.

Spørjeskjemaet inneheld til saman 30 spørsmål. Dei tre første spørsmåla er under seksjonen om deg (“About you”) og desse spørsmåla er nødvendige for å bevisе at ein deltakar er ein student som studerer mastergrad i informasjonsvitenskap ved Universitetet i Bergen.

Deretter følger 27 spørsmål kor deltakaren skal lese ei utsegn, og deretter svare på ein skala frå veldig ueinig (“Strongly disagree”) til veldig einig (“Strongly agree”). Skalaen 1 til 5 er nytta for å analysere data. Spørjeskjema for kommando- og kontrollsystemet er i vedlegg A og spørjeskjema for helseforsikring er i vedlegg B.

5.3.3 Analyse

For å analysere data som blei samla inn vart *Mann-Whitney*-testen nytta. Mann-Whitney er ifølgje Wohlin mfl. (2000) ein ikkje-parametrisk test som er eit alternativ til den kjende t-testen. Den testar ei null-hypotese og sjekkar om to grupper er like mot ein alternativ hypotese, for eksempel at ein gjeven populasjon har større verdiar enn ein anna populasjon. Ein nyttar Mann-Whitney-testen for å samanlikne forskjellar mellom to uavhengige grupper. Testen vart nytta for å undersøkje om domeneforståinga til eksperimentgruppene og kontrollgruppene var forskjellig, med utgangspunkt i den uavhengige variabelen som var *Semantizer*-applikasjonen. Testen vart også nytta for å sjå etter forskjellar mellom dei to domena.

Mann-Whitney-testen vart utført ved å leggje alle resultata frå eksperimentet inn i applikasjonen SPSS Statistics. Dette er ein statistikk-applikasjon som er tilgjengeleg via Universitetet i Bergen ⁴, og generer tabellar og grafar som ein nyttar i presentasjonen av kvantitative data i seksjon 6.1.

5.4 Pilotstudie

For å avdekke eventuelle feil eller manglar ved eksperimentet vart det gjennomført eit pilotstudie med ein deltakar. Ifølgje Sharp mfl. (2009) nyttar ein eit pilotstudie for å sjekke om ein metode eller i dette tilfelle eit eksperiment er gjennomførleg før det ordentlege eksperimentet byrjar. Ved å ha eit pilotstudie kan ein la deltakarar undersøkje datamateriale og spørjeskjema for manglar. Eventuelle problem kan bli oppdaga og retta før ein går vidare til sjølve eksperimentet.

I pilotstudiet er framgangsmåten lik eksperimentet, det vil sei at ein først presenterer eksperimentet. Dette inneber å forklare for deltakaren kva oppgåvene er, korleis ein skal

⁴<https://tjinfo.uib.no/program>

svare på spørjeskjemaet, kor lang tid deltakaren har og at han kan trekkje seg når som helst. Deretter forklarar ein funksjonane til *Semantizer* og *SynsetTagger*, og til sist kan deltakaren stille spørsmål eller kome med tilbakemeldingar på forklaringa av eksperimentet og applikasjonane. Etter dette startar eksperimentet, kor brukaren les brukarhistorier og nyttar *Semantizer* og *SynsetTagger* for å undersøkje konsept i *WordNet*. Deltakaren kan gje tilbakemeldingar undervegs i pilotstudie. Når deltakaren er ferdig med å lese brukarhistoriene vert spørjeskjemaet gjeve til deltakaren, som kan kome med innspel på manglar og forbetringar.

5.4.1 Funn i pilotstudiet

I pilotstudiet rapporterte deltakaren at språket i brukarhistoriene var vanskeleg og dårleg strukturert. I tillegg var det nokre grammatiske feil som også burde rettast på. Dette vart teke opp av deltakaren når første del av eksperimentet var ferdig. Det har difor blitt lagt ekstra arbeid i å endre språket på brukarhistoriene, og fått språket vurdert ein gong til ved å la ein ekstern medhjelpar sjå over desse. Sidan nokre spørsmål i spørjeskjemaet ikkje var mogleg å svare på basert på brukarhistoriene, vart det lagt til nokre nye brukarhistorier.

Tilbakemeldingane på spørjeskjemaet gav mykje av dei same feila med tanke på struktur og grammatikk som brukarhistoriene, men deltakaren oppdaga også spørsmål som ikkje var tydelege nok. Det vil sei at det var vanskeleg for deltakaren å vite basert på brukarhistoriene i datasetta kva som var riktig svar. Desse spørsmåla vart endra på, og nye brukarhistorier lagt til. Eit spesifikt problem som gjekk igjen i kommando- og kontrollsystemet var at prosessar beskrive i brukarhistoriene og spørjeskjemaet var sett i hermeteikn. Dette førte til forvirring for deltakaren, som fleire gongar prøvde å søkje etter prosessane i *Semantizer*. Tilbakemeldinga for å rette opp i dette var å leggje til “the” før prosessen og “process” etter, slik at det vart sjåande slik ut:

“As a commanding officer I want the “process data” process to convert data...”

Dette vart gjort for at deltakarane lettare skulle forstå at “process data” var namnet på ein prosess i det militære kommando- og kontrollsystemet.

Brukargrensesnittet har ikkje hatt prioritet i denne masteroppgåva, og deltakaren i pilotstudiet kom med fleire råd om korleis dette kunne bli betra. Sidan pilotdeltakaren si utdanning er retta mot *HCI* (“Human-computer interaction”), hadde deltakaren mange råd om korleis brukargrensesnittet kunne forbetrast. Nokre små feil i brukargrensesnittet vart endra før eksperimentet med deltakarar starta. Dei råda som ville medført større endringar i brukargrensesnittet vart lagt til i produktloggen til prosjektet som brukarhistorier. Desse er aktuelle å ha med i komande iterasjonar i utviklinga av *Semantizer*.

Vidare sa pilotdeltakaren at bruken av *SynsetTagger* ikkje var nyttig for å betre forstå brukarhistoriene. Deltakaren forklarte at sidan denne applikasjonen listar opp alle substantiv i ein tre-struktur, kunne dette vere nyttig for å lage ontologiar, men for deltakaren var treet for uoversiktleg. Det var enklare for deltakaren å nytte *Semantizer* for å undersøkje konsept i *WordNet*.

5.5 Intervju

Etter eksperimentet var gjennomført vart det sett opp intervju med nokre av deltakarane. Deltakarane var tilfeldig utvalt frå eksperimentgruppene i eksperimentet, som betyr at dei har kjennskap til *Semantizer* og eksperimentdesignet. Intervjua fungerte som eit supplement til eksperimentet, og vart nytta for å forklare resultatata og indikere forbetringar ved *Semantizer*.

Semi-strukturerte intervju vart nytta, nærare forklart i seksjon 3.4. I denne typen intervju er det viktig å ha eit klar mål slik at ein kan rette samtalen mot det som er viktig. Samtidig fokuserer ein på å føre ein samtale og kome med oppfølgingsspørsmål dersom temaet er aktuelt. For eksempel kan det vere enklare for nokre deltakarar å sjå forbetringar i brukargrensesnittet, som er gode tilbakemeldingar for vidare utvikling av verktøyet, men hovudsakleg ønskjer ein å finne ut kva som fungerte bra og ikkje bra i *Semantizer* og *SynsetTagger* og kva deltakarane meiner om ideen med *WordNet* som domeneforklaring.

Til intervjua vart det nytta opptaksutstyr i form av mobiltelefon med mikrofon, og det vart skrive notat undervegs med penn og papir. Ifølgje Sharp mfl. (2009) distraherer penn og papir deltakaren i mindre grad enn videokamera og lyden av skriving på tastatur. Intervjua vart gjennomført på eit seminarrom, kor observatør og deltakar var til stades.

Semantizer og *SynsetTagger* vart sett opp, og deltakaren nytta funksjonane og utforska applikasjonane. Når deltakaren kjente seg ferdig med å utforskinga, vart det stilt spørsmål frå skjemaet i vedlegg E. Eventuelle oppfølgingsspørsmål frå observatør vart stilt når deltakaren ikkje hadde meir informasjon å kome med. Om deltakaren hadde spørsmål vart desse svart på etter beste evne. Totalt var det sett av ein time til intervju for kvar deltakar.

Resultatet av intervjua er diskutert nærare i kapittel 6.

Kapittel 6

Resultat og diskusjon

Dette kapitlet presenterer resultatet av eksperimentet skildra i kapittel 5. Resultatet vert analysert ved å nytte Mann-Whitney testen som beskrive i seksjon 5.3.3. Deretter følgjer ein diskusjon om brukarhistorier, spørjeskjema og innspel frå intervju. Ei evaluering av eksperimentet avsluttar dette kapitlet.

6.1 Analyse av kvantitative data

I denne seksjonen vert resultata frå eksperimentet presentert. Først ei oversikt for begge domena presentert i tabell 6.1. På grunn av for mange positive riktige alternativ, har spørsmåla blitt balansert. Dette førte til at totalt tjuesju spørsmål har blitt redusert til tolv spørsmål i denne analysen. Jamfør seksjon 6.2.2 for detaljar om denne balanseringa.

Av denne tabellen kan ein lese at det er kommando- og kontrollsystemet deltakarane skårar høgast på. Resultata representerer totalsummen som deltakarar har fått i både eksperimentgruppene og kontrollgruppene. Resultatet syner at deltakarane i eksperimentet skårar rundt 67% betre på kommando- og kontrollsystemet, enn på helseforsikringsdomenet. Dette kan bety at domenet for kommando- og kontrollsystemet var enklare å forstå enn helseforsikringsdomenet. I seksjon 6.2.1 og 6.2.2 vert brukarhistoriene og spørjeskjema diskutert, som kan bidra med forklaring på kvifor resultata varierer mellom desse domena.

Ranks				
Domain		N	Mean rank	Sum of ranks
	Command and control	20	25.65	513.00
	Health insurance	20	15.35	307.00
	Total	40		

TABELL 6.1: Middelerdi og sum for domena

Tabell 6.2 syner forskjellen på poengskåren mellom bruk og ikkje bruk av *Semantizer* for kommando- og kontrollsystem domenet. Denne syner at deltakarane i kontrollgruppene og i eksperimentgruppene skårar likt. I dette eksperimentet er signifikant nivået sett til $p = .05$, som er det vanlegaste signifikans nivået. Som tabell 6.3 viser er $p = 0.97$, som betyr at differansen ikkje er statistisk signifikant. Difor må ein anta at det ikkje finst nokon forskjell på resultatata for eksperimentgruppene og kontrollgruppene i dette domenet.

Ranks				
Application		N	Mean rank	Sum of ranks
	No application	10	10.45	104.50
	Application	10	10.55	105.50
	Total	20		

TABELL 6.2: Kommando- og kontrollsystem: applikasjon vs. ingen applikasjon

	Sum
Mann-Whitney U	49.500
Wilcoxon W	104.500
Z	-.038
Asymp. Sig. (2-tailed)	.970
Exact Sig. [2*(1-tailed Sig.)]	.971

TABELL 6.3: Statistikk for kommando- og kontrollsystem

Tabell 6.4 syner resultatata for helseforsikringsdomenet med og utan bruken av *Semantizer*. I dette domenet skårar deltakarane med applikasjon ca. 25% betre enn deltakarane utan applikasjon. Som tabell 6.5 syner er $p = 0.379$, som betyr at forskjellen blant gruppene ikkje er statistisk signifikant.

Resultata støttar null-hypotesen som er definert i seksjon 1.2.

Ranks				
Application		N	Mean rank	Sum of ranks
	No application	10	9.35	93.50
	Application	10	11.65	116.50
	Total	20		

TABELL 6.4: Helseforsikringsdomenet applikasjon vs. ingen applikasjon

	Sum
Mann-Whitney U	38.500
Wilcoxon W	93.500
Z	-.880
Asymp. Sig. (2-tailed)	.379
Exact Sig. [2*(1-tailed Sig.)]	.393

TABELL 6.5: Statistikk for helseforsikringsdomenet

6.2 Diskusjon

I denne delen av oppgåva vert det diskutert kvifor eksperimentet ikkje gjev statistisk signifikante resultat. Først vert brukarhistoriene og spørjeskjema diskutert, deretter bruken av applikasjonane *Semantizer* og *SynsetTagger*, og til sist eksperimentdesignet.

6.2.1 Brukarhistorier

Ein grunn til at resultatet ikkje er statisk signifikant kan vere brukarhistoriene og spørjeskjemaet. Som nemnt i seksjon 5.4.1, så vart brukarhistoriene forbetra strukturelt og grammatisk. Denne kvalitetskontrollen blei utført av ein ekstern person med gode engelskkunnskapar. Dette fjerna ein del av problema med lesbarheita. Likevel ville det vore mogleg å gjere brukarhistoriene og spørsmåla i spørjeskjemaet betre. Nokre av desse problema vert diskutert i denne seksjonen.

Eit viktig poeng i dette studiet er at når ein ønskjer å måle domenekunnskapen til deltakarane, vil det vere interessant å måle kor mykje kunnskap deltakarane klarer å tileigne seg frå datasetta. Dersom ein brukar har kunnskap om det gjeve domenet på førehand, kan ein anta at denne kunnskapen mest sannsynleg vil gje utslag på deltakaren sitt resultat. Difor er det ønskeleg å nytte domene som deltakarane har lite eller ingen kjennskap til. Helseforsikringsdomenet og kommando- og kontrollsystemdomenet var to

domene som fylte dette kravet, og difor vart det lagt ressursar i å skape brukarhistorier for desse domena.

Spesielt for helseforsikringsdomenet var det to kjelder som vart mykje nytta, som nemnt i seksjon 5.3.1. Dette var eit UML-diagram og ein artikkel om helseforsikring. Sjølv om desse to kjeldene utfylte kvarandre godt, var det ikkje nok til å få med alle detaljane om korleis eit informasjonssystem i dette domenet skulle fungere. Difor vart ikkje datasettet så spesifikt som ein kunne ønskje. Nokre av brukarhistoriene vart ikkje til krav som spesifikt gav informasjon om korleis dette informasjonssystemet skulle fungere, og er av den grunn ikkje gyldige brukarhistorier. For å validere brukarhistoriene burde ein ekstern person med god kunnskap til helseforsikringsdomenet vore med i skapinga av brukarhistoriene, slik som var tilfelle med brukarhistoriene for kommando- og kontrollsystemet. I tillegg ville det vore gunstig å setje av meir tid til denne delen av masteroppgåva, slik at ein kunne lage datasett og spørsmål av høgare kvalitet. Dette ville kanskje leia til andre resultat enn det som er tilfelle i dette eksperimentet.

I motsetning til dei litt vage kjeldene til helseforsikringsdomenet, var kjeldene for det militæret kommando- og kontrollsystemet meir presise. Dette kjem av som nemnt i seksjon 5.3.1 at masteroppgåva til AL-Aqrabawi (2001) gav svært detaljert informasjon om dette domenet. Prosessane frå oppdaging av ein trussel til evalueringa om målet er uskadeleggjort, er presentert med UML-diagram som klasse-diagram (“class diagram”), sekvensdiagram (“sequence diagram”) og aktivitetsdiagram (“activity diagram”). I tillegg var desse diagramma nøyaktig forklart, som gjorde arbeidet med å forstå domenet og deretter trekkje ut brukarhistorier enklare. For å halde nivået på brukarhistoriene så høgt som mogleg vart ein ekstern person med kjennskap til kommando- og kontrollsystemet hyra på frivillig basis. Personen tok på seg rolla som produkteigar, noko som gav ein produktiv sesjon med oppretting av og nærare forklaring av brukarhistorier og krav til systemet generelt. Difor var desse brukarhistoriene av høg kvalitet, og om ein liknande sesjon hadde vore gjennomført for helseforsikringsdomenet ville det kanskje auka kvaliteten til desse brukarhistoriene.

6.2.2 Spørjeskjema

Utforminga av spørsmåla i spørjeskjemaet var ein avgjerande faktor for kva resultat som blir samla inn. Som presentert i seksjon 5.4.1, vart nokre uklare spørsmål i begge domene

endra etter tilbakemelding frå deltakaren.

Eit mistak som vart gjort i utforminga av desse spørsmåla vart at balansen mellom positive og negative svar ikkje vart vurdert. Dette førte til at det riktige alternativet oftast var av positiv karakter, noko som leia til ein sterk representasjon av positive svar. Dette kan ha ført til at deltakarane oftare svarte positivt enn negativt sjølv om dei var usikre, så for å førebygge dette vart analysen redusert frå tjuesju til tolv spørsmål.

Desse tolv spørsmåla er representert med seks positive og seks negative riktige alternativ. Det vil sei at riktige svar er likt fordelt på negativ og positiv side. Sjølv etter denne balanseringa, så syner figur 6.6 at deltakarane har ein tendens til å svare positivt. Omkring 58% av svara er positive, medan ca. 22% er negative. Det kan kome av at seks spørsmål er av negativ karakter av totalt 27 spørsmål i spørjeskjemaet.

	Responses		Percent of Cases
	N	Percent	
Strongly disagree	19	7.9%	95.0%
Disagree	32	13.4%	160.0%
Neutral	50	20.9%	250.0%
Agree	49	20.5%	245.0%
Strongly agree	89	37.2%	445.0%
Total	239	100.0%	1195.0%

TABELL 6.6: Frekvenstabell for kommando- og kontrollsystem

I figur 6.7 kan ein lese at for helseforsikringsdomenet er ca. 45% av svara av positiv karakter gjeven ved “Strongly agree” eller “Agree”. Prosentvis er ca. 32% av svara av negativ karakter, det vil sei “Disagree” eller “Strongly disagree”. Denne overrepresentasjonen av positive svar i begge domena burde vore unngått ved å endre spørsmåla i spørjeskjemaet. Dette kunne vore gjort ved å setje inn til dels ordet “not” i spørsmålet, som i mange tilfelle ville gje motsett svar. For å få meir balanserte resultat vart det nytta same framgangsmåte som for kommando- og kontrollsystemet, som betyr at seks positive og seks negative riktige alternativ vart nytta i analysen.

	Responses		Percent of Cases
	N	Percent	
Strongly disagree	37	15.4%	185.0%
Disagree	40	16.7%	200.0%
Neutral	56	23.3%	280.0%
Agree	54	22.5%	270.0%
Strongly agree	53	22.1%	265.0%
Total	240	100.0%	1200.0%

TABELL 6.7: Frekvenstabell for helseforsikringsdomenet

Av desse resultatata ser ein raskt at ein med fordel burde lagt meir arbeid i utforminga av brukarhistoriene og spørjeskjemaet. Manglane nemnt over syner at den interne validiteten til eksperimentet er svekka, med fokus på instrumentering som nemnt i punkt tre i figur 3.2, truslar mot intern validitet. Brukarhistoriene og spørjeskjemaet er instrument som kunne bidratt til å betre den interne validiteten. For eksempel ville ein ekstern person med god kjennskap til helseforsikringsdomenet vore nyttig for både skapinga av brukarhistorier og spørsmål, i tillegg til å validere desse før eksperimentet vart gjennomført. Eit døme på eit informasjonssystem nytta i helseforsikringsdomenet kunne også bidratt til å auke forståinga for kva prosessar, aktørar og informasjon eit slikt system inneber. Ei masteroppgåve med detaljert informasjon og UML-diagram om helseforsikringsdomenet ville også vore nyttig for å betre kvaliteten på spørsmåla, som var tilfelle med kommando- og kontrollsystemet.

6.2.3 Innspel frå intervju

Etter eksperimentet vart det gjennomført intervju med tre av deltakarane frå eksperimentgruppene, som forklart i seksjon 5.5. Resultatet av analysen utført på desse intervju er diskutert i denne seksjonen.

Spørsmålet “kva meiner du om ideen med å bruke *WordNet* til å forklare eit domene?” gav utelukkande positive tilbakemeldingar frå deltakarane. Ein deltakar nemnte at å bruke *WordNet* var “en smart måte å nøyaktig spesifisere hva som menes med ulike ord i en brukerhistorie og dermed få en nøyaktig definisjon av et ord og brukerhistorie for et

gitt domene.” Den positive tendensen blant deltakarane syner at å nytte *WordNet* for å forklare eit domene var ein god ide.

Fungerer bra i *Semantizer*

Neste spørsmål var “kva meiner du fungerer bra med *WordNet* i desse applikasjonane?”. Ein deltakar svarte at måten ein kan slå opp *WordNet* definisjonar på ord i applikasjonen (*Semantizer*) gjorde det veldig lett å finne desse definisjonane, og påpeikte at det var enkelt å velje definisjonane og endre definisjonane dersom ein for eksempel ein produkteigar hadde gjort feil i utgangspunktet. På eit oppfølgingsspørsmål om *WordNet* kan vere nyttig i ein situasjon med produkteigar og utviklarlag, svarer deltakaren:

“Ja, fordi man har mindre tvitydigheter. Hva mener produkteier med dette ordet? Jo, da slår du opp *WordNet* definisjonen og det er da viktig at produkteier bruker riktig synonym og er flink til å tagge brukerhistoriene sine.”

Med tagge meiner deltakaren å tileigne eit synonym til eit ord, som synt i figur 4.7.

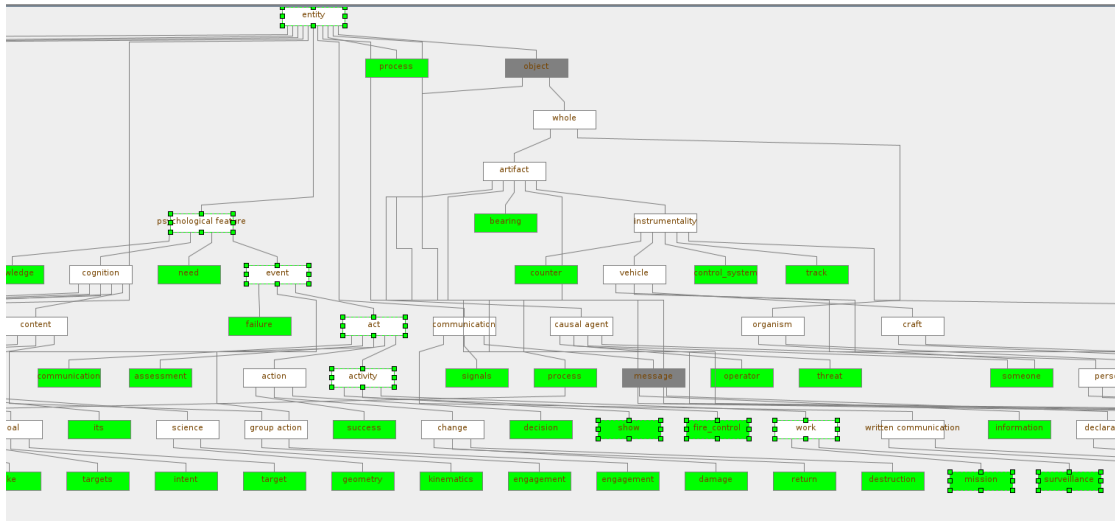
Fungerer bra i *SynsetTagger*

Når det gjeld spørsmålet om kva som fungerer bra i *SynsetTagger*, så seier ein deltakar at konteksten til eit ord hjelp til med å forstå kva eit domene dreiar seg om. I tillegg seier deltakaren at *SynsetTagger* syner kvar i hierarkiet ordet ein undersøker er, og *SynsetTagger* gjer det enkelt å sjå ord som anten er ei generalisering eller ei spesialisering av dette ordet. På denne måten kan ein forstå eit ord basert på andre ord ein kanskje kan frå før. Dette kan bidra til å auke domeneforståinga meiner deltakaren.

Eit eksempel på dette er synt i figur 6.1. Relaterte konsept er markert med små grønne firkantar på dei relevante nodane. I dette tilfellet er konseptet “activity” valt, som er forelder til “show”, “fire control” og “work”. Sistnemnte er foreldre til “mission” og “surveillance”. Om ein brukar ønskjer å sjå hierarkiet for “surveillance” kan brukaren følgje grafen tilbake til “work”, deretter “activity”, “act” og så vidare. Dette kan gje brukaren svar på kva “surveillance” tyder og korleis konseptet er klassifisert.

Fungerer mindre bra i *SynsetTagger*

Vidare til neste spørsmål “kva meiner du fungerer mindre bra med *WordNet*?” er det ingen konkrete svar, og spørsmålet vart fokusert meir mot applikasjonane *Semantizer*



FIGUR 6.1: Relasjonar til “activity”-noden.

og *SynsetTagger*. Ein kommentar til *SynsetTagger* var at applikasjonen gav for mykje informasjon, slik at det vart uoversiktleg for brukaren. Som nemnt i 4.3, vart SuperTags-perspektivet valt for å fjerne ein del av informasjonen som vart presentert for brukaren. Sidan alle substantiva i brukarhistoriene vart visualisert i *SynsetTagger* var det likevel for mange nodar i treet til at det gav deltakarane noko oversikt. To av deltakarane nemnte i intervjuet at ved å berre visualisere ei valt brukarhistorie om gongen, i staden for alle brukarhistoriene samtidig, kan dette gjere det enklare å sjå relasjonar og dra nytte av *SynsetTagger*. Dette meinte brukarane kunne hjelpe med å forstå eit domene betre.

Eit oppfølgingsspørsmål vedkomande *SynsetTagger* vart stilt: “kjenner du til andre måtar å modellere brukarhistoriene på, som for eksempel UML, som kan gjere domenet meir forståeleg?”. Ein deltakar svarte at sidan UML har “is a” og “has a” relasjonar så kan desse hjelpe med forståinga, men deltakaren gjorde også merksam på at UML-diagram raskt kan bli teknisk vanskeleg å forstå. UML-diagram kan bli eit problem for produkteigarar som ikkje har erfaring med å lage slike diagram. Det same gjeld for utviklarar som ikkje har noko erfaring med å lese og tolke UML-diagram. Konklusjonen til deltakaren var at UML-diagram godt kan fungere som eit supplement til *SynsetTagger* og *Semantizer*. Ein annan deltakar svarte at UML-diagram var mykje nytta i samband med utvikling av krav for informasjonssystem, og at slike diagram var nytta per dags dato av ein tidlegare arbeidsgjevar. Denne deltakaren meinte at UML-diagram kan fungere svært godt for å forstå både eit domene og brukarhistorier betre.

Fungerer mindre bra i *Semantizer*

Noko som ikkje fungerer så bra med *Semantizer* er ifølgje ein deltakar at “det er veldig mange brukarhistorier, og måten de blir listet opp på er ikke så oversiktlig.” Sidan *Semantizer* har fastsett høgde og breidde, så kan ikkje brukarar utvide applikasjonen for å lage meir plass til brukarhistoriene. Fastsett høgde og breidde var eit val som blei tatt i utviklinga av applikasjonen, og dette er gjort fordi det kan vere enklare å konstruere brukargrensesnittet slik ein ønskjer med faste storleikar. Eit oppfølgingsspørsmål vert stilt: “kan du nemne nokre måtar å betre strukturen på brukarhistoriene på?”. Deltakaren svarer at ein kan gruppere brukarhistoriene, slik at relaterte brukarhistorier er i same gruppe. På denne måten kan ein også visualisere ei gruppe i *SynsetTagger*. Dette vil kanskje fungere betre enn ved å visualisere alle brukarhistoriene samtidig, slik det fungerer i dag. Dette er avhengig av talet på brukarhistorier, for vert dette talet for høgt kan det føre til at *SynsetTagger* vert uoversiktleg.

For å gruppere brukarhistorier kan ein nytte dei relasjonane som allereie eksisterer i *WordNet*, og la relaterte brukarhistorier bli gruppert automatisk. Dette vil kanskje føre til at ein får visualisert relativt like tre i *SynsetTagger* kor brukarhistoriene er relevante for kvarandre, seier ein deltakar.

Korleis forstå eit domene betre

Neste spørsmål er “kva kunne vore gjort for å forstå eit domene betre?”. Eitt svar var å supplere *WordNet* med fleire kjelder, til dømes ordbøker eller leksikon frå Oxford¹ eller Merriam-Webster². I staden for å gje brukarane forslag frå alle kjeldene, kan brukarane sjølv velje kva kjelde dei vil bruke i *Semantizer*. Eventuelt om eit ord ikkje er representert i den kjelda ein nyttar, kan ein produkteigar søkje i dei andre tilgjengelege kjeldene.

Vidare seier deltakaren at det kunne vere interessant å hente meir informasjon frå for eksempel DBpedia³, som igjen hentar informasjon frå Wikipedia⁴. På denne måten kan brukaren slå opp eit ord, for eksempel “sensor grid”, og lese meir om teorien bak dette konseptet. Det kan vere nyttig sidan definisjonen gitt av *WordNet* og andre ordbøker kan vere kortfatta, og med å lese om konseptet kan ein betre forstå både konseptet og det aktuelle domenet. Denne ideen argumenterer også Miller mfl. (1990) for:

¹<http://www.oxforddictionaries.com/>

²<http://www.merriam-webster.com/>

³<http://dbpedia.org/About>

⁴<http://www.wikipedia.org/>

“A dictionary definition draws some important distinctions and serves to remind the reader of something that is presumed to be familiar already; it is not intended as a catalogue of general knowledge. There is a place for encyclopedias as well as dictionaries.” (Miller mfl., 1990, s. 11)

6.3 Evaluering av eksperimentet

Truslar mot den interne validiteten til eksperimentet, jamfør tabell 3.2, er blant anna at historiske hendingar kan ha påverka resultatet i eksperimentet, dersom spesielle arrangement har funne stad samtidig som eksperimentet vert gjennomført. Det er ikkje undersøkt om dette er tilfelle, og difor kan dette påverke resultatet. Ein anna trussel er at sidan deltakarane testar to ulike domene, vil deltakarane lære korleis testinga fungerer frå første til andre test. Dette kan føre til at deltakaren betre forstår kva spørsmål som vert stilt, og korleis deltakaren skal førebu seg best mogleg for å skåra så bra som mogleg.

I pilotstudiet vart det oppdaga manglar ved brukarhistorier og spørjeskjemaet, som vart forbetra før eksperimentet byrja. Som nemnt tidlegare bidrog denne gjennomgangen av data til å rette opp i ein del manglar, men nokre problem vart ikkje oppdaga før i analysen, blant anna for mange positive i forhold til negative riktige svar. Dette førte til at instrumenta er svekka i dette eksperimentet.

Kompenserande rivalisering blant medlemmar i kontrollgruppa kan også ha påverka resultatet, sidan det vart observert at kontrollgruppene las og studerte brukarhistoriene meir nøye enn deltakarar i eksperimentgruppene. Ei av årsakene til dette kan vere det valte eksperimentoppsettet. I motsetning til deltakarane i eksperimentgruppa, vart eksperimentet med deltakarane i kontrollgruppene gjennomført i grupper beståande av maksimalt seks personar. Det vart observert at deltakarane i kontrollgruppene gjorde ein grundigare jobb med å lese brukarhistoriene, blant anna ved å bla fram og tilbake og lese brukarhistoriene fleire gongar. Dette er ein sterk kontrast til deltakarane i eksperimentgruppene som sat aleine under eksperimentet, og gjerne las brukarhistoriene i *Semantizer* ein eller to gongar før dei var ferdige. Ei form for press eller konkurranse mellom deltakarane i kontrollgruppene om å prestere så bra som mogleg på eksperimentet, kan vere ein årsak til eit godt resultat frå desse gruppene.

Det vart også observert at ein deltakar las igjennom brukarhistoriene éin gong, og venta til tidsfristen skulle gå ut. Dette kan ha påverka resultatet til deltakaren og gruppa i negativ forstand. Ein kan anta at deltakaren var umotivert eller demoralisert, som er ein trussel mot intern validitet.

Truslar mot ekstern validitet for eksperimentet, jamfør tabell 3.3, var at studentar som spesifikt studerer mastergrad i informasjonsvitenskap ved Universitetet i Bergen, vart nytta som deltakarar i eksperimentet. Dette medførte at eksperimentet ikkje representerte den generelle befolkninga. Sidan utviklarlaget var ei spesifikk målgruppe med tanke på utdanning og kompetanse, var studentar som studerer mastergrad i informasjonsvitenskap så nære utviklarlaget som ein kunne kome i eit eksperiment. Ved å nytte profesjonelle utviklarar kan ein gjennomføre eit meir spesifikt eksperiment for utviklarlaget.

Ein annan trussel var at eksperimentet vart utført i eit kontrollert miljø som ikkje er naturleg for utviklarar, og som difor kan ha påverka resultatet av eksperimentet. Dersom ein skal måle kor godt *Semantizer* fungerer i eit naturleg miljø, må dette utførast på arbeidsplassen til utviklarlaget.

Ein av truslane mot konstruksjonsvaliditeten til eksperimentet, synt i tabell 3.4, er at det berre er brukt éin versjon av *Semantizer*. For å minimere denne trusselen må ein utføre fleire eksperiment med ulike versjonar av *Semantizer*, som er i tråd med designvitenskap som ein iterativ metode for å skape eit artefakt.

Neste trussel er deltakarane kan bli påverka av testinga, og av den grunn oppstår ein ikkje ønskja effekt mellom testing og forskingssubjekt. Dette kan ha blitt observert i kontrollgruppene kor ein deltakar tydeleg forsøker å anten gjere det bra eller dårleg på eksperimentet. Det kan henge saman med eit punkt i truslar mot intern validitet, kor rivalisering og demoralisering er truslar mot eksperimentet av liknande karakter.

Truslane mot konklusjonsvaliditeten til eit eksperiment, synt i tabell 3.4, er blant anna at styrken til statistisk test er svekka sidan det er relativt få deltakarar i eksperimentet. For å betre dette er det nødvendig med fleire deltakarar. I tillegg er spørjeskjemaet ein trussel mot eksperimentet, sidan spørsmåla har ein tendens til å ha positive svar. Neste trussel er at ein forskar kan leite etter resultat som gagnar forskingsspørsmålet, både bevist og ubevisst. I dette eksperimentet vart ikkje data eller testar manipulert, endra eller ordna slik at resultatet av forskingsprosjektet av den grunn skulle bli ugyldig.

Ein trussel allereie nemnt er instrumentet, og måten ein ordlegg seg på i instrumenta er ein avgjerande faktor for pålitelege mål. For å undersøkje om ein har pålitelege mål i dette eksperimentet, kan ein gjennomføre same eksperimentet med nye deltakarar og forventa å få likt resultat.

Dette summerer opp evalueringa av eksperimentet, og i neste iterasjon vil eit fokus på desse svakheitene bidra til å gjere eksperiment meir robust og kanskje gje meir innbringande resultat.

Kapittel 7

Konklusjon og vidare arbeid

7.1 Konklusjon

Hensikta med dette prosjektet var å undersøke om bruken av den semantiske teknologien *WordNet* ville bidra til å auke domeneforståinga blant utviklarlaget i eit smidig systemutviklingsprosjekt. For å finne ut om dette var tilfelle, vart applikasjon *Semantizer* utvikla. Denne applikasjonen nyttar hovudsakleg semantiske teknologiar som *WordNet* og SenseRelate-prosjektet som nyttar relasjonane som fins i *WordNet*.

For å svare på forskingsspørsmålet definert i seksjon 1.2, vart eit eksperiment gjennomført for å evaluere om bruken av *Semantizer* leia til auka domeneforståing for deltakarane i eksperimentet. Dette eksperimentet gjekk ut på å lese brukarhistorier frå to domene, kor domena var helseforsikring og eit militært kommando- og kontrollsystem. Etter dette blei deltakarane bedt om og svare på eit spørjeskjema. Spørjeskjemaet bestod av totalt 30 spørsmål for kvart domene, og deltakarane skulle svare på ein skala frå svært ueinig til svært einig. Kontrollgruppene gjennomførte eksperimentet med penn og papir, medan eksperimentgruppene gjennomførte eksperimentet med *Semantizer*. *Semantizer* gav deltakarane moglegheit til å søke etter ord i *WordNet*, og i tillegg fekk deltakarane ei tilbakemelding på kva synonym som var mest rett i ein gjeven kontekst.

Dei kvantitative data frå eksperimentet vart analysert, og analysen synte at det ikkje var grunnlag for å anta at desse var statistisk signifikante. Dette støttar null-hypotesen definert i seksjon 1.2, som er:

H_0 : Bruk av den semantiske teknologien **WordNet** bidreg *ikkje* til å auke domeneforståinga for utviklarlaget i eit smidig utviklingsprosjekt.

Den alternative hypotesen er H_1 :

H_1 : Bruk av den semantiske teknologien **WordNet** bidreg *ikkje* til å auke domeneforståinga for utviklarlaget i eit smidig utviklingsprosjekt.

Null-hypotesen kan difor ikkje motseiast, og den alternative hypotesen H_1 kan ikkje bekreftast.

For å få tilbakemeldingar på ideen med å nytte **WordNet** som domeneforklaring, og kva som fungerte bra eller mindre bra i **Semantizer** og **SynsetTagger**, vart eit intervju med tre deltakarar gjennomført. Tilbakemeldingane var positive, og samtlege av deltakarane syntest ideen med å nytte **WordNet** for å forklare eit domene var ein god ide. I tillegg kom deltakarane med moglege forbetringar og tips til blant anna korleis ein kan bruke andre eksterne informasjonskjelder, som DBpedia, for å gje utviklarlaget meir informasjon om eit konsept.

7.2 Refleksjonar over studiet

I dette studiet har eg nytta designvitskap som overordna metode og eksperimentell design for å evaluere artefakten som vart utvikla. Studiet har vore lærerikt og spanande, på grunn av utfordringane og løysingane eg har kome opp med i utviklinga av **Semantizer**, i tillegg til forskinga og planlegginga som eksperimentell design medførte. Ved å nytte designvitskapleg metode har eg hatt klare retningslinjer som seier kva dette studiet skal levere. Eksperimentet medførte at eg måtte setje meg inn i to domene eg tidlegare ikkje har hatt kjennskap til, og difor måtte nytte dei ressursane som fanst rundt meg. Planlegginga, utføringa og analysen av resultatata frå eksperimentet var lærerike og interessante aktivitetar i eksperimentet.

Ved å sjå tilbake på studiet som er gjort, så ville eg ha nytta meir tid på å planleggje og opprette instrumenta som var nytta i eksperimentet. Denne delen av studiet fekk for lite merksemd og var i tillegg meir ressurskrevjande enn først påreknna. Sidan eg nytta eit

eksperiment beskrive i detalj i kapittel 5, skal det vere mogleg å gjennomføre nye eksperiment som byggjer på dette eksperimentdesignet. For eksempel kan eksperiment med fleire studentar eller med utviklarar vere aktuelt, eller med brukarhistorier og spørjeskjema for andre domene.

7.3 Vidare arbeid

Meir utvikling av *Semantizer* er nødvendig for å finne ut om semantiske teknologiar har påverknad på domeneforståinga til utviklarlaget i smidige systemutviklingsprosjekt. Som allereie nemnt er det funksjonar i *Semantizer* som ikkje er implementert, men som kan vere nyttige for brukarane. Desse funksjonane er blant anna moglegheita for å leggje til eigne konsept og lagre desse i ein database lik *WordNet*. Denne databasen vil bli ei utviding av *WordNet*, og dette kan bidra meir presise brukarhistorier med fagspesifikke synonym og forklaringar.

Sidan eksperimentet ikkje gav så positive resultat som ein kanskje hadde tenkt, kan det vere aktuelt å implementere støtte for blant anna å hente informasjon frå fleire eksterne kjelder. Dette kan vere ordbøker eller databasar, kor ein for eksempel kan hente ein artikkel frå DBpedia om eit konsept ein ønskjer å lese meir om. UML-representasjonar av brukarhistoriene kan også vere eit aktuelt supplement, som kan bidra til å forstå flyten og domenet til brukarhistoriene betre enn kun tekstrepresentasjon av brukarhistoriene.

For å handtere brukarhistoriene i tråd med retningslinene i Scrum, så trengs det ei Scrum-tavle. I dette prosjektet vart det ikkje prioritert å implementere funksjonar i *Semantizer* for å handtere brukarhistorier slik ein gjer med ei Scrum-tavle. Ein kombinasjon av *Semantizer* og ei Scrum-tavle via eit vev-verkty som for eksempel Trello¹, kan bidra til organiseringa av brukarhistoriene. Dersom Trello har eit API ein kan nytte, kan det vere aktuelt å integrere desse to verktya. Dette medfører at ein kan opprette brukarhistorier i anten *Semantizer* eller Trello, og deretter tileigne ekstra informasjon via *Semantizer*, som også blir tilgjengeleg i Trello. På denne måten kan informasjon om brukarhistorier delast med utviklarlaget og produkteigar via verdsveven. Dette kan betre kommunikasjon og samarbeid mellom partane, spesielt i situasjonar kor desse gruppene ikkje er lokalisert på same stad.

¹<https://www.trello.com>

Referansar

- Ambriola, V. & Gervasi, V. (1997). Processing natural language requirements. I *Proceedings 12th ieee international conference automated software engineering* (s. 36–45). IEEE Comput. Soc. doi:10.1109/ASE.1997.632822
- AL-Aqrabawi, M. S. (2001 mai). Combat System Modeling: Modeling Large-Scale Software and Hardware Application Using UML. Hentet fra <http://scholar.lib.vt.edu/theses/available/etd-05242001-141445/>
- Aurum, A. & Wohlin, C. (Red.). (2005). *Engineering and Managing Software Requirements*. Berlin/Heidelberg: Springer-Verlag. doi:10.1007/3-540-28244-0
- Banerjee, S. & Pedersen, T. (2002 februar). An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet, 136–145. Hentet fra <http://dl.acm.org/citation.cfm?id=647344.724142>
- Bryman, A. (2012). *Social Research Methods*. Oxford University Press.
- Dingsøy, T., Dybå, T. & Moe, N. B. (Red.). (2010). *Agile Software Development*. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-12575-1
- Erdmann, M., Maedche, A., Schnurr, H. & Staab, S. (2000). From manual to semi-automatic semantic annotation: About ontology-based text annotation tools. *Group. Proceedings of the COLING '00 Workshop on Semantic Annotation and Intelligent Content*, August 5-6, 6(1), pages. Hentet fra <http://www.ida.liu.se/ext/etai/received/semaweb/003/paper.pdf>
- Farell, C., Narang, R., Kapitan, S. & Webber, H. (2002). Towards an effective onsite customer practice. I *Third international conference on extreme programming and agile process in software engineering, (italy, 2002)*. Hentet fra <http://cf.agilealliance.org/articles/system/article/file/1014/file.pdf>
- Gervasi, V. (2005). Reasoning About Inconsistencies in Natural Language Requirements. Hentet fra <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.331.4923>

- Haugen, N. (2006). An Empirical Study of Using Planning Poker for User Story Estimation. I *Agile 2006 (agile'06)* (s. 23–34). IEEE. doi:10.1109/AGILE.2006.16
- Hevner, A. & Chatterjee, S. (2010 mai). *Design Research in Information Systems*. Integrated Series in Information Systems. Boston, MA: Springer US. doi:10.1007/978-1-4419-5653-8
- Hevner, A., March, S. T., Park, J. & Ram, S. (2004 mars). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105. Hentet fra <http://dl.acm.org/citation.cfm?id=2017212.2017217>
- Jiang, J. J. & Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. I *Proc. of the int'l. conf. on research in computational linguistics* (s. 19–33). Hentet fra <http://www.cse.iitb.ac.in/~cs626-449/Papers/WordSimilarity/4.pdf>
- Lazar, D. J., Feng, D. J. H. & Hochheiser, D. H. (2010). *Research Methods in Human-Computer Interaction*. John Wiley & Sons.
- Leacock, C., Miller, G. A. & Chodorow, M. (1998 mars). Using corpus statistics and WordNet relations for sense identification. *Computational Linguistics*, 24(1), 147–165. Hentet fra <http://dl.acm.org/citation.cfm?id=972719.972726>
- Lyse, G. I. (2003). Fra speilmetoden til automatisk ekstrahering av et betydningstagget korpus for WSD-formål. The University of Bergen. Hentet fra <https://bora.uib.no/handle/1956/1594>
- Martin, A., Biddle, R. & Noble, J. (2010). An Ideal Customer: A Grounded Theory of Requirements Elicitation, Communication and Acceptance on Agile Projects. I T. Dingsøy, T. Dybå & N. B. Moe (Red.), *Agile software development* (Kap. 6, s. 111–141). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-12575-1_6
- Michelizzi, J. (2005). *Semantic relatedness applied to all words sense disambiguation /* (Doktoravhandling, University of Minnesota, Duluth). Hentet fra <http://www.d.umn.edu/~tpederse/Pubs/jason-thesis.pdf>
- Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D. & Miller, K. J. (1990 januar). Introduction to WordNet: An On-line Lexical Database *. *International Journal of Lexicography*, 3(4), 235–244. doi:10.1093/ijl/3.4.235
- Moe, N. B. & Dingsøy, T. (2008). Scrum and Team Effectiveness : Theory and Practice. I P. Abrahamsson (Red.), *Xp 2008, Inbip 9* (9. utg., 7465, s. 11–20). Limerick,

- Ireland: Springer-Verlag Berlin Heidelberg. Hentet fra DOI:10.1007/978-3-540-68255-4
- Oates, B. J. (2006). *Researching information systems and computing*. Sage.
- Osada, A., Ozawa, D., Kaiya, H. & Kaijiri, K. (2007 februar). The role of domain knowledge representation in requirements elicitation, 84–92. Hentet fra <http://dl.acm.org/citation.cfm?id=1332044.1332059>
- Pedersen, T. & Kolhatkar, V. (2009 juni). WordNet::SenseRelate::AllWords: a broad coverage word sense tagger that maximizes semantic relatedness, 17–20. Hentet fra <http://dl.acm.org/citation.cfm?id=1620959.1620964>
- Rogers, Y., Sharp, H. & Preece, J. (2011). *Interaction Design: Beyond Human - Computer Interaction*. John Wiley & Sons, Ltd.
- Schwaber, K. (2004). *Agile project management with Scrum*. O'Reilly Media, Inc.
- Sharp, H., Robinson, H. & Petre, M. (2009 januar). The role of physical artefacts in agile software development: Two complementary perspectives. *Interacting with Computers*, 21(1-2), 108–116. Hentet fra <http://linkinghub.elsevier.com/retrieve/pii/S0953543808000659>
- Studer, R., Benjamins, V. & Fensel, D. (1998 mars). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1-2), 161–197. doi:10.1016/S0169-023X(97)00056-6
- Sutherland, J. & Schwaber, K. (2010). The Scrum Guide. Hentet fra <http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide.pdf%5C#zoom=100>
- Veres, C., Johansen, K. & Opdahl, A. (2013 juni). SynsetTagger. I *Proceedings of the 3rd international conference on web intelligence, mining and semantics - wims '13* (s. 1). New York, New York, USA: ACM Press. doi:10.1145/2479787.2479802
- Wake, W. C. (2001). *Extreme Programming Explored*. Addison-Wesley Professional; 1 edition.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B. & Wesslén, A. (2000). *Experimentation in Software Engineering: An Introduction*. Norwell, MA, USA: Kluwer Academic Publishers.
- Zowghi, D., Gervasi, V. & McRae, A. (2001). Using default reasoning to discover inconsistencies in natural language requirements. I *Proceedings eighth asia-pacific software engineering conference* (s. 133–140). IEEE Comput. Soc. doi:10.1109/APSEC.2001.991469

Tillegg A

Spørjeskjema for Command and Control System

Questionnaire for Command and Control System

Welcome to this questionnaire where you will be asked to answer a few questions based on the user stories you just read. Please read the questions carefully and select one alternative for each question. This questionnaire is completely voluntary and you are free to leave at any time. The collected will be used for statistically analysis on how domain knowledge can be improved by using an application for my master thesis, "Brukarhistorier med mening".

No personal information will be stored from this questionnaire. The study, degree and age questions are used to prove that the participant are a master student of Information Science at the University of Bergen.

Thank you very much for your participation. You have 10 minutes to answer all the questions. Please do not communicate with anyone during the questionnaire.

About you

1. Study: _____
2. Expected degree: _____
3. Age: _____

Questions

4. **The sensor grid is the first three sensors on the ship**
Strongly disagree ———— Strongly agree
5. **Signal converting is done by the target grid**
Strongly disagree ———— Strongly agree
6. **A commanding officer can get an overview of all available sensors**
Strongly disagree ———— Strongly agree
7. **The infra-red signature of a threat is saved in the database**
Strongly disagree ———— Strongly agree
8. **A commanding officer can obtain information on the exact dimensions of an enemy**
Strongly disagree ———— Strongly agree
9. **“Process data” is responsible for converting signals to data**
Strongly disagree ———— Strongly agree
10. **Sensor information consists of target range, bearing and kinematics**
Strongly disagree ———— Strongly agree
11. **A possible threat is detected by the target sensor and information is sent to the surveillance sensor**
Strongly disagree ———— Strongly agree
12. **“Process data” should notify “Process information”**
Strongly disagree ———— Strongly agree
13. **The target sensor can get an image of a possible threat**
Strongly disagree ———— Strongly agree
14. **Evaluation of a target’s intent and geometry is done by “Process information”**
Strongly disagree ———— Strongly agree
15. **“Process knowledge” is responsible for identifying targets**
Strongly disagree ———— Strongly agree
16. **If a threat is discovered the command and control system will coordinate with allied partners in the battlefield**
Strongly disagree ———— Strongly agree
17. **The command and control system should evaluate the capabilities of the ship**
Strongly disagree ———— Strongly agree

18. **The shooter grid is responsible for engagement decisions**
Strongly disagree ———— Strongly agree
19. **“Make decision” is a process to evaluate the target’s capabilities versus the ship’s capabilities**
Strongly disagree ———— Strongly agree
20. **“Make decision” should evaluate the target’s destruction level**
Strongly disagree ———— Strongly agree
21. **Probability to destroy is set to a specific number by the commanding officer**
Strongly disagree ———— Strongly agree
22. **An interceptor is a type of weapon used to eliminate a threat**
Strongly disagree ———— Strongly agree
23. **“Evaluate outcome” reports failure if a threat is not destroyed**
Strongly disagree ———— Strongly agree
24. **When only one sensor detect a threat it’s probably not a good decision to engage**
Strongly disagree ———— Strongly agree
25. **Weighting a specific sensor to be more important than other sensors can help to choose the best interceptor**
Strongly disagree ———— Strongly agree
26. **A commanding officer wants to classify threats based on size and speed**
Strongly disagree ———— Strongly agree
27. **By setting target acquisition on a sensor, that sensor has to detect a threat before any engagement can be done**
Strongly disagree ———— Strongly agree
28. **Spoofing is impossible to detect by cross-checking sensors**
Strongly disagree ———— Strongly agree
29. **The infra-red and EMS sensors can both detect threats**
Strongly disagree ———— Strongly agree
30. **A commanding officer retrieves all the information and knowledge before a decision is made**
Strongly disagree ———— Strongly agree

Tillegg B

Spørjeskjema for Health insurance

Questionnaire for health insurance

Welcome to this questionnaire where you will be asked to answer a few questions based on the user stories you just read. Please read the questions carefully and select one alternative for each question. This questionnaire is completely voluntary and you are free to leave at any time. The collected data will be used for statistically analysis on how domain knowledge can be improved by using an application for my master thesis, "Brukarhistorier med mening".

No personal information will be stored from this questionnaire. The study, degree and age questions are used to prove that the participant are a master student of Information Science at the University of Bergen.

Thank you very much for your participation. You have 10 minutes to answer all the questions. Please do not communicate with anyone during the questionnaire.

About you

1. Study: _____
2. Expected degree: _____
3. Age: _____

Questions

4. **A patient always have to pay a specific amount to receive treatment from an in-network provider**
Strongly disagree ———— Strongly agree
5. **A patient has to be enrolled in an insurance plan to get prescription drugs from a doctor**
Strongly disagree ———— Strongly agree
6. **A patient has to buy his own insurance even if the patient's job has a group policy insurance**
Strongly disagree ———— Strongly agree
7. **A 65 year old person can sign up for a governmental insurance program**
Strongly disagree ———— Strongly agree
8. **A person working in the army can sign an health maintenance organization plan called Tricare**
Strongly disagree ———— Strongly agree
9. **An indemnity health insurance plan is paid up front by the patient**
Strongly disagree ———— Strongly agree
10. **Most of the insurance plans have a stop loss protection**
Strongly disagree ———— Strongly agree
11. **Only a healthy person can sign up for a care plan like health maintenance organization plan**
Strongly disagree ———— Strongly agree
12. **A managed care plan has contracts with specific providers**
Strongly disagree ———— Strongly agree
13. **A person can combine a health maintenance organization plan and a catastrophic health insurance**
Strongly disagree ———— Strongly agree
14. **A patient can view his own patient record when visiting any provider**
Strongly disagree ———— Strongly agree
15. **A provider always gets compensated by the insurance company**
Strongly disagree ———— Strongly agree
16. **A patient can transfer the patient record between providers**
Strongly disagree ———— Strongly agree
17. **Most of the insurance plans are federal social insurance plans**
Strongly disagree ———— Strongly agree

18. **A provider can be an insurance company**
Strongly disagree ———— Strongly agree
19. **A provider can not offer services to people without insurance plans**
Strongly disagree ———— Strongly agree
20. **An insurance company will pay a premium for a patient's treatment**
Strongly disagree ———— Strongly agree
21. **A family can buy a group insurance**
Strongly disagree ———— Strongly agree
22. **An actuary can raise the premium of an insurance plan**
Strongly disagree ———— Strongly agree
23. **An actuary can update a patient record**
Strongly disagree ———— Strongly agree
24. **An actuary may consider the chances a person has of being sick**
Strongly disagree ———— Strongly agree
25. **When selecting the correct insurance plan a sales agent can customize it to fit the customer's needs.**
Strongly disagree ———— Strongly agree
26. **Where a customer lives may disqualify the customer from a certain insurance plan**
Strongly disagree ———— Strongly agree
27. **A provider may only transfer a patient to another clinic or hospital if the patient has a insurance plan that covers for in-network providers**
Strongly disagree ———— Strongly agree
28. **A patient can get treatment without any insurance plan**
Strongly disagree ———— Strongly agree
29. **An actuary can calculate risk for groups of people based on age, address and gender**
Strongly disagree ———— Strongly agree
30. **A patient with more than one insurance plan can use both insurance plans to get re-funded**
Strongly disagree ———— Strongly agree

Tillegg C

Brukarhistorier for Command and control system

Command and control system

1. As a commanding officer I want to always search for possible threats by surveillance sensors
2. As a commanding officer I want the sensor grid to be responsible for converting signals for each target into data
3. As a commanding officer I need to know what sensors are available
4. As a commanding officer I need to know the infrared signature of my enemy
5. As a commanding officer I need to know the dimensions of my enemy
6. As a commanding officer I want the "process data" process to convert data from the sensors to information whether there is observed a possible threat
7. As a commanding officer I want sensor information consisting of target range, bearing and kinematics.
8. As a commanding officer I want information about a possible threat from surveillance sensor to be sent to a target sensor providing the location of a target
9. As a commanding officer I want information about a possible threat from sensor grid to be sent to a fire control sensor providing the location of a target

10. As a commanding officer I want the "process data" process to retrieve surveillance data and reconnaissance data
11. As a commanding officer I want the "process data" process to calculate target range, target track and update target knowledge
12. As a commanding officer I want the "process data" process to notify "process information"
13. As a commanding officer I want a target sensor to get target data from "process data"
14. As a commanding officer I want to adjust target sensor to the location of the target to get a image
15. As a commanding officer I want a fire control sensor to get target data from "process data"
16. As a commanding officer I want to adjust fire control sensor to the location of the target
17. As a commanding officer I want fire control sensor to get a image of the target
18. As a commanding officer I want to send output from fire control sensor to "process information" and "Process knowledge"
19. As a commanding officer I want the "process information" process to retrieve data from "Process data"

20. As a commanding officer I want the "process information" process to get data from target and fire control sensors.
21. As a commanding officer I want the "process information" process to evaluate target intent and geometry to recognize target
22. As a commanding officer I want the "process information" process to send information to "Process Knowledge"
23. As a commanding officer I want the "process knowledge" process to get information from "Process information" and "Fire control sensor".
24. As a commanding officer I want the "process knowledge" process to identify the target either as a threat or not a threat.
25. As a commanding officer I want that in case of a threat, the command and control system should coordinate with partners in the battlefield
26. As a commanding officer I want that in case of a threat, the command and control system to update ship status
27. As a commanding officer I want that in case of a threat, the command and control system should evaluate the capabilities of the ship
28. As a commanding officer I want that in case of a threat, the command and control system to notify the shooter grid for initiating engagement decision
29. As a commanding officer I want that the "make decision" process should be executed by the command and control system

30. As a commanding officer I want that the "make decision" process should be executed iteratively whenever damage assessment and outcome evaluation show the target destruction level was less than the needed level.
31. As a commanding officer I would like that the "make decision" process evaluates the capability to destroy a target based on the knowledge of the target
32. As a commanding officer I would like that the "make decision" process evaluates the capability to destroy multiple types of threats
33. As a commanding officer I would like that the "make decision" process evaluate the probability to destroy target based on its own capability
34. As a commanding officer I would like that the "make decision" process evaluate the needed capability if engagement is taken
35. As a commanding officer I would like that the "make decision" process engage if probability to destroy is greater or equal to a specified value
36. As a commanding officer I would like that if the "make decision" process engage a interceptor is sent
37. As a commanding officer I would like that the "process effect" process is engaged after interceptor is launched
38. As a commanding officer I would like that the "process effect" process return damage assessment about the target
39. As a commanding officer I would like that the shooter grid update the target's knowledge object with new information

40. As a commanding officer I would like that the "process effect" process notifies "Evaluate outcome" with the results
41. As a commanding officer I would like that the "evaluate outcome" process compare target damage to needed level of damage.
42. As a commanding officer I would like that the "evaluate outcome" process set needed level of damage based on specific target and mission
43. As a commanding officer I would like that the "evaluate outcome" process set the engagement marked as a success if needed level of damage is achieved by current interceptor
44. As a commanding officer I would like that the "evaluate outcome" process should evaluate need for further action if needed level of damage is not achieved by current interceptor
45. As a commanding officer I would like that the "evaluate outcome" process notify "Make decision" process to re-evaluate if outcome is failure
46. As a commanding officer I want to send a counter interceptor as a defensive maneuver
47. As a commanding officer I want that when only one sensor detect a threat, a message is sent to the operator
48. As a commanding officer I want to know what types of interceptors my enemy have

49. As a commanding officer I want to weight sensors
50. As a commanding officer I want to know what weapons I have available
51. As a commanding officer I want to know how many of a type of weapon is available
52. As a commanding officer I want to know if I have any Anti-Air Missiles
53. As a commanding officer I want to evaluate information gathered by the sensors
54. As a commanding officer I want to set a percentage of how sure I am that someone is a enemy before I decide to engage
55. As a commanding officer I want to classify my targets based on how dangerous they are
56. As a commanding officer I want to classify how sure I am that a target is an enemy
57. As a commander I want to set target acquisition on a sensor
58. As a commander I want to be able to detect if someone is spoofing my sensor by checking other sensors for verification
59. As a commanding officer I want to detect threats based on sensors reliability
60. As a commanding officer I want to know how a enemy will defend themselves

61. As a commanding officer I want to know what weapons an enemy has
62. As a commanding officer I want to have a secure communication to other allies in the battlefield
63. As a commanding officer I want to know what weapons my allies have in the battlefield

Tillegg D

Brukarhistorier for Health insurance

Health insurance

1. As a patient I would like to sign up for a insurance plan
2. As a patient I would like to know how much the insurance company will cover for a certain treatment
3. As a patient I would like to know if an insurance plan demands an sum from my pocket to receive treatment
4. As a patient I would like to know if my employer has a group insurance
5. As a patient I would like to know if my employer offers an insurance plan managed by a health maintenance organization
6. As a patient I would like to know if I need individual health insurance if the coverage through my employee isn't enough
7. As a patient I would like to sign up for a governmental insurance program
8. As a patient older than 65 I would like to know if I'm eligible to sign up for a governmental insurance program
9. As a patient employed by the army I want to know what kind of coverage the insurance offers
10. As a patient I would like to sign up for an indemnity insurance
11. As a patient I would like to know what the deductible amount for an indemnity insurance is

12. As a patient I want to know what the stop loss protection for my policy is
13. As a patient I would like to sign up for a managed care plan like the health organization plan
14. As a patient in a managed care plan I would like to be informed of all available hospitals in the network
15. As a patient I would like to sign up for supplemental insurance planning like catastrophic health insurance
16. As a patient I would like to view my patient record
17. As a patient I would like to transfer my patient record to a hospital
18. As a patient I would like to know what benefits an insurance plan offer
19. As a patient I would like to know if my insurance plan cover transportation costs
20. As a patient I would like to know the provider's address
21. As a patient I would like to know what is deductible and what is not
22. As a patient I would like to know what coverage my insurance plan offers
23. As a patient without insurance I would like to see my insurance plan options
24. As a patient I would like to know if I am eligible to a job based coverage

25. As a patient I would like to get an insurance plan where I can get treatment from in-network providers
26. As a patient I would like to know if I am eligible to federal social insurance
27. As a patient I would like to buy an insurance plan that covers for prescription drugs
28. As a patient I would like to know the provider's company code
29. As a staff user I would like to find out if a patient has an insurance plan
30. As a staff user I would like to charge someone for the treatment a patient has received
31. As a staff user I would like to know which providers are in the same network
32. As a staff user I would like to send a patient to an in-network clinic
33. As a staff user I would like to charge someone for a service given to a patient
34. As a staff user I would like to be able to find out what insurance plan is relevant for a specific patient
35. As a staff user I would like to give advice about family policy coverage
36. As a staff user I would like to retrieve a patient's health maintenance organization insurance plan details

37. As a staff user I would like to give advice to patients about treatments based on their insurance plan
38. As a staff user I would like to know if the patient's insurance plan cover prescription drugs
39. As a staff user I would like to contact a clinic on behalf of a patient
40. As a staff user I would like to setup payment to a provider
41. As a staff user I would like to get the address to a clinic nearby
42. As a staff user I would like to know how much of the payment is deductible for a treatment at an out-of-network hospital
43. As a staff user I would like to transfer a patient to a physiotherapist that has the specific knowledge to treat the patient
44. As a staff user i would like to view a patient's records
45. As a staff user i would like to create a new record for a new patient
46. As a staff user I would like to know what coverage a patient has
47. As a staff user i would like to add data to an existing patient's record
48. As an actuary I would like to see the patient's record
49. As an actuary I would like to calculate premiums and reserves for insurance policies
50. As an actuary I would like to evaluate the likelihood of a patient getting sick

51. As an actuary I would like to calculate insurance policies based on location
52. As an actuary I would like to develop insurance plans for specific situations
53. As an actuary I would like to report to a provider about a patient's coverage
54. As a sales agent I would like to sell insurance policies
55. As a sales agent I would like to help a customer choose the correct insurance policy
56. As a sales agent I would like to add details to new health insurance products
57. As a sales agent I would like to calculate sales prices for regular patients
58. As a sales agent I would like to maintain records related to sales
59. As a sales agent I would like to confer with clients to obtain information when claims are made on a policy
60. As a sales agent I would like to customize insurance programs to suit individual customers
61. As a sales agent I would like to contact customers and submit forms to obtain binder coverage

Tillegg E

Intervju

Deltakar brukar Semantizer og SynsetTagger og kjem med tilbakemeldingar undervegs. Fokuset er på funksjonar i verktya og WordNet. Nokre spørsmål for å leie samtalen mot riktig mål

- kva meiner du om ideen med å bruke WordNet til å forklare eit domene?
- kva meiner du fungerer bra med WordNet i desse applikasjonane?
- Kva meiner du fungerer mindre bra med WordNet?
- korleis bidrar Semantizer til å forstå eit domene?
- kva gjer at Semantizer ikkje bidreg til forståing?
- kva kunne vore gjort for å forstå eit domene betre?
- korleis meiner du SynsetTagger bidrar til å forstå domenet betre?
- er det nokon funksjonar du savnar? kva?