



Paula Savioja

Käyttäjäkeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa

Käyttäjäkeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa

Paula Savioja
VTT Tuotteet ja tuotanto



ISBN 951-38-6186-4 (nid.)

ISSN 1235-0605 (nid.)

ISBN 951-38-6187-2 (URL: <http://www.vtt.fi/inf/pdf/>)

ISSN 1455-0865 (URL: <http://www.vtt.fi/inf/pdf/>)

Copyright © VTT 2003

JULKAISIJA – UTGIVARE – PUBLISHER

VTT, Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 4374

VTT, Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax (09) 456 4374

VTT Technical Research Centre of Finland, Vuorimiehentie 5, P.O.Box 2000, FIN-02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Tuotteet ja tuotanto, Tekniikantie 12, PL 1301, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 6752

VTT Industriella System, Teknikvägen 12, PB 1301, 02044 VTT
tel. växel (09) 4561, fax (09) 456 6752

VTT Industrial Systems, Tekniikantie 12, P.O.Box 1301, FIN-02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 6752

Toimitus Leena Ukskoski

Otamedia Oy, Espoo 2003

Savioja, Paula. Käyttäjakeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa [User-centred methods in presenting the requirements of complex systems]. Espoo 2003. VTT Tiedotteita – Research Notes 2216. 132 s. + liitt. 10 s.

Avainsanat Cognitive Work Analysis (CWA), user-centred design, user-centered design, requirements specification, complex systems, usability

Tiivistelmä

Käyttäjakeskeisillä suunnittelumenetelmillä pyritään parantamaan suunniteltavien tuotteiden käytettävyyssominaisuuksia. Hyvä käytettävyys lisää tuotteen hyödyllisyyttä, tehokkuutta ja käyttömukavuutta. Monimutkaisten järjestelmien avulla hallitaan laajoja prosesseja, jotka ovat luonteeltaan dynaamisia, turvallisuuskriittisiä, hajautettuja ja pitkälti automatisoituja. Monimutkaisia järjestelmiä ovat esimerkiksi prosessilaitosten ohjausjärjestelmät sekä erilaiset liikenteenohjausjärjestelmät. Vaatimusten määrittely on suunnitteluprosessin vaihe, jossa tuodaan julki suunnittelun kohteen halutut ja merkitykselliset ominaisuudet. Kokonaisvaltaisen käytettävyyden kannalta käyttäjakeskeisten menetelmien käyttö on erityisen hyödyllistä vaatimusten määrittelyssä.

Työssä esitetään melko laajasti erilaisia vaatimusten määrittelyyn liittyviä käyttäjakeskeisiä menetelmiä. Nämä on jaoteltu sen mukaan, koskevatko ne käyttäjätutkimusta vai vaatimusten kuvaamista. Menetelmien osalta on pohdittu, mitkä niistä sopivat parhaiten ns. monimutkaisten järjestelmien suunnitteluun. Sopivien menetelmien on nostettava esiin juuri niitä vaatimuksia, jotka ovat suunniteltavan kohteen monimutkaisuuden kannalta olennaisia. Syvällisimmin perehdytään kognitiiviseen työn analyysiin, jossa suunniteltavaa järjestelmää lähestytään sillä hallittavan kohteen ominaisuuksien kautta. Tällä pyritään formatiiviseen malliin järjestelmästä. Formattiivisen suunnittelun tuloksena on järjestelmä, joka tukee käyttäjien yksilöllisiä ongelmanratkaisutapoja ja adaptoituu käyttötilanteeseen tuottaen informaatiota käyttäjän päätöksenteon kannalta merkityksellisistä kohteen osista.

Työn käytännön osuudessa vertaillaan Kognitiivisen työn analyysin ADS-malleilla sekä UML:n käyttötapauksilla esitettyjä vaatimuksia. Mallinnuskohteenä on Suomenlahden alusten ilmoittautumisjärjestelmä (SRS). Käytännön osuudessa havaittiin, että mallinnustavat tuottavat täysin erityyppisiä vaatimuk-

sia. ADS-malleilla kuvatut vaatimukset kuvaavat käyttäjän tarvitsemaa kohteesta saatavaa informaatiota, sillä ADS:lla voidaan mallintaa sitä, mikä tieto on käyttäjän kannalta merkityksellistä informaatiota. Tätä voidaan käyttää hyväksi monimutkaisen järjestelmän käyttöliittymäsunnittelussa.

Työssä havaittiin, että tilanteessa, jossa käyttäjäkeskeisen suunnittelun kohteena on jokin monimutkainen sosiotekninen järjestelmä, vaaditaan sekä käyttäjätutkimus- että vaatimusten kuvaus -menetelmiltä erityisominaisuuksia. Molemmissa vaiheissa on pystyttävä ilmentämään mallinnuksen kohteen niitä piirteitä, jotka tekevät siitä monimutkaisen ja sosioteknisen. Näitä ovat esimerkiksi järjestelmän dynaaminen käyttäytyminen, sen toimintaan liittyvät suuret riskitekijät sekä suoritettavien tehtävien yhteistoiminnallinen luonne. Käyttäjätutkimusmenetelmistä näitä piirteitä tukevat esimerkiksi Contextual Inquiry ja erilaiset simulaatiot. Vaatimusten kuvausmenetelmistä erityisen sopivia ovat UML:n käyttötapaukset sekä kognitiivisen työn analyysin ADS-mallit.

Tutkimuksessa havaittiin myös, että monimutkaisten järjestelmien käytettävyys on monimutkainen käsite, jonka teoreettinen määrittely vaatii syvällistä analyysia siitä, mitkä ovat järjestelmien hyvyyden kriteerit. Monimutkaisen järjestelmän käytettävyys ilmenee ihmisen toiminnan tavoitteiden täyttymisenä tiettyjen reunaehtojen vallitessa.

Savioja, Paula. Käyttäjäkeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa [User-centred methods in presenting the requirements of complex systems]. Espoo 2003. VTT Tiedotteita – Research Notes 2216. 132 p. + app. 10 p.

Keywords Cognitive Work Analysis (CWA), user-centred design, user-centered design, requirements specification, complex systems, usability

Abstract

User-centred design methods aim at improving the usability of the object being designed. High degree of usability leads to an increase in the product's effectiveness, efficiency and user satisfaction. Complex systems are used to control extensive processes, which can be characterised as dynamic, safety critical, distributed and highly automated. Examples of such systems are the various control systems in process industry and different traffic control systems. Requirement specification is a phase in a design process in which the required and significant features of the object being designed are made explicit. In achieving comprehensive usability the User-centred design methods are specifically significant in the requirements elicitation and specification phase of the design process.

This study introduces various user-centred methods that are relevant in the requirements phase of a design process. The methods are divided into two categories according to their relevance in either gathering the requirements or presenting them. The methods are presented in order to investigate which ones of them would best suit the design of so-called complex systems. The suitable methods shall be able to present the requirements of the system being designed that are relevant relating to its complexity. In this study the most thorough analysis is done on the Cognitive Work Analysis (CWA), in which the system being designed is approached through the qualities of the domain of system. This aims at creating a formative model of the system. The result of a formative design process is a system that supports the users' unique problem solving strategies and adapts to the context of use producing relevant domain information for the users' decision-making needs.

Requirements represented as the abstraction-decomposition space (ADS) of CWA and as UML's use cases are compared in the empirical part of this study. The domain of the modelling is the Ship reporting System (SRS) of the Gulf of

Finland. The requirements produced by the two modelling techniques are of a comprehensively different type. The ADS models represent requirements for the domain information that the user needs because ADS depicts what domain information is relevant for the users. This can be used in the user interface design of complex systems.

In this study it was discovered that in situations in which the object of user-centred design is a complex sociotechnical system there are some special requirements upon the user study and requirements representation of the design process. In both those phases the methods shall be able to represent the aspects of the domain that specifically make it a complex sociotechnical system. These are for example the dynamic behaviour of the system, the high degree of potential hazards and the social aspects of the work relating to it. The user study methods which support the eliciting of these aspects are for example Contextual Inquiry and different types of simulations. In representing the requirements suitable methods are Use Cases and Abstraction-decomposition spaces.

In this study it was also found out the usability of complex systems is a complex concept and the definition of which requires a profound theoretical analysis on the criteria of appropriateness of a system. The usability of a complex sociotechnical system appears in the action of the users of the system under certain conditions.

Alkusanat

VTT:n Tuotteet ja tuotanto -yksikön Tuotantojärjestelmät-alueella on pitkät perinteet erilaisten teollisuusympäristöön sijoittuvien prosessien käyttäjien toiminnan tutkimuksesta. Tutkimus on hyvin motivoitua, sillä paras tapa kehittää järjestelmien ihminen-tekniikka-rajapintaa, on tutkia niiden käyttöä todellisessa ympäristössä. Näin mahdollistetaan turvallinen ja entistä tehokkaampi teollinen tuotanto.

Tämä julkaisu perustuu TKK:n käytettävyystudkimus-professorin alla tehtyyn diplomityöhön (Savioja 2003), joka on tehty VTT:n Tuotteet ja tuotanto -yksikön perusrahoitteisena tutkimuksena. Koko VTT:n tutkimustoiminta on osittain orientoitu teknologiateemojen ympärille. Turvallisuus ja käyttövarmuus -teemaan sijoittuvassa T4METODI-projektissa on pitkän tähtäimen tavoitteena tuottaa metodologia laajojen vuorovaikutteisten järjestelmien käyttäjakeskeiseen kehittämiseen, ja näin aikaansaada turvallisempia, tuottavampia ja hyvinvointia paremmin edistäviä järjestelmiä. Tämä tutkimus palvelee METODI-projektin tavoitteita esiselvityksen tavoin.

Haluan kiittää kaikkia tutkimusta VTT:n puolesta tutkimusta ohjanneita, ryhmäpäällikkö tekn. tri Olli Ventää, erikoistutkija, dipl.ins. Teemu Tommilaa sekä erityisesti erikoistutkija fil. tri Leena Norrosta. Tutkija, kapt. Sanna Sonnista kiitän laajan merenkulun asiantuntemuksen jakamisesta. Alkuperäistä diplomityötä valvonutta TKK:n käyttöliittymien ja käytettävyyden professori Marko Niemistä haluan kiittää diplomityön hyvästä ohjauksesta.

Otaniemessä 2.6.2003

Paula Savioja

Sisällysluettelo

Tiivistelmä	3
Abstract	5
Alkusanat	7
1. Johdanto	13
1.1 Teoreettinen tausta	13
1.1.1 Käyttäjäkeskeinen suunnittelu	13
1.1.2 Järjestelmien monimutkaisuus	16
1.1.3 Vaatimusten määrittely	19
1.2 Tutkimuskysymykset ja työn tavoitteet	22
1.3 Työn rakenne	23
2. Kognitiivinen työn analyysi	24
2.1 CWA:n teoria	24
2.1.1 Kolme tapaa mallintaa järjestelmän toimintaa	24
2.1.1.1 Normatiivinen mallinnus	25
2.1.1.2 Deskriptiivinen mallinnus	26
2.1.1.3 Formatiivinen mallinnus	26
2.1.2 Ekologinen suunnittelu	27
2.2 CWA prosessina	31
2.2.1 Ensimmäinen vaihe: Työkohteen analyysi	31
2.2.2 Toinen vaihe: Hallintatehtävien analyysi	32
2.2.3 Kolmas vaihe: Strategian analyysi	32
2.2.4 Neljäs vaihe: Organisaation ja yhteistyön analyysi	33
2.2.5 Viides vaihe: Työntekijän osaamisen analyysi	33
2.3 CWA:n rooli vaatimusten määrittelyssä	34
3. Käyttäjätutkimukset	36
3.1 Käyttäjätiedon tarpeet	36
3.2 Käyttäjätutkimusmenetelmät	37
3.2.1 Epäsuorat tutkimusmenetelmät	37
3.2.1.1 Haastattelut	37
3.2.1.2 Kyselyt	38

3.2.1.3	Ryhmämenetelmät	38
3.2.2	Suorat menetelmät.....	39
3.2.2.1	Havainnointi	39
3.2.2.2	Etnografinen tutkimus.....	40
3.2.2.3	Ääneen ajattelu	41
3.2.2.4	Contextual Inquiry	42
3.2.2.5	Simulaatiot.....	44
3.3	Käyttäjätutkimusmenetelmän valinta monimutkaisen järjestelmän suunnittelussa	45
3.3.1	Tutkimusmenetelmän valinnan perusteita.....	45
3.3.2	Monimutkaisen työn tutkimukseen sopivat menetelmät	46
4.	Vaatimusten muodostaminen ja esittäminen	49
4.1	Suunniteltava järjestelmä kuvataan vaatimusten avulla	49
4.1.1	Millainen on hyvä kuvaus?	49
4.1.2	Vaatimusten määrittely on kommunikointia	51
4.1.3	Vaatimusten luokittelu	52
4.1.3.1	Toiminnallisuuteen perustuva luokittelu	52
4.1.3.2	Laatuominaisuuksien mukainen luokittelu	53
4.1.3.3	Prioriteettiin perustuva luokittelu	53
4.1.3.4	Informaatiojärjestelmän rooleihin perustuva luokittelu....	53
4.1.4	Vaatimukset validoinnin tukena.....	55
4.2	Vaatimusten esittäminen	56
4.2.1	Epäformaalit mallit vaatimuksista.....	56
4.2.1.1	Vaatimusluettelo	56
4.2.1.2	Skenaariot	58
4.2.1.3	Storyboardit eli kuvakäsikirjoitukset	58
4.2.1.4	Prototyypit	59
4.2.2	Puoliformaalit mallit vaatimuksista	61
4.2.2.1	Contextual Designin kuvaustapoja	62
4.2.2.2	Unified Modelling Languagen kuvaustapoja.....	72
4.2.2.3	Kognitiivisen työn analyysin mallinnustekniikat.....	76
4.2.3	Formaalit mallit vaatimuksista	81
4.3	Vaatimusten muodostaminen käyttäjakeskeisenä prosessina.....	82
4.3.1	Käyttäjätarpeiden dokumentointi	83
4.3.2	Contextual Designin prosessi	85
4.3.3	Ihmiskeskeinen turvallisuuslähtöinen prosessi	86

4.3.4	Vaatimusten määrittely ISO 13407:ssä	88
4.4	Vaatimusten kuvausmenetelmän valinta	90
5.	Tapaustutkimus: Alusten ilmoittautumisjärjestelmän vaatimusten määrittely	94
5.1	Taustaa: Vessel Traffic Services	94
5.1.1	VTS:n tehtävät	95
5.1.2	VTS:n kehitys	96
5.2	Suomenlahden SRS-järjestelmä	97
5.3	Tapaustutkimuksen tavoitteet	97
5.4	Tutkimusmenetelmät ja tutkimuksen kulku.....	98
5.4.1	Käyttäjätutkimus	98
5.4.1.1	Contextual Inquiry -havainnointi.....	98
5.4.1.2	Asiantuntijakeskustelujen havainnointi	99
5.4.1.3	Simulointi	100
5.4.1.4	Empiirinen aineisto	101
5.4.2	Vaatimusten kuvaaminen	101
5.4.3	Kuvausten validointi	102
5.4.4	Kuvausten vertailu	102
5.5	Tulokset	102
5.5.1	Järjestelmävaatimukset	103
5.5.1.1	Vaatimukset ADS:lla kuvattuna	103
5.5.1.2	Vaatimukset käyttötapauksina	109
5.5.2	Arviot vaatimuksista ja kuvausmenetelmistä.....	115
5.5.2.1	Asiantuntijahaastattelu: vaatimusten validointi	115
5.5.2.2	Vaatimusten kuvaustapojen vertailu.....	116
5.6	Tapaustutkimuksen johtopäätökset	117
6.	Yhteenveto ja pohdinta	119
6.1	Vastaukset tutkimuskysymyksiin	119
6.1.1	Ensimmäinen tutkimuskysymys.....	119
6.1.2	Toinen tutkimuskysymys	120
6.1.3	Kolmas tutkimuskysymys	121
6.2	Monimutkaisen järjestelmän suunnitteluprosessin käyttäjäkeskeisyys	123
6.2.1	CWA monimutkaisen työn analyysina.....	124
6.2.2	Käyttäjätutkimuksen näkökulmat.....	124
6.2.3	Vaatimusten määrittely	125
6.3	Tutkimuksen puutteet	126

6.4	Jatkotutkimusmahdollisuuksia	127
	Lähdeluettelo	128
Liitteet		
	Liite A: Standardin ISO 13407 periaatteet	
	Liite B: Standardin ISO 11064 periaatteet	
	Liite C: ISO 11064 -prosessin vaiheet	
	Liite D: Alustavat vaatimukset	

1. Johdanto

Tietoteknisten järjestelmien sekä niillä hallittavien kokonaisuuksien laajuus kasvaa jatkuvasti. Monimutkaisuuden lisääntyminen on herättänyt tarpeen kiinnittää huomiota järjestelmien käytettävyyden tasoon. Yleisesti käytettävyydellä tarkoitetaan jonkin tuotteen tai välineen sopivuutta sille tarkoitettuun tehtävään. Kasvaneita käytettävyyksivaatimuksia aiheuttavat odotukset tehokkaammasta tuotannosta sekä työntekijöiden paremmista työolosuhteista. Käytettävyys voi vaikuttaa myös yleiseen turvallisuuteen pienempänä inhimillisen virheen todennäköisyytenä. Tässä työssä tarkastellaan menetelmiä, joilla suunnitteluvaiheessa voidaan vaikuttaa monimutkaisten järjestelmien käytettävyyteen.

1.1 Teoreettinen tausta

Tämän tutkimuksen aihe on monimutkaisten järjestelmien käyttäjakeskeinen vaatimusmäärittely. Työ liittyy yhteen kolme eri käsitettä: käyttäjakeskeisen suunnittelun, järjestelmien monimutkaisuuden ja vaatimusten määrittelyn. Näillä kaikilla osa-alueilla on oma tutkimusperinteensä ja problematiikkansa. Tässä johdantoluvussa esitellään nämä käsitteet sekä niiden yhteys tutkimuskysymyksiin.

1.1.1 Käyttäjakeskeinen suunnittelu

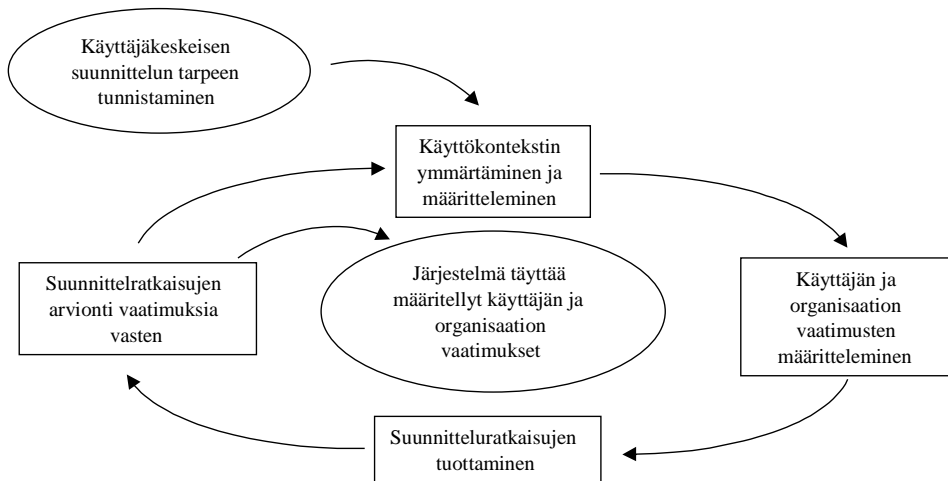
Käyttäjakeskeisellä suunnittelulla pyritään suunniteltavien tuotteiden käytettävyyssominaisuuksien parantamiseen. Käytettävyys on yksi standardin ISO/IEC 9126 (1991) määrittelemistä ohjelmistojen laatuominaisuuksista. ISO DIS 9241-11 (1998) määrittelee käytettävyyden tarkoittavan ”hyödyllisyyttä, tehokkuutta ja käytön helppoutta, jolla määrätty käyttäjä voivat saavuttaa määrättyjä tavoitteita määrättyssä ympäristössä”.

Yksittäisistä käytettävyyden määrittelijöistä tunnetuimpia on Jakob Nielsen (1993), jonka mukaan tuotteen käytettävyys tarkoittaa sen käytön opittavuutta, tehokkuutta, muistettavuutta ja virheettömyyttä sekä käyttäjän tyytyväisyyttä. Tämä määritelmä pureutuu hyvin tietokoneohjelmistojen käyttöliittymien testattavissa oleviin käytettävyyssominaisuuksiin. Se ei kuitenkaan ota huomioon käytettävyyttä laajemman ihmisen ja järjestelmän välisen vuorovaikutuksen laatuominaisuutena.

Keinonen (2000) määrittelee käytettävyyden kuluttajan valintoihin liittyvän päätöksenteon kannalta. Hänen mukaansa käytettävyyden kriteerit ovat: toiminnallisuus, loogisuus, informaation esitystapa, käyttöohjeet, hyödyllisyys, helpokäyttöisyys ja tunteisiin vaikuttavuus. Vaikka määritelmä on kehitetty kuluttajatuotteiden tutkimuksen yhteydessä, tuntuvat kriteerit intuitiivisesti täydentävän käytettävyyden määritelmää myös monimutkaisempien järjestelmien osalta. Varsinkin kriteerit toiminnallisuus, loogisuus ja informaation esitystapa ovat varmasti merkityksellisiä myös teollisuuden erilaisille järjestelmille.

Kuutti (2000) on referoinut käytettävyydetutkimuksen historiallista kehitystä. Hän kertoo, että käyttöliittymätutkimus sai alkunsa toisen maailmansodan aikaan ergonomisena tutkimuksena, jolloin sotateknologian kehityksessä havaittiin ihmisen suorituskyvyn rajoitteet uusien laitteiden käyttöönotolle. Kognitiivinen taso tuli mukaan tutkimuskenttään 1970-luvulla tietokoneiden näyttöpäätteen yleistyessä. Tuolloin ihmistä käsiteltiin tiedon prosessoijana, jonka kognitiiviset kyvyt oli otettava huomioon suunnittelussa. 1980-luvulla käyttöliittymätutkimuksessa teoreettisten mallien merkitys väheni, ja pääosan sai käytännön yrityksen ja erehdyksen kautta tapahtuva insinöörimäinen suunnittelutyö. Tälle aikakaudelle siijoittuu graafisten käyttöliittymien logiikan ja esimerkiksi ikkunointitapojen kehittyminen. 1990-luvulla vakiintui käsitys siitä, että hyvään käytettävyyteen voidaan päästä vain tuomalla käyttäjät mukaan suunnitteluun. (Kuutti 2000.) Näin on päästy moderniin käsitykseen käyttäjäkeskeisestä suunnittelusta, jonka keskeisiä tunnusmerkkejä käyttäjien osallistuttaminen suunnitteluprosessiin on.

Käyttäjäkeskeistä tuotekehitystä käsittelee kaksi tämän työn kannalta merkittävää standardia. Standardi ISO 13407 (1999) määrittelee interaktiivisten järjestelmien käyttäjäkeskeisen suunnitteluprosessin ja ISO 11064 (2000) valvontakeskusten ergonomisen suunnitteluprosessin. Molemmissa standardeissa on määritelty yleiset periaatteet, joiden ympärille suunnitteluprosessin pitäisi rakentua (Liite A, Liite B). Molemmat määrittelevät suunnitteluprosessin olevan luonteeltaan iteratiivinen ja koostuvan erillisistä vaiheista (Kuva 1, Liite C). Vaiheiden mukaan prosessi etenee käyttäjätutkimuksen, vaatimusten määrittelyn, suunnitteluratkaisujen tuottamisen ja arvioinnin kautta käytettävyydeltään korkeatasoiseen lopputulokseen. Molemmat standardit määrittelevät yksittäisiin vaiheisiin liittyvät toiminnot.



Kuva 1. Standardin mukaisen käyttäjakeskeisen tuotekehitysprosessin eri vaiheiden riippuvuussuhteet (ISO 1999).

Käyttäjakeskeisen suunnittelun prosessimalli riippuu aina suunnittelua toteuttavan organisaation kulttuurista ja tavoitteista. Erilaisia malleja on olemassa erittäin monia. Useimmista (esim. Jokela 2001, Beyer & Holtzblatt 1998, Kreitzberg 1999) löytyy kuitenkin jonkinlainen yhteys edellä mainittuihin standardeihin.

Vredenburg ym. (2002) ovat selvittäneet, kuinka yleistä käyttäjakeskeisten prosessien käyttö tämän päivän yrityksissä on. Vielä 1990-luvun alkupuolella käyttäjakeskeisen suunnittelun metodiikka oli käytännön kehitystyössä vähäistä. Tähän olivat syynä niin organisatoriset kuin teknisetkin tekijät. Käyttäjakeskeiset menetelmät koettiin monimutkaisiksi, aikaa vieviksi ja kalliiksi toteuttaa.

Vredenburgin ym. (2002) tutkimuksen mukaan yritysten välillä on tällä hetkellä suuria eroja käytettävyystekniikoiden hallinnassa ja käyttöasteessa. Käyttäjakeskeisten suunnittelumenetelmien käytön katsotaan kuitenkin aina parantavan lopputuotteen käyttökelpoisuutta. (Vredenburg ym. 2002)

1.1.2 Järjestelmien monimutkaisuus

Tämä työ käsittelee käyttäjakeskeistä tuotekehitystä, jossa kehitettävää kohdetta voidaan luonnehtia monimutkaiseksi järjestelmäksi. Monimutkaisuus tarkoittaa Oxfordin sanakirjan (1995) määritelmän mukaan sitä, että jokin ilmiö on vaikea ymmärtää tai selittää siihen liittyvien erilaisten näkökulmien takia. Saman lähteen mukaan järjestelmä on monimutkainen silloin, kun se rakentuu useista osista, jotka liittyvät toisiinsa jonkin määritellyn kaavan mukaan.

Miller (2000) huomauttaa monimutkaisuuden olevan suhteellinen käsite. Tilanne tai järjestelmä on monimutkainen silloin, kun tietty yksilö pitää sitä vaikeasti ymmärrettävänä tai hänellä on vaikeuksia muuten käsitellä sitä. Sama tilanne tai järjestelmä voi jonkun toisen yksilön kannalta olla yksinkertainen.

Woods (1988) on määritellyt monimutkaisuuden koostuvan dynaamisuudesta, osien ja niiden välisten suhteiden määrästä sekä epävarmuuden ja riskitekijöiden korkeasta määrästä. Hän toteaa, että tilanteeseen tai järjestelmään, joka on monimutkainen, liittyy jokaisen edellä mainitun dimension korkea taso. Samansuuntaisiin päätelmiin on tullut myös Miller (2000), joka jakaa järjestelmien monimutkaisuuden kolmeen eri alaluokkaan. Nämä ovat komponentti-, suhteellinen ja käyttäytymismonimutkaisuus. Komponenttimonimutkaisuus tarkoittaa järjestelmän rakentumista osista. Mitä enemmän aliosia järjestelmä sisältää, sitä monimutkaisempi se on. Suhteellisella monimutkaisuudella tarkoitetaan sitä, miten monella eri tavalla järjestelmän eri osat voivat vaikuttaa toisiinsa. Tällöin esimerkiksi sata diodia pakkauslaatikossa on vähemmän monimutkainen järjestelmä kuin sata diodia kytkettynä erilaisten piirien kautta yhteen virtalähteeseen. Käyttäytymismonimutkaisuus tarkoittaa järjestelmän eri tilojen määrää. Käsite on yhteydessä ennakoimattomuuden käsitteeseen. Usein järjestelmä, jonka erilaisten tilojen määrä on suuri, mielletään näiden selkeistä syy- ja seuraussuhteista huolimatta monimutkaiseksi, sillä tilojen suuren määrän takia ihmisen on vaikea ennustaa järjestelmän käyttäytymistä. (Miller 2000)

Vicente (1999) on määritellyt ominaisuuksia (Taulukko 1), jotka tekevät järjestelmästä monimutkaisen sosioteknisen systeemin. Ominaisuudet rajaavat monimutkaisen sosioteknisen järjestelmän olevan tekninen järjestelmä, jolla yhteistoiminnan menetelmillä hallitaan laajaa, dynaamista ja epävarmuustekijöitä sisältävää kohdetta.

Taulukko 1. Monimutkaisen sosioteknisen järjestelmän ominaisuudet (Vicente 1999).

Laaja-alaisuus	<ul style="list-style-type: none"> • Järjestelmän avulla hallittava kokonaisuus on laaja. • Toimintaan vaikuttavat monet eri osatekijät. • Osatekijöiden käyttäytymistä mahdotonta ennakoida järjestelmän suunnitteluvaiheessa.
Sosiaalisuus	<ul style="list-style-type: none"> • Vaatii eri käyttäjiltä yhteistyötä ja kommunikointia. • Käyttäjien määrä jopa satoja tai tuhansia.
Heterogeeniset perspektiivit	<ul style="list-style-type: none"> • Eri käyttäjillä on erilainen suhde järjestelmään. • Eri käyttäjillä on erilaiset tarpeet ja tavoitteet, joita he pyrkivät saman järjestelmän avulla toteuttamaan.
Hajautettu	<ul style="list-style-type: none"> • Käyttäjät hajautettu eri toimipisteisiin, jotka voivat sijaita eri rakennuksissa, paikkakunnilla tai jopa mantereilla. • Ohjausjärjestelmän kohde voi olla fyysisesti hajautettu.
Dynaaminen	<ul style="list-style-type: none"> • Järjestelmän tila muuttuu ajan funktiona. • Käyttäjät joutuvat ennakoimaan muutoksia.
Suuret riskitekijät	<ul style="list-style-type: none"> • Mahdollisten järjestelmän virhetilanteista johtuvien onnettomuuksien seuraukset ovat erittäin vakavia.
Rakentuu alijärjestelmistä	<ul style="list-style-type: none"> • Järjestelmä rakentuu alijärjestelmistä. • Alijärjestelmillä on omat käyttäjänsä.
Automatisoitu	<ul style="list-style-type: none"> • Järjestelmään liittyy paljon korkean tason automatiikkaa. • Käyttäjät toimivat ongelmanratkaisijoina tilanteissa, joissa automatiikka pettää
Epävarmuus	<ul style="list-style-type: none"> • Järjestelmästä saatuun tietoon liittyy epävarmuustekijöitä. • Epävarmuustekijät vaihtelevat ulkoisista tekijöistä riippuen.
Välittyntä vuorovaikutus	<ul style="list-style-type: none"> • Käyttäjän kokonaistavoitteiden täyttymistä ei aina voi havaita suoraan järjestelmän tilan avulla.
Häiriöt	<ul style="list-style-type: none"> • Järjestelmä voi joutua tilaan, jota suunnittelija ei ole voinut ennakoita.

Kognitiivinen työn analyysi (engl. Cognitive Work Analysis, CWA) on Vicenten (1999) kehittämä metodologia monimutkaisten sosioteknisten järjestelmien suunnitteluun. Metodologiaan kuuluu erilaisia tapoja mallintaa järjestelmävaatimuksia sekä teoreettinen viitekehys suunnittelun toteuttamiselle. Kognitiivinen työn analyysi esitellään tarkemmin tämän työn toisessa luvussa.

Esimerkiksi prosessiteollisuudessa käytettävä ohjausjärjestelmä on Vicenten (1999) ja Millerin (2000) määritelmät toteuttava monimutkainen järjestelmä. Prosessiteollisuuden haasteita ovat tänä päivänä uudet vaikeammat tuotteet, kiristyneet laatu- ja tehokkuusvaatimukset, kierrätys ja automaation mahdollisuuksien hyödyntäminen. Zuboff (1988) tutki jo 1980-luvulla automatisoinnin vaikutuksia työhön. Hänen mukaansa prosessiteollisuudelle on tyypillistä, että automaatiota ei varsinaisesti ole käytetty työn yksinkertaistamiseen ja osittamiseen, vaan automaatio on myös lisännyt työn vaativuutta. Automatisoinnin myötä fyysinen yhteys ohjattavaan prosessiin katoaa helposti. Tämä vaikuttaa osaltaan työn abstrahoitumiseen ja siten sen vaativuuden kasvamiseen. (Zuboff 1988) Kallelan (1996) mukaan käyttäjiä kannattaa kuitenkin osallistuttaa automaation suunnitteluun. Hän toteaa käyttäjakeskeisen automaatio suunnittelun vaikuttavan positiivisesti kehitettävän järjestelmän ominaisuuksiin. Voidaan siis sanoa, että kehitettävän järjestelmän monimutkaisuudesta huolimatta käyttäjakeskeisellä suunnittelulla on mahdollista parantaa lopputuotetta.

Monimutkaisen järjestelmän käyttäjakeskeinen kehittäminen asettaa prosessin käyttäjätutkimusvaiheelle joitain haasteita. Tutkimusmenetelmän on oltava sellainen, että se nostaa esiin niitä käyttäjän toiminnan piirteitä, jotka ovat työn kannalta merkityksellisiä. Tässä tutkimuksessa monimutkaisella järjestelmällä tarkoitetaan yleensä turvallisuuskriittisen prosessin reaaliaikaista ohjausjärjestelmää. Tämälkaltaisten järjestelmien käyttäjät ovat tyypillisesti oman alansa asiantuntijoita ja heillä on takana perusteellinen koulutus tehtäviensä hoitamiseen. Tämän asiantuntemuksen esiin saaminen on käyttäjätutkimuksen suurimpia haasteita, sillä tutkija ei mitenkään voi saavuttaa samanlaista asiantuntemuksen tasoa, joka käyttäjillä on. Tässä on huomattava ero esimerkiksi kuluttajatuotteiden käyttäjakeskeiseen kehittämiseen, jossa suunnittelijat voivat itsekin olla tuotteen potentiaalisia käyttäjiä.

1.1.3 Vaatimusten määrittely

Vaatimusten määrittelyllä pyritään tuomaan julki suunniteltavan tuotteen keskeiset ominaisuudet. Suunnitteluprosessi alkaa sillä, että päätetään tai selvitetään mitä oikeastaan ollaan tekemässä. Vaatimusten määrittelyn käytännöt ovat alunperin muodostuneet perinteisten insinöörialojen, kuten arkkitehtuurin ja rakennesuunnittelun piirissä, mutta nykyisin se mielletään yhdeksi ohjelmistotuotannon menetelmäksi. Schach (2002) määrittelee ohjelmistotuotannon tarkoittavan ohjelmistojen suunnittelun, toteuttamisen ja ylläpidon hallittuja menetelmiä, joilla pyritään kustannustehokkaasti toteuttamaan laadullisesti hyvä sekä käyttäjän kohtuulliset toiveet ja odotukset täyttävä lopputulos.

Ohjelmistoja voidaan tuottaa erilaisilla prosessimalleilla. Malleille on yhteistä se, että kehitysprosessi on jaettu linkaaren mukaan yksittäisiin vaiheisiin. Vaiheet on nimetty malleissa hieman eri tavoin, mutta ne noudattavat suurin piirtein kaavaa: määrittely, suunnittelu, toteutus, käyttöönotto ja ylläpito. Samoin vaiheiden väliset panos-tuotossuhteet vaihtelevat ja toisiin malleihin liittyy enemmän ja pitempiä iterointikierroksia. Oleellista ja kaikille eri malleille yhteistä on, että ohjelmistoprojektin toteutus viedään läpi suunnitelmallisesti. (Schach 2002)

Käytettävästä ohjelmiston elinkaarimallista riippumatta vaatimusten määrittely on aina ohjelmistokehitysprojektin ensimmäinen osa (Johnson 2002, s. 14). Hyvä käytäntö järjestelmien kehittämisessä perustuu vaatimusten määrittelyyn ennen varsinaisen suunnittelun aloittamista, jolloin toteutettavan järjestelmän ominaisuudet vastaavat ennalta määriteltyihin reaali maailman tarpeisiin. Parnasin (2000) mukaan nimenomaan vaatimusten täydellinen ja tarkka kuvaaminen on ainoa tapa mahdollistaa – joka tapauksessa jo olemassa olevien – vaatimusten toteutuminen järjestelmässä. Samoin hän toteaa, että epätäydellisellä tai epäkonsistentilla vaatimusten määrittelyllä helposti johdetaan harhaan järjestelmän suunnittelijoita ja toteuttajia.

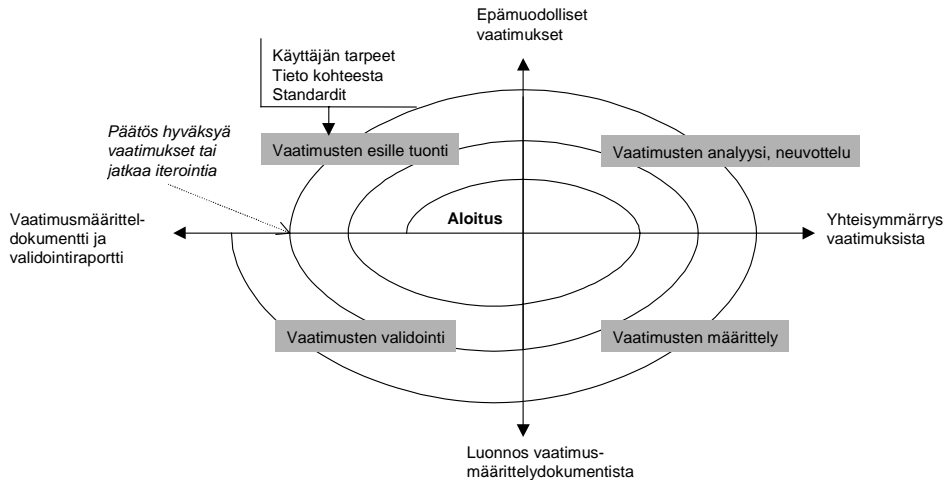
Vaatimusten määrittely (engl. requirements engineering) toimintana käsittää vaatimusten hankintaa, jalostamista ja tarkistamista. Parnasin (2000) mukaan ohjelmistokehityksen vaatimusten määrittely -vaiheen tavoitteena on:

- päättää, mitä tehdään, ennen kuin tekeminen aloitetaan
- tuottaa organisoitu lähdedokumentti järjestelmän toteuttajalle
- tuottaa tarvittava taustatieto laadunvarmistajille
- tuoda julki järjestelmän toteutuksen rajoitteet ja reunaehdot.

Vaatimusten määrittely voidaan kuvata prosessina (Kuva 2) johon kuuluu erillisiä vaiheita. Sawyerin ja Kotonyan (2001) mallissa painottuu vaatimusten määrittelyn iteratiivinen luonne. Ensimmäisessä vaiheessa nostetaan esiin olemassa olevia suunnittelun kohdetta koskevia vaatimuksia. Toisessa vaiheessa pyritään suunnittelun eri osapuolten välillä muodostamaan yhteisymmärrys kohteen vaatimuksista. Neljännessä vaiheessa tuotetaan vaatimusmäärittelydokumentti, joka lopuksi validoidaan sovitulla menetelmällä. Havaitut virheet, mahdolliset tarpeettomat vaatimukset sekä vaatimuksiin sisältyvät epäloogisuudet korjataan tarvittaessa uuden iteraatiokierroksen aikana. (esim. Sawyer & Kotonya 2001, Schach 2002)

Sutcliffen (2002) mukaan vaatimusmäärittelyvaihe kattaa tyypillisesti noin 10–15 % koko tuotekehityksen aikana syntyvistä kustannuksista, mutta vaatimusmäärittelyssä tehdyistä virheistä syntyvät kustannukset ovat oleellisesti suuremmat. Mitä pidempään jokin yksittäinen virhe säilyy mukana kehitysprosessissa, sitä kalliimmaksi sen korjaaminen muodostuu. (Sutcliffe 2002)

Monimutkaisten järjestelmien vaatimusten määrittelyssä on otettava erityisen tarkkaan huomioon turvallisuuteen liittyvät näkökohdat. Leen ym. (2002) mukaan 90 % turvallisuuteen liittyvistä ratkaisuista tehdään jo järjestelmäsuunnittelun alkuvaiheissa. Asiantuntijoiden tekemä määrittelyn tarkastaminen on yksi tehokkaimpia ja vähiten kustannuksia tuottavia tapoja parantaa järjestelmien turvallisuusominaisuuksia. Lee ym. (2002) toteaa vaatimusmäärittelyn suuresta merkityksestä vielä, että suurin osa viimeisen kahdenkymmenen vuoden aikana tapahtuneista ohjelmistokomponenttien väärästä käyttäytymisestä johtuvista onnettomuuksista on voitu jäljittää takaisin virheelliseen tai epätäydelliseen vaatimusten määrittelyyn.



Kuva 2. Malli vaatimusten määrittelyn prosessista (Sawyer & Kotonya 2001).

Käyttäjakeskeisiä elementtejä voidaan tuoda mukaan tuotekehitysprosessin eri vaiheisiin. Esimerkiksi testausvaiheessa voidaan tehdä käytettävyystestejä. Kujalan (2002) mukaan käyttäjakeskeiset menetelmät ovat kuitenkin erityisen hyödyllisiä tuotekehitysprosessin alkuvaiheissa, joihin vaatimusmäärittelyvaihe kuuluu. Alkuvaiheessa tuotteen määrittelyyn sulautettu tieto käyttäjän tarpeista ja toiminnasta on siis merkittävä lopputuotteen käytettävyyden kannalta. Kujalan (2002) mukaan vaatimusmäärittelyssä tarvittavaa käyttäjätietoa ei kuitenkaan voida suoraan kysyä käyttäjiltä. Hänen mukaansa käyttäjien on vaikea artikuloida työhönsä liittyvää osaamistaan, joten vaatimustenhankintaan on käytettävä muita menetelmiä.

Erilaisilla käyttäjätiedon keräysmenetelmillä on saatavissa valtava määrä loppukäyttäjiä koskevaa raakatietoa, jota kutsutaan käyttäjätiedoksi. Tiedon määrä saattaa kuitenkin olla niin suuri, että sen käsittely muodostuu ongelmaksi. Tämä tieto on jalostettava järjestelmän suunnittelun kannalta merkitykselliseksi informaatioksi jollain menetelmällä. Tiedon jalostamisprosessi, jossa vaatimukset vähitellen muotoutuvat, korostaa vaatimusmäärittelyn kommunikatiivista puolta. Vaatimusten määrittelyssä jonkin suunnittelun osapuolen omaksuma tieto käyttäjän tarpeista on pystyttävä eksplikoimaan niin, että se on koko suunnitteluryhmän käytettävissä. Tiedon jalostamisprosessissa joudutaan tekemään joskus suuriakin käsitteellisiä hyppäyksiä, kun mietitään, minkälaiset vaatimukset par-

haiten toteuttaisivat käyttäjän tarpeet täyttävän järjestelmän. Vaatimusten kuvaustavalla on merkitystä käsitteellisen kuilun ylittämässä. Hyvällä kuvaustavalla tätä kuilua voidaan kaventaa ja samalla myös parantaa suunnitteluryhmän sisäistä kommunikointia.

Suunniteltavan järjestelmän monimutkaisuus asettaa myös vaatimuksia kuvaustavalle, sillä sen myötä myös kuvauksen monimutkaisuus helposti kasvaa. Tällöin kuvaustavan on mahdollistettava mahdollisimman yksiselitteinen, mutta samalla ilmaisuvoimainen tapa mallintaa järjestelmän merkittäviä ominaisuuksia.

Käyttäjakeskeisen tuotekehitysprosessin vaatimusmäärittelyvaihe on voitava validoida loppukäyttäjien kanssa. Myös tämä asettaa erityisiä vaatimuksia vaatimusten kuvaustavalle. Vaatimukset on esitettävä tavalla, jonka käyttäjä pystyy ymmärtämään.

1.2 Tutkimuskysymykset ja työn tavoitteet

Tämän työn tavoitteena on tutkia, millainen kokonaisuus muodostuu monimutkaisten järjestelmien käyttäjakeskeisen suunnittelun vaatimusten määrittelystä.

Tutkimukselle on muotoiltu seuraava ylätasoinen tutkimuskysymys:

Miten monimutkaisen järjestelmän suunnitteluprosessin vaatimusten määrittelyvaihe voidaan toteuttaa käyttäjakeskeisesti?

Yllä oleva tutkimuskysymys on jaoteltu alakysymyksiin. Näihin vastaamalla pyritään muodostamaan kokonaiskuva käyttäjakeskeisen suunnitteluprosessin alkupään menetelmistä

Tarkennetut tutkimuskysymykset ovat:

- 1. Mitkä käyttäjätutkimusmenetelmät soveltuvat monimutkaisen järjestelmän tiedonkeräysmenetelmiksi?**
- 2. Mitkä vaatimusten kuvausmenetelmät tukevat monimutkaisten järjestelmien ominaispiirteiden mallintamista?**

3. Millaisia vaatimuksia Kognitiivisen työn analyysin (Vicente 1999) mallinnustekniikat nostavat esiin?

Työn tavoitteena on siis selvittää monimutkaisten järjestelmien käyttäjäkeskeisen suunnittelun menetelmiä. Työ keskittyy tyypillisen suunnitteluprosessin ensimmäiseen vaiheeseen eli vaatimusten määrittelyyn. Tavoitteena on selvittää tekniikoita, joilla monimutkaisten järjestelmien vaatimuksia voidaan käyttäjäkeskeisesti tuottaa ja mallintaa. Käyttäjäkeskeisyydellä tarkoitetaan tässä yhteydessä sitä, että vaatimukset perustuvat todelliselle käyttäjätutkimukselle, niiden perustana on tietoa käyttäjän tarpeista ja että niitä voidaan arvioida käyttäjien kanssa.

1.3 Työn rakenne

Työ jakautuu kuuteen eri lukuun. Luvut 2–4 muodostavat työn kirjallisuusosan. Ne on jäsenneilty siten, että jokaisen luvun viimeinen kappale sisältää kirjoittajan omaa pohdintaa sekä yhteenvedon edellä käsitellyistä aiheista. Kirjallisuustutkimuksen avulla pyritään esittämään tietoa, johon ensimmäisen ja toisen tutkimuskysymyksen vastaukset perustuvat. Viides luku muodostaa työn käytännön osan. Siinä selostetaan tapaustutkimuksen kohde, tutkimuksen kulku sekä siinä saavutetut tulokset ja niistä tehdyt johtopäätökset. Viidennen luvun avulla vastataan kolmanteen tutkimuskysymykseen. Kuudennessa luvussa esitetään yksittäisten tutkimuskysymysten vastaukset sekä tutkimuksen kokonaisuuden pohdinta ja yhteenveto.

2. Kognitiivinen työn analyysi

Luvussa esitellään Kim Vicenten kehittämä monimutkaisten järjestelmien suunnittelun metodologinen viitekehys Cognitive Work Analysis (CWA), eli Kognitiivinen työn analyysi. CWA on prosessimainen työn analysointitekniikka, jota voidaan soveltaa käyttäjäkeskeisen suunnittelun eri vaiheissa. Se asettaa vaatimuksia sekä suunnitteluprosessille, että lopputuotteelle.

Tämän luvun kohdat 2.1 ja 2.2 perustuvat Kim Vicenten (1999) kirjan ”Cognitive Work Analyses – Towards Safe, Productive, and Healthy Computer-Based Work” kolmeen ensimmäiseen osaan. Vicenten tekstiä on luettu suunnittelun näkökulmasta. Luvun lopussa, kohdassa 2.3, esitetään kirjoittajan omat pohdinnat Vicenten teoriasta.

2.1 CWA:n teoria

Käyttäjäkeskeiset suunnittelumenetelmät (esim. Beyer & Holtzblatt 1998, Jokela 2001) lähestyvät suunniteltavaa järjestelmää sillä suoritettavien tehtävien kautta. Kim Vicente toteaa tällaisen lähestymistavan olevan tietyissä tilanteissa puutteellinen. Tämä tulee ilmi etenkin suunniteltaessa järjestelmiä, jotka ovat luonteeltaan monimutkaisia ja sosioteknisiä. Ominaisuudet, jotka tekevät järjestelmästä tällaisen, esitellään tämän julkaisun Johdanto-luvussa.

2.1.1 Kolme tapaa mallintaa järjestelmän toimintaa

Informaatiojärjestelmän suunnittelu perustuu aina jonkinlaiseen analyysiin tilanteesta, jossa järjestelmää käytetään sekä tarpeista, joihin järjestelmän tulisi vastata. Vicente on pohtinut millainen analyysitekniikka olisi hyödyllisin ja tarkin monimutkaisten sosioteknisten järjestelmien suunnittelussa. Erilaisten mallinnustekniikoiden jako kolmeen alaluokkaan on alunperin peräisin Jens Rasmussenilta (ks. Vicente 1999, s. 61–62), joka jäsensi monella eri tieteenalalla tapahtunutta mallinnuksen evoluutiota. Vicente laajentaa Rasmussenin mallinnusta koskevaa teoriaa koskemaan myös työn analyysia ja sen tuloksena tehtäviä malleja.

2.1.1.1 Normatiivinen mallinnus

Normatiivisessa mallinnuksessa pyritään kuvaamaan suunniteltava järjestelmä sen tavoitellun käyttäytymisen avulla, eli järjestelmästä mallinnetaan se, miten sen tulisi toimia suunnitelluissa tilanteissa. Tällainen malli perustuu yleensä tehtäväänalyysin perusteella tehtyihin päätelmiin.

Normatiivinen analyysi on Vicenten mukaan puutteellinen, sillä se ei ota tarpeeksi huomioon ympäristön avoimelle systeemille aiheuttamia vaihteluita. Normatiivinen malli tekee helposti myös epärealistisia oletuksia ihmisten tekemästä työstä. Normatiivinen malli perustuu tyypillisesti ideaaliseen tilanteeseen, jossa kontekstisidonnaiset työtilanteen ja -tehtävän vaihtelut jäävät huomiotta.

Esimerkki normatiivisesta lähestymistavasta työn analyysiin on perinteinen tehtäväänalyysi, joka perustuu sekvenssimäiseen osatehtävien suoritusproseduurin mallintamiseen. Tästä seuraa ongelmia, sillä määritelmän mukaan monimutkaisten sosioteknisten järjestelmien käyttäjät joutuvat tilanteisiin, joita ei suunnitteluvaiheessa ole pystytty ennakoimaan. Tällöin tehtäväänalyysiin perustuva suoritusproseduuri katkeaa, ja järjestelmä ajautuu virhetilaan. Tyypillisesti erilaiset häiriöt ja onnettomuudet johtuvat juuri tilanteista, joita ei ole suunnitteluvaiheessa osattu tunnistaa ja siten ennakoida, jolloin niille ei ole osattu suunnitella suoritusproseduureja. Jos työntekijät ovat tottuneet tekemään työnsä ohjeistavassa järjestelmässä, jossa työtehtävien oikeaoppinen suoritus on määritelty toimenpiteiden sekvenssinä tai muina toimintaohjeina, on heidän mahdotonta toimia odottamattomassa ja epätavallisessa tilanteessa, johon ei ohjeita ole olemassa. Odottamattomat tilanteet ovat myös juuri niitä tilanteita, joissa käyttäjä tarvitsisi järjestelmältä luotettavaa tietoa systeemin tilasta. Normatiivisen analyysin keinoin on mahdotonta selvittää, millaista tietoa käyttäjät poikkeustilanteissa tarvitsisivat päätöksentekonsa tueksi. Vicenten mukaan tehtävien ja ympäristön vuorovaikutuksen tulisi olla vuorovaikutteisen järjestelmän suunnittelun perusta.

2.1.1.2 Deskriptiivinen mallinnus

Deskriptiivisessä mallissa järjestelmä kuvataan sen perusteella, miten se tällä hetkellä toimii. Deskriptiivinen mallinnus perustuu usein jo olemassa olevien, ei välttämättä tietoteknisten, järjestelmien käytön havainnointiin.

Deskriptiiviset mallit nostavat hyvin esiin nykyisen työn ongelmakohtia. Deskriptiivisten mallien taustalla on usein laaja tutkimus, jossa on kerätty aineistoa käyttäjien tilannesidonnoisesta toiminnasta. Näin on voitu havaita ne erot, jotka jäävät käyttäjien toiminnan normatiivisen mallin ja todellisen toiminnan väliin. Nämä ovat deskriptiivisten mallien hyviä puolia.

Deskriptiiviseen malliin tähtäävän analyysin heikkous on se, että se keskittyy olemassa oleviin työtapoihin ja niiden kuvaamiseen. Tämä aiheuttaa sen, että analyysin pohjalta suunniteltava järjestelmä vastaa paremminkin menneen ajan tarpeita, eikä pysty ollenkaan ennakoimaan mahdollisia tulevaisuudessa tapahtuvia muutoksia. Deskriptiivisen analyysin vaarana on myös liiallinen tilanteen yksinkertaistaminen. Havainnoitaessa nykytilannetta jää analyysi helposti laajentamatta. Tällöin työtilanteet, jotka eivät havainnointivaiheessa toteutuneet, jäävät analyysin ulkopuolelle. Tämä voi johtaa analyysin huomattavaan yksinkertaistamiseen sekä samalla kapeahkoon skenaarioperustaiseen suunnitteluun.

Työn havainnointi ja analyysin tekeminen deskriptiivisesti ei saisi toimia informaatiojärjestelmän suunnittelun ainoana lähtökohtana, sillä todellisuudessa suunnittelun pohjana toimisivat tällöin nykyiset työrutiinit ja toimintatavat. Nykyiset käytännöt saattavat olla epäkäytännöllisiä tai jopa virheellisiä. Samoin vain pieni osa toiminnasta saattaa olla suunniteltavan järjestelmän kannalta merkityksellistä.

2.1.1.3 Formatiivinen mallinnus

Ratkaisuksi edellä esiteltyihin normatiivisen ja deskriptiivisen mallinnuksen ongelmiin Vicente esittää työn mallintamista formatiivisesti. Tällä hän tarkoittaa sitä, että suunniteltava järjestelmä kuvataan sen ominaisvaatimusten (engl. intrinsic constraints) avulla. Ominaisvaatimukset sisältävät työkohteen järjestelmälle asettamat rajoitteet ja mahdollisuudet. Näin järjestelmästä kuvataan ne reunaehdot, joiden puitteissa toiminnan on tapahduttava.

Työkohteen ominaisvaatimukset ovat kohteeseen itseensä sisältyviä rajoituksia ja mahdollisuuksia, joihin ei suunnitteluratkaisuilla voida vaikuttaa. Ominaisvaatimukset ovat ainoita rajoitteita innovatiiviselle työn uudelleen suunnittelulle ja työtapojen järjeistämiseksi. Ajatuksena on, että jos nämä sisäiset rajoitteet pystytään ottamaan suunnittelun lähtökohdiksi, on mahdollista muodostaa työtehtävät järkevällä tavalla rationaalisiksi, jolloin niihin sisältyy ainoastaan kokonaistehtävän kannalta merkitykselliset vaiheet. Myös Beyer ja Holtzblatt (1998) pitävät työn järjeistämistä yhtenä työjärjestelmien suunnittelun tavoitteista.

Formatiivinen malli kohteesta yleistää työn analyysin tulokset abstraktimmalle tasolle kuin mihin yksittäisten työtilanteiden normatiivinen tai deskriptiivinen malli antaisi puitteet. Mallinnettavasta kohteesta esitetään reunaehdot, jotka rajoittavat työn tavoitteiden saavuttamista. Samalla mallintuvat myös ne mahdollisuuksien rajat, joilla tavoitteet voidaan saavuttaa. Yksittäisiä tehtäväsarjoja, joilla tavoitteet voidaan saavuttaa, voi olla olemassa monia.

Vicente kuvaa formatiivisen mallin eroa kahteen edellä mainittuun seuraavalla esimerkillä: Jos henkilön tavoitteena on siirtyä paikasta A paikkaan B ja hänellä on apunaan yksityiskohtaiset ajo-ohjeet, saavuttaa hän tavoitteensa helposti, jos vallitseva ympäristö on samassa tilassa, missä ohjeiden tekohtekellä on oletettu. Toisaalta jos esimerkiksi jokin katu, jota henkilön pitäisi käyttää, on jostain syystä suljettu, joutuu hän tilanteeseen, jossa annetut ohjeet eivät enää päde. Tällöin toimivampi malli tavoitteen tukemiseen olisi antaa henkilölle käyttöön kartta, johon pisteet A ja B on merkattu. Kartta on formatiivinen malli ympäristöstä, sillä se kuvaa siihen liittyviä sisäisiä rajoitteita (esimerkiksi korttelin läpi ei voi ajaa autolla). Kartan avulla henkilö voisi etsiä uuden reitin suljetun katuosuuden tilalle ja pääsisi perille ennakoimattomasta tilanteesta huolimatta.

2.1.2 Ekologinen suunnittelu

Vicenten määrittelemä ekologinen suunnittelu on suunnittelua, jossa työkohteen suunniteltavalle järjestelmälle asettamia vaatimuksia käsitellään suunnittelun lähtökohtina. Ekologinen suunnittelu toteuttaa formatiivisen mallintamisen periaatteita, eli suunniteltava järjestelmä kuvataan työkohteen ominaisvaatimusten perusteella.

Monimutkaiset sosiotekniset järjestelmät kytkeytyvät tyypillisesti johonkin fyysiseen prosessiin. Järjestelmät tuottavat tietoa prosessin tilasta sekä mahdollistavat tilan muutosten hallinnan käyttäjän toimesta. Ekologinen lähestymistapa on näiden järjestelmien suunnittelussa tarpeellinen järjestelmästä saadun informaation välittyneisyyden takia. Koska ohjattavan prosessin käyttäytymistä määräävät tietyt fysikaaliset lainalaisuudet, on samat tekijät otettava huomioon myös ohjausjärjestelmää ja etenkin sen käyttöliittymää suunniteltaessa.

Vicenten kantava ajatus koko teoriassa on se, että monimutkaista sosioteknistä järjestelmää ohjaavan työn suunnittelussa pitää kuitenkin ainakin jollain tasolla lähteä liikkeelle siitä, mitkä ovat ympäristön käyttäjästä täysin riippumattomat fyysiset realiteetit. Ympäristöllä Vicente tarkoittaa toiminnan kohdetta eli työkohdetta. Työkohde ei ole staattisessa tilassa vaan se muuttuu sekä käyttäjän että ulkopuolisten vaikuttimien toimesta. Esimerkkejä järjestelmistä, niiden käyttäjiä ja kohteista on esitelty alla (Taulukko 2).

Taulukko 2. Esimerkkejä monimutkaisista järjestelmistä, niiden käyttäjistä ja käyttöympäristöistä.

Järjestelmä	Käyttäjä	Ympäristö
Ydinvoimalan ohjausjärjestelmä	Operaattori	Ydinvoimalaitoksen prosessi
Lentokoneen automaatiojärjestelmä	Lentäjä	Ilmatilan kaikkien alusten sekä maan pinnanmuotojen muodostama kokonaisuus
Anestesia­lääkärin potilas­seurantajärjestelmä	Anestesia­lääkäri	Ihmisen fysiologia
Osakekurssien ja -salkun arvojen seurantajärjestelmä	Pörssimeklari	Pörssissä vaikuttavat lainalaisuudet

Esimerkiksi ydinvoimalaitoksen ohjausjärjestelmää ekologisesti suunniteltaessa on ensin analysoitava ohjattava prosessi. Laitoksen operaattorille tulisi ekologisen suunnittelun teorian mukaan ohjausjärjestelmän avulla välittyä kokonaiskuva koko laitoksen kriittisten tekijöiden tilasta. Jotta käyttäjä voisi toiminnallaan ehkäistä erilaisia epätoivottuja poikkeustilanteita, on hänen pystyttävä havainnoimaan oman toimintansa vaikutukset esimerkiksi massatasapainoon tai muuhun ydinvoiman tuottoon liittyvään merkitykselliseen tekijään.

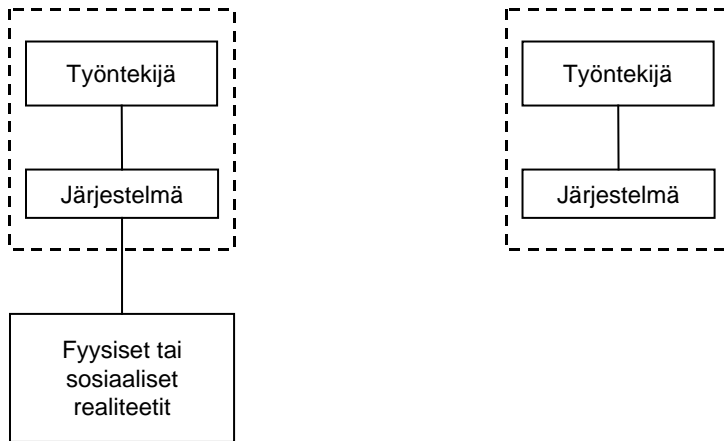
Ekologisen suunnittelun vastakohta on kognitiivinen suunnittelu. Nämä eroavat siten, että ekologisessa tarkastelussa lähdetään liikkeelle kohteen ominaisvaatimuksista ja tarkennetaan niitä kohti yksilön kognitiivisten rajoitusten asettamia vaatimuksia. Kognitiivisessa suunnittelussa suunta on päinvastainen. Yksilön kognitiivisten kykyjen asettamat vaatimukset toimivat siinä lähtökohtina. Kognitiivinen suunnittelu tarkastelee ilmiötä yksilön näkökulmasta, kun taas ekologisessa suunnittelussa näkökulma on ympäristöstä yksilöön päin. Ekologisessa suunnittelussa pyritään siis ottamaan huomioon ne ympäristön piirteet, jotka vaikuttavat yksilöön ympäristön osana.

Esimerkki kognitiivisesta lähestymistavasta suunnitteluun on järjestelmäkehitys käyttäjien mentaalisten mallien (Norman 1988) perusteella. Vicenten mukaan monimutkaisten sosioteknisten järjestelmien suunnittelussa tämä ei ole tarkoituksenmukaista. Hän kertoo esimerkin ydinvoimalan valvomon käyttäjakeskeisestä suunnittelusta, jossa suunnitteluun osallistui kokenut operaattori. Tuloksena oli ohjausjärjestelmän käyttöliittymä, jota ei osannut käyttää kukaan muu kuin itse suunnitteluun osallistunut käyttäjä. Vika oli siinä, että suunnittelun tietolähteeksi valitun käyttäjän sisäinen malli ydinvoimalaitoksen toiminnasta poikkesi muiden käyttäjien mallista. Jos suunnittelu tehdään työntekijöiden mentaalisten mallien perusteella, niihin sisältyvät puutteet ja väärinkäsitykset periytyvät suunniteltavaan järjestelmään. Kokeneetkin työntekijät yksinkertaistavat monimutkaisen järjestelmän ominaisuuksiin liittyviä syy- ja seuraussuhteita. Koska työntekijät ovat aina yksilöitä, ovat sisäiset mallitkin aina yksilöllisiä, joten suunnittelu niiden perusteella on käytännössä hyödytöntä, sillä erilaisia malleja on yhtä monta kuin työntekijöitä.

Tämän esimerkin sanoma on, että kognitiivinen suunnittelu ei tuota hyödyllisiä ratkaisuja, jos käyttäjien ympäristöä koskevat sisäiset mallit eivät ole yhteensopivia fyysisen ympäristön todellisen toiminnan kanssa. Tämän takia käyttäjakeskeisessä suunnittelussakin on lähdettävä liikkeelle fyysisen ympäristön realiteeteista. Kognitiivisella yhteensopivuudella on merkitystä vasta silloin, kun yhteensopivuus koskee kaikkia mahdollisia käyttäjiä. Kohteen monimutkaisuuden lisääntyessä kasvavat myös erot sitä koskevien eri yksilöiden sisäisten mallien välillä.

Vicenten ajatuksena on, että järjestelmä toimii ympäristönä, joka tukee yksilöllisiä tapoja suorittaa työtehtäviä. Oleellista on myös tunnistaa tehtävien ja ympäristön suhde eli se, millainen vuorovaikutus esimerkiksi dynaamisen prosessin ja

sen ohjausjärjestelmän avulla suoritettavan operoinnin välillä on. Informaatiojärjestelmän avulla saatava tieto työkohteesta on monimutkaisilla sosioteknisillä järjestelmillä aina välittynyttä (Kuva 3). Järjestelmä ei ole itsetarkoitus, vaan se on olemassa välittääkseen tietoa tietyistä fyysisistä todellisuudesta. Tämän takia ekologinen näkökulma on niiden suunnittelussa perusteltua.



Kuva 3. Välittynyt vuorovaikutus monimutkaisissa järjestelmissä.

Kun suunnittelu perustuu järjestelmän todelliseen käyttäytymiseen, on mahdollista havaita työntekijöiden vääristyneet käsitykset ja rajoitukset ja korjata niitä. Kognitiivisen työn analyysin tavoitteena on kehittää järjestelmä, joka sallii poikkeamat määritellyistä rutiineista. Järjestelmä antaa reunaehdot sille, mitä missäkin tilanteessa voidaan tehdä, mutta ei välttämättä ”ohjeista” tiettyä ratkaisumallia aina tiettyyn tilanteeseen. Tällaisen järjestelmän käyttöönotto vaatii myös organisatorisia muutoksia, jotka ulottuvat työntekijöiden koulutuksen perusteisiin asti. Jotta päästäisiin parhaaseen mahdolliseen lopputulokseen, olisi ekologiseen suunnitteluun kuitenkin pystyttävä yhdistämään kognitiivinen näkökulma. Tähän Vicente on pyrkinyt kehittäessään työn analysointitekniikan nimeltä Kognitiivinen työn analyysi.

2.2 CWA prosessina

Kognitiivisen työn analyysi eli CWA toimii ekologisen formatiivisen suunnittelun metodologiana. Kognitiivinen näkökulma yhdistetään ekologiseen näkökulmaan suunnitteluprosessin viidennessä vaiheessa. Työn ominaisvaatimuksia analysoitaessa lähdetään liikkeelle fyysisestä maailmasta, ympäristöstä ja työn kohteesta. Analyysin vaiheiden kautta vaatimuksia tarkennetaan kohti kognitiivisia käyttäjän kykyjä kuvaavia vaatimuksia.

CWA:han kuuluu joukko mallinnustyökaluja. Nämä esitellään tämän työn neljännessä luvussa vaatimusten esittämistä koskevassa kohdassa.

2.2.1 Ensimmäinen vaihe: Työkohteen analyysi

Kognitiivinen työn analyysi alkaa työn ympäristön ja kohteen tutkimuksella ja analyysillä. Tavoitteena on selvittää systemaattisesti ympäristön työlle asettamat reunaehdot ja rajoitteet. Esimerkiksi ohjausjärjestelmää suunniteltaessa työkohte on ohjauksen alainen, teollinen tai muu prosessi. Tiedon keräysmenetelmiin Vicente ei ota kantaa. Kognitiivinen työn analyysi käsittelee ainoastaan jo jollain metodilla kerätyn datan analysointia.

Ensimmäisen vaiheen tuotoksena on määritelmä järjestelmän ympäristön lähtökohdille, jotka Vicente määrittelee suunniteltavalle järjestelmälle asetettaviksi vaatimuksiksi. Nämä vaatimukset ovat työntekijästä riippumattomia esim. fyysisen tai sosiaalisen ympäristön asettamia ja erityisesti ohjattavasta prosessista johdettavia, kuten esimerkiksi vasteajat.

Keskeisenä ajatuksena CWA:n ensimmäisessä vaiheessa on kuvata se työympäristö, joka pysyy muuttumattomana järjestelmän tai ohjattavan prosessin tilasta riippumatta. Kun ympäristö on kuvattu, voidaan samaan malliin liittää tieto siitä, mikä informaatio on merkityksellistä käyttäjän kannalta.

2.2.2 Toinen vaihe: Hallintatehtävien analyysi

Kognitiiviseen työn analyysin toinen vaihe on panos-tuotostyyppinen tehtävä-analyysi. Tehtäväanalyysi sekä ympäristön analyysi liittyvät tiukasti toisiinsa. Ympäristö analysoidaan ensin ja sen jälkeen tehtävät, joita siinä nimenomaisessa ympäristössä tulisi suorittaa. Näin ollen tehtäviä ei koskaan irroteta kontekstistaan. Tämä ei edelleenkään poikkea merkittävästi ISO-standardin mukaisesta käyttäjakeskeisestä tuotekehityksestä, jossa käyttökonteksti määritellään ennen tehtäväanalyysia, mutta tästä poiketen CWA:ssa tehtäväanalyysi tehdään järjestelmän kokonaistavoitteiden, ei käyttäjäyksilöiden, lähtökohdista. Hallintatehtävien analyysin tavoitteena on siis selvittää vaatimukset, jotka liittyvät tunnistettuihin tapahtumiin ja tavoitteisiin tunnetussa ympäristössä. Panos-tuotos mallia käytetään tehtäväanalyysissa, koska monimutkaiset sosiotekniset järjestelmät ovat avoimia systeemejä, jolloin mahdolliset ulkoiset häiriötekijät vaikuttavat tehtäviin. Tämän takia erilaisia häiriötekijöistä riippuvia ratkaisuvaihtoehtoja on olemassa useita.

Hallintatehtävien analyysissa mallinnetaan toimintoja. Näitä kuvataan verbeillä. Analyysissa kuvataan kuitenkin vain se, *mitä* tehtävillä pitää saavuttaa, ei sitä, *miten* se saavutetaan. Tällä pyritään saavuttamaan se, että kehitetyt ratkaisut ovat laite-, tekniikka- yms. riippumattomia.

2.2.3 Kolmas vaihe: Strategian analyysi

CWA:n kolmas vaihe on nimeltään strategia-analyysi (Strategies Analysis). Strategialla tarkoitetaan tässä sitä, *miten* yksittäinen työntekijä edellisessä vaiheessa määritellyt tavoitteet saavuttaa. Vicente korostaa, että tässä vaiheessa tulevat esiin yksittäisten työntekijöiden erot. Saman tavoitteen saavuttamiseksi voi kahdella eri käyttäjällä olla hyvinkin erilainen strategia. Toisaalta myös ensimmäisessä vaiheessa analysoitu ympäristö asettaa rajoitteita strategialle. Joku tietty strategia ei toimikaan kaikissa järjestelmän eri tiloissa. Tietoteknisen, ihmisten työtä tukemaan suunnitellun järjestelmän tulisi tukea käyttäjien yksilöllisiä tilanneriippuvaisia ongelmanratkaisustrategioita. Vicente toteaa myös, että yhden strategian toteuttamiseen on aina olemassa monta eri toteutusvaihtoehtoa.

Strategia-analyysissä tunnistetut strategiat hyödynnetään suunniteltaessa järjestelmän dialogia sekä tehtäväsekvenssejä. Samoin kuin edellisinkin työn analyysin vaiheet myös tämä kolmas vaihe tehdään formatiivisen analyysin menetelmällä, johon kuuluu, että tulokset esitetään vaatimuksina strategioiden toimivuudelle.

2.2.4 Neljäs vaihe: Organisaation ja yhteistyön analyysi

Kognitiivisen työn analyysin neljäs vaihe on organisaation ja yhteistyön analyysi (Social Organisation and Cooperation Analysis). Siinä analysoidaan, miten eri strategioihin pohjautuvat ongelmanratkaisumallit jaetaan ihmisen ja automaation välillä. Samalla määritellään se, miten ihmiset ja tekniikka kommunikoivat ja tekevät yhteistyötä. Vaiheen tavoitteena on myös yhdistää monitieteellisellä tavalla tietämystä esimerkiksi organisaatioteorian ja työpsykologian aloilta teknisen järjestelmän suunnitteluun. Samalla, kun suunnitellaan tietoteknistä järjestelmää jonkin työn suorittamiseen, joudutaan ottamaan kantaa myös uuden järjestelmän aiheuttamiin organisatorisiin muutoksiin työyhteisössä. Tämä vaihe määrittelee siis vaatimuksia organisaation rakenteelle.

2.2.5 Viides vaihe: Työntekijän osaamisen analyysi

Viides, ja samalla viimeinen Kognitiivisen työn analyysin vaihe on työntekijöiden osaamisen analyysi (Worker Competencies Analysis). Siinä on tavoitteena tunnistaa työntekijöiden eli järjestelmän käyttäjien kognitiivisten kykyjen asetamat reunaehdot järjestelmän suunnittelulle. Vicenten mukaan vasta tämä vaihe ottaa esiin perinteiset HCI-alan kysymykset siitä, miten ihmisten kyvyt ja heikoudet vaikuttavat tietoteknisten järjestelmien suunnitteluun. Kun kompetenssi-analyysi tehdään formatiiviseen tapaan, tulee silloin vaiheen tuotoksena itse asiassa vaatimuksia työntekijöiden kompetensseille. Vicente itse ehdottaakin, että suunniteltaessa järjestelmää jo olemassa olevalle työntekijäjoukolle, kannattaa analyysin tämä vaihe tehdä deskriptiivisesti, eli kuvailla käyttäjien kompetenssit ja johtaa niistä vaatimuksia kehitettävälle systeemille.

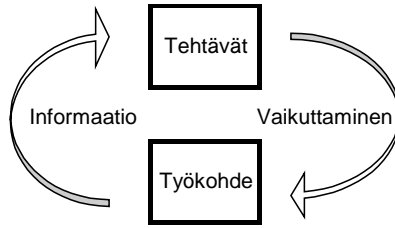
2.3 CWA:n rooli vaatimusten määrittelyssä

Vaatimusten määrittelyä ei ole CWA:ssa eroteltu mitenkään omaksi kokonaisuudekseen, vaan koko prosessi käsittelee sitä, miten erilaisia järjestelmävaatimuksia muodostetaan työtä analysoimalla.

Mielestäni CWA:n merkittävin anti vaatimusmäärittelyille on se, että suunniteltaessa tietynlaisia reaaliaikaisia järjestelmiä, joiden piirteisiin kuuluvat monimutkaiset ja sosiotekniset yksityiskohdat, on vaatimusten määrittely aloitettava fyysisten realiteettien tunnistamisella. Eli käytännössä, jos esimerkiksi suunnitellaan ydinvoimalan automaatiojärjestelmää käyttöliittymineen, on käytännön kenttätutkimus aloitettava tutkimalla ydinvoimalaprosessia ja sen käynnissäpitoon ja tuottavuuteen liittyviä reunaehtoja. Näin luodaan edellytykset sille, että voidaan suunnitella järjestelmään käyttöliittymä, joka välittää käyttäjälle prosessin todellisen tilan ja siihen vaikuttavien kriittisten tekijöiden merkitykset.

Toinen tärkeä huomio Vicenten teoriassa on se, että järjestelmävaatimusten määrittely ei voi perustua sekvenssimäiseen tehtäväanalyysiin. Hän toteaaakin, että tämänkaltainen tehtävien suoritus on varsinkin tulevaisuudessa se, joka automatisoidaan ja jätetään näin koneiden hoidettavaksi. Ihmisten adaptiivista ongelmanratkaisukykyä tarvitaan paljon monimutkaisemmassa päätöksenteossa. Tämän takia työn tukijärjestelmäkään eivät saisi perustua proseduraaliseen tehtävien suoritukseen, vaan niiden pitäisi enemminkin olla formatiivisesti määriteltäviä ja toteutettuja, jolloin käyttäjälle jää vapausasteita oman osaamisensa käyttämiseen järjestelmän tavoitetilan saavuttamisessa ja ylläpidossa.

Kognitiivinen työn analyysi yhdistää mielenkiintoisella tavalla ihmisen toiminnan tutkimusta sekä insinööritieteitä. Lähestymistapa on hieman perinteisestä käytettävyyssajattelusta poikkeava, mutta yhtäläisyyksiäkin on. Esimerkiksi käyttökontekstin määrittelyn ja erilaisten järjestelmävaatimusten muodostamisen merkitystä korostetaan, mutta näkökulma mallinnukseen on insinöörimäinen. Tässähän ei sinänsä ole mitään vikaa; näkemys on vain käyttäjälähtöisen suunnittelun perinteen kannalta yllättävä, sillä nimenomaan tämän tapaisesta ”insinööriajattelusta” on aikaisemmin pyritty irtautumaan. Malli tuntuu myös hieman rajalliselta, sillä ihmisen toiminta on huomattavasti monimutkaisempaa kuin esimerkiksi säätöpiirin, johon Vicente välillä ihmistä vertaa.



Kuva 4. Käyttäjän ja kohteen välinen vuorovaikutussuhde.

Monimutkaisten sosioteknisten järjestelmien ollessa kyseessä ovat käyttäjän toiminta ja käyttöympäristö jatkuvassa vuorovaikutuksessa keskenään (Kuva 4). Käyttäjä vaikuttaa tehtävillään ympäristöön, joka reagoi. Ympäristön tilan muutoksista syntyvä informaatio taas vaikuttaa edelleen käyttäjän tehtäviin ja siten hänen toimintaansa. Ekologisessa suunnittelussa olisi pystyttävä pureutumaan käyttäjän ja ympäristön vuorovaikutuksen mallintamiseen. Nykyisellään Vicenten kuvaama ekologinen lähestymistapa ei mahdollista tätä. Siinä mallintaminen lähtee liikkeelle suoraan kohteesta, eikä ota huomioon sitä, että kohteen malliin vaikuttaa myös se, missä suhteessa käyttäjä on kohteeseen. Ei nimittäin voida olettaa, että mitään työkohdetta pystyttäisiin täysin objektiivisesti mallintamaan. Esimerkiksi eri työtehtävissä toimivat käyttäjät näkevät työkohteensa hyvinkin erilaisena. Tällöin tuntuisi ehkä luonnollisemmalta lähteä liikkeelle mallintamisesta siitä, mitkä ovat tilannesidonnaiset käyttäjän tavoitteet jonkin tietyn työkohteen suhteen.

3. Käyttäjätutkimukset

Tässä luvussa esitellään käyttäjäkeskeisen suunnittelun ensimmäinen osa, käyttäjätutkimus. Erilaiset menetelmät käydään läpi lyhyesti, jolloin ne muodostavat taustatietoa luvussa neljä selostetuille tiedon analysointi- ja vaatimusten määrittelytekniikoille.

3.1 Käyttäjätiedon tarpeet

Relevantti ja oikea tieto käyttäjistä on koko käyttäjäkeskeisen suunnittelun perusta. Käyttäjätiedolla tarkoitetaan suunnittelun kohteelle olennaista, käyttäjää ja hänen toimintaansa koskevaa tietoa. Tällaista informaatiota ovat esimerkiksi käyttäjän persoonaa, koulutustaustaa ja kokemuksia koskevat tiedot. Toisaalta käyttäjätiedon piiriin kuuluu myös tieto käyttötilanteesta ja ympäristöstä sekä käyttäjän tehtävistä. Käyttäjätieto on tarpeen hankkia ennen vaatimusmäärittelyn tekemistä, jolloin se käytetään hyväksi suunnittelun perustana toimivien vaatimusten muodostamisessa.

Käyttäjäkeskeisen suunnittelun periaatteita (Liite A, Liite B) toteuttavassa suunnittelussa järjestelmävaatimusten muodostaminen on haastava tehtävä. Käyttäjätietoa tarvitaan, jotta vaatimukset osataan määrittellä lopputuotteen laatua palveleviksi. Laatuvaatimusten taustalla on tarve kehittää tuotteita, joita loppukäyttäjä pystyy käyttämään ja joita hän jopa haluaa käyttää. Käyttäjävaatimusten määrittelyä helpottaa niiden johtaminen loppukäyttäjän tarpeista (Kujala 2002), joita pyritään selvittämään käyttäjätietoa keräämällä.

Fyysiseen prosessiin liittyvää informaatio- tai ohjausjärjestelmää suunniteltaessa on tärkeää ottaa huomioon myös prosessia koskevat vaatimukset. Tällöin varsinkin ylätasolla käyttäjävaatimukset perustuvat nimenomaan prosessiin. Näin myös käyttäjävaatimuksia muodostettaessa on prosessin ymmärtäminen tärkeää, eli käyttäjän tarpeet muodostuvat tarpeesta saavuttaa prosessin optimaalinen tila tai ylläpitää sitä.

3.2 Käyttäjätutkimusmenetelmät

Tässä kappaleessa esiteltävät menetelmät on jaoteltu epäsuoriin ja suoriin sen mukaan suoritetaanko tutkimus järjestelmän käyttötilanteessa vai ei. Nielsenin (1993) tämä jaotteluperuste on käytettävyyttä kehitettäessä järkevä. Epäsuorat menetelmät esitellään ensin lyhyesti. Näissä menetelmissä tutkija voi tehdä lähes koko tutkimustyön omassa ympäristössään esimerkiksi analysoiden jonkin lomakkeen avulla kerättyä tietoa. Suorat menetelmät perustuvat siihen, että tutkimuksen tekijä on aidossa vuorovaikutuksessa loppukäyttäjän kanssa. Tutkimus tapahtuu tyypillisesti käyttäjän ympäristössä ja hänen ehdoillaan.

3.2.1 Epäsuorat tutkimusmenetelmät

Seuraavat menetelmät ovat luonteeltaan suunnittelijakeskeisiä. Näillä menetelmillä suoritettujen tutkimusten kulku on suunniteltu etukäteen melko tarkkaan tutkijan toimesta. Epäsuorilla menetelmillä kerätty tieto on usein luonteeltaan kvantitatiivista tai kerätty aineisto pyritään jollain menetelmällä kvantifioimaan.

3.2.1.1 Haastattelut

Haastattelu on tutkimusmenetelmä, jossa haastattelija esittää kysymyksiä, joihin haastateltava vastaa. Haastattelija tallentaa haastateltavan antamat vastaukset hyväksi katsomallaan menetelmällä. Haastattelu sopii tutkimusmenetelmäksi silloin, kun ei vielä täysin tarkasti tiedetä, minkälaista ilmiötä ollaan tutkimassa tai mitä tarkalleen ottaen ollaan etsimässä. (Nielsen 1993)

Haastattelija valmistautuu haastattelutilanteeseen miettimällä etukäteen asiat, jotka hän haluaa tutkimuksellaan selvittää, ja muotoilemalla kysymyksensä tavoitteidensa mukaisiksi. Mitä tarkemmin haastattelun kulku on etukäteen suunniteltu, sitä strukturoidummasta haastattelusta on kyse. Puolistrukturoitua haastattelua käytetään kun halutaan saada yksityiskohtainen kuva haastateltavan uskomuksista ja havainnoista jonkin tietyn asian suhteen. (Smith 1995)

3.2.1.2 Kyselyt

Kyselyllä tarkoitetaan erityisen, tiettyyn tarkoitukseen kehitetyn kyselylomakkeen avulla tehtyä tutkimusta. Kyselylomake voidaan esittää käyttäjälle joko paperilla tai tietokoneavusteisesti. Käyttäjä vastaa kyselyyn yksin. (Nielsen 1993)

Kyselymenetelmä on hyödyllinen silloin, kun käyttäjät ovat vaikeasti tavoitettavia tai muuten vaikeasti haastateltavissa. (Sutcliffe 2002) Kyselyillä voidaan myös suhteellisen pienellä vaivalla saada erittäin kattava, joskus jopa kaikki käyttäjät kattava otos. (Nielsen 1993)

Kyselylomakkeella voidaan kerätä tietoa olemassa olevan järjestelmän puutteista. Tällöin kysymysten mutoilu voi olla ongelmallista, sillä kysymysten tulisi olla mahdollisimman yksiselitteisiä. (Sutcliffe 2002) Toisaalta kyselyillä voidaan mitata myös käyttäjäkunnan kvantitatiivisista ominaisuuksista kuten esimerkiksi ikä-, sukupuoli- tai koulutusjakaumaa. Tällöin kyselyn avulla voidaan luoda erilaisia käyttäjäprofiileja.

3.2.1.3 Ryhmämenetelmät

Menetelmiä, joissa ryhmä tuotteen käyttäjiä kokoontuu yhteen tilaan keskustelemaan mielipiteistään tuotteesta, kutsutaan ryhmämenetelmiksi. Tunnetuin ja menetelmänä vahvin ryhmämenetelmä lienee fokusryhmä. Käytettävyytutkimuksessa käytetty fokusryhmä eroaa perinteisestä markkinoinnissa käytetystä fokusryhmästä siten, että käytettävyytutkimuksessa menetelmää käytetään ennen tuotteen suunnittelua potentiaalisten käyttäjien piirteiden kartoittamiseen, kun taas markkinoinnissa tutkitaan reaktioita valmiisiin tuotteisiin. (Ede 1998) Fokusryhmätyöskentelyssä alle kymmenen suunniteltavan tuotteen potentiaalista käyttäjää kokoontuu yhteen keskustelemaan uudesta konseptista. Keskustelua johtaa moderaattori, joka on tyypillisesti käytettävyys- tai suunnitteluinsinööri. (Nielsen 1993)

Menetelmän idea on, että kun käyttäjiä on muutama pohtimassa samaa asiaa, pystyvät he inspiroimaan toisiaan ja näin nostamaan esille enemmän tietoa ja yksittäisiä mielipiteitä kuin silloin, kun vain yhtä henkilöä haastatellaan kerrallaan. Menetelmän onnistumisen edellytys on kuitenkin ryhmässä syntyvän keskustelun hyvä henki. Yksi käyttäjä ei saa dominoida keskustelua. (Hackos &

Redish 1998) Tähän tarvitaan käyttäjiä, jotka osaavat ilmaista itseään ja pystyvät analysoimaan omaa toimintaansa.

Fokusryhmien käytön työläys syntyy siitä, että käyttäjien edustajia on saatava samaan aikaan paikalle suhteellisen paljon, eli vähintään kuusi. Tämän lisäksi yksi ryhmähaastattelu ei yleensä riitä antamaan tarpeeksi kokonaisvaltaista kuvaa tarkasteltavasta ilmiöstä. (Nielsen 1993) Ainakin kaksi ryhmähaastattelua täytyy tehdä jo senkin takia, että voidaan varmistaa tulosten olevan tilanne- ja osallistujariippumattomia. Ns. hiljaista tietoa on vaikea ryhmämenetelmillä saada esiin, sillä sille on määritelmän mukaan ominaista se, että se pystytään ilmaisemaan ainoastaan luonnollisessa käyttötilanteessa. Ryhmämenetelmistä voi saada vähän parempia tuloksia, kun ottaa tilanteeseen mukaan artefakteja, jotka luovat mielleyhtymiä todellisiin toimintatilanteisiin. Toisaalta esimerkiksi Ede (1998) kertoo fokusryhmillä voitavan välttää tämäkin ongelma, kun keskustelun aiheet pidetään tarpeeksi konkreettisina ja lähellä normaalin työpäivän rutiineja.

3.2.2 Suorat menetelmät

Seuraavat menetelmät ovat luonteeltaan käyttäjakeskeisempiä kuin edellä esitellyt epäsuorat menetelmät. Käyttäjakeskeisillä menetelmillä kerätty tieto on tyyppillisesti kvalitatiivista.

3.2.2.1 Havainnointi

Havainnointi on yksi tärkeimmistä käyttäjätutkimuksen menetelmistä. Anttila (1998) jakaa erilaiset havainnointimenetelmät neljään eri luokkaan: suora, strukturoitu, strukturoimaton ja osallistuva havainnointi. Suorassa havainnoinnissa tutkija havainnoi käyttäjien toimintaa näiden luonnollisessa toimintaympäristössä. Havainnoinnin strukturoinnin aste kertoo havainnoinnin ennalta määriteltävien tavoitteiden määrästä. Mitä tarkemmin havainnoinnille on ennalta määriteltäviä tavoitteita, eli ”tutkimuskysymyksiä” sitä strukturoidummasta havainnoinnista on kyse. Strukturoimatonta havainnointia käytetään esimerkiksi silloin, kun halutaan kerätä mahdollisimman paljon ennakkotietoja jostain etukäteen tuntemattomasta ilmiöstä. Osallistuvassa havainnoinnissa tutkija on vahvasti läsnä tutkimuskohteessaan. Tutkija yrittää selvittää, mikä tarkasteltava ilmiö ylipäättään on ja mitä kaikkea tilanteeseen liittyy. Hän selvittää myös miten tapahtu-

maan osallistujat itse kuvaavat tilannettaan, ja miten he itse sen ilmaisevat. Tutkija on osallistuja, mutta toisaalta hän seuraa toisten käyttäytymistä. Tilanteesta riippuen hän osallistuu toimintaan enemmän tai vähemmän aktiivisesti, mutta täysin ulkopuoliseksi hän ei voi jättäytyä. On tärkeää, että tutkijan rooli on tietoinen. Roolin tulee olla myös tutkimusongelman kannalta mielekäs. (Anttila 1998)

Käyttäjien havainnoinnissa on muistettava, että tutkijan läsnäolo vaikuttaa aina jollain tavalla havainnoitaviin. Tämä johtuu yksinkertaisesti siitä, että havainnoitavat ovat ihmisiä, eivätkä pysty kovin nopeasti unohtamaan olevansa havainnoitavina. Käytöksen muutos voi olla siis myös täysin tiedostamatonta. (Faulkner 2000)

3.2.2.2 Etnografinen tutkimus

Etnografinen tutkimus perustuu erittäin perusteelliseen käyttäjien toiminnan havainnointiin. Se on alunperin antropologian tutkimusalueelle tyypillinen menetelmä, jossa tutkimuksen kohteen elämää ja toimintaa seurataan niin kauan, että sen lainalaisuudet ja rutiinit alkavat hahmottua. Etnografisella menetelmällä kerätyn aineiston analysoinnin peruseriaatteena on löytää ja kehittää sellaisia käsitteitä, jotka auttavat ymmärtämään, mistä tutkittavassa ilmiössä on kysymys. Ei niinkään pyritä ymmärtämään, *miksi* jotain tapahtuu, vaan *mitä* tapahtuu. (Anttila 1999)

Etnografiselle tutkimukselle on kasvava tilaus myös käyttäjakeskeisen suunnittelun tutkimusmenetelmänä, sillä sen avulla uskotaan pystyttävän hahmottamaan käyttäjien toiminnan sosiaalista kontekstia. (Faulkner 2000)

Klassinen esimerkki etnografisesta tutkimuksesta käyttäjätiedon keräysmenetelmänä on Heathin ja Luffin (1992) tutkimus Lontoon metron erään linjan valvontakeskuksesta. Tutkijat pystyivät etnografisin menetelmin selvittämään liikenteenvalvontatyön sosiaalista ja yhteisöllistä luonnetta. Samoin saatiin selville hienovaraisia tapoja, joilla työntekijät kommunikoivat keskenään. Näitä tietoja ei olisi voitu saada esiin haastattelemalla työntekijöitä. (Heath & Luff 1992)

Etnografia on erittäin raskas käyttäjätutkimusmenetelmä. Sekä aineiston kerääminen, että sen analyysi ovat aikaa vieviä prosesseja. Tyypillisesti aineisto

koostuu ääni- ja videonauhoitteista, tutkijan omista musitiinpanoista, sekä talteen otetuista artefaktoista. Aineiston analyysissä käytetään laadullisen analyysin menetelmiä, eli havaintoja tyypitellään ja niistä pyritään muodostamaan merkityksellisiä kategorioita.

Etnografisen tutkimuksen tulokset ovat kuvauksia havainnoiduista tilanteista. Yksi menetelmän haittapuolia onkin, että tuloksia on erittäin vaikea suoraan hyödyntää suunnitteluprosessissa. (Faulkner 2000)

3.2.2.3 Ääneen ajattelu

Havainnoitaessa käyttäjiä, voi tutkijan olla joskus vaikea ymmärtää miksi käyttäjä toimii niin kuin hän toimii. Tähän ongelmaan voidaan pureutua ääneen ajattelu -menetelmällä, joka toimii käytännössä niin, että käyttäjä suorittaa normaaleja rutiinejaan, mutta ajattelee koko ajan ääneen, eli periaatteessa selittää havainnoijalle, mikä saa hänet tietyssä tilanteessa tekemään tietyn päätöksen oman toimintansa suhteen. (Nielsen 1993) Havainnointitilanne voidaan nauhoittaa ja nauhan avulla tutkijan on helppo palata todelliseen tilanteeseen ja sen ongelmakohtiin.

Menetelmän hyvä puoli on se, että se ei tuo mitään lisäkustannuksia tavalliseen havainnointitilanteeseen. Se on siis halpa käyttää. Sen avulla saa myös hyvin esiin käyttäjän ongelmatilanteet ja väärät ajatusmallit joihin käytettävä järjestelmä on antanut aiheen. Toisaalta ääneen ajattelu on hieman epäluonnollista ja kaikille käyttäjille ainakin aluksi vaikeaa. Varsinkin eksperttikäyttäjien on todettu olevan vaikeaa ilmaista päätöksentekoprosessiaan verbaalisti. (Nielsen 1993, s. 224) Menetelmälle on tyypillistä, että kun käyttäjä ajautuu tilanteeseen, jossa hän miettii ankarasti seuraavaa toimintoaan, ääneen ajattelu katkeaa. Juuri tällöin olisi mielenkiintoista tietää mitä hänen ajatuksissaan liikkuu, mutta tutkija joutuu ainoastaan toteamaan, että tämän kaltaisessa tilanteessa käyttäjä on vaikeuksissa. Jos tutkija huomauttaa käyttäjälle ääneen ajattelun katkenneen, käyttäjän ajatteluprosessia häiriintyy jonkin verran.

3.2.2.4 Contextual Inquiry

Contextual Inquiry (CI) on kaikkia edellä esiteltyjä menetelmiä yhdistelevä käyttäjätiedon keräysmenetelmä. CI:n ovat kehittäneet Hugh Beyer ja Karen Holtzblatt. Menetelmä on osa laajempaa suunnitteluprosessin viitekehystä, Contextual Designia (CD), mutta sitä voidaan käyttää yksistäänkin minkä tahansa käyttäjakeskeisen tuotekehitysprosessin käyttäjätiedon keräysmenetelmänä. (Beyer & Holtzblatt 1998)

Contextual Inquirya suorittavan tutkijan tulisi suhtautua haastateltavaan samalla tavalla, kuin oppipoika suhtautuu mestariinsa. Hän seuraa mestaria tämän työssä. Hän haluaa oppia työstä kaiken mahdollisen ja tietää aina myös sen, miksi jokin tietty työvaihe tehdään niin kuin se tehdään. Tutkija esittää kysymyksiä, kun ei ymmärrä jotain asiaa, ja hän aina myös varmistaa ymmärtäneensä asiat oikein. Toisaalta hän antaa mestarille tilaa tehdä työnsä eikä häiritse tätä työn kriittisissä vaiheissa. Tutkijan omaksuma harjoittelijan asenne on CI:n ensimmäinen lähtökohta. (Beyer & Holtzblatt 1995)

CI rakentuu neljän eri pääperiaatteen ympärille. Nämä ovat (Beyer & Holtzblatt 1998):

1. **Konteksti:** Tällä tarkoitetaan sitä, että haastattelu tehdään aina haastateltavan työympäristössä, työtehtävien lomassa. Tämä mahdollistaa sen, että tutkija pystyy keräämään konkreettista tietoa meneillään olevasta kokemuksesta eikä yleistyksiä.
2. **Kumppanuus:** Periaatteella pyritään siihen, että sekä haastattelija, että haastateltava työskentelevät yhdessä tasavertaisina yhteisen tavoitteen eteen. Tämä poikkeaa perinteisestä haastattelusta, jossa vastuu on ensisijaisesti aina haastattelijalla.
3. **Tulkinta:** Havainnointi tai haastattelu ei ole itsetarkoitus. Tulkinta antaa havainnoille merkityksen. Tulkinnan avulla määritellään, mitä tämänhetkisen työn rakenne tarkoittaa tulevan ratkaisun kannalta. Tulkinnat on aina todennettava käyttäjän kanssa.

4. **Fokus:** Tällä tarkoitetaan haastattelijan näkökulmaa työn tutkimukseen. Fokuksen avulla haastattelijaa rajaa aiheensa ja pitää haastattelutilanteessa keskustelun painopisteen haluamissaan asioissa.

Ennen minkään tutkimuksen aloittamista tutkijan on määriteltävä ongelma, jonka aikoo tutkimuksellaan ratkaista. CI:ssa tutkimusongelma asetetaan suhteessa työhön, jonka järjestelmiä ollaan suunnittelemassa (Beyer & Holtzblatt 1998). Tyypillisesti esimerkiksi markkinointiosasto voi määrittellä suunnittelijat tekemään järjestelmän, joka automatisoi jonkin nykyisin manuaalisen työn vaiheen tai korvaa jonkin olemassa olevan järjestelmän. Tutkijan on tämän lisäksi määriteltävä havainnointinsa fokus. On mietittävä kysymyksiä: Mikä on se työ, jota järjestelmän tulisi tukea? Minkälaisen osan se muodostaa työntekijän tehtävistä? Mitkä ovat työhön liittyvät avaintehtävät? Tutkijan on myös päätettävä, keitä hän haluaa CI-menetelmällä haastatella. Vastaus löytyy miettimällä seuraavia kysymyksiä: Kuka tekee työn tällä hetkellä? Keneltä hän saa muodollista tai epämuodollista apua tehtävissään? Keneltä hän saa tarvitsemansa tiedot työn suorittamiseen? Kuka käyttää työn tuloksia? Mikä on työhön liittyvä kulttuurinen ja sosiaalinen konteksti? Näiden kysymysten avulla tutkija voi miettiä, keitä hän haluaa haastatella, ja minkä tehtävien yhteydessä. (Beyer & Holtzblatt 1998)

Tutkimuksen yhteydessä etsitään myös reaali maailmasta metaforia, jotka auttavat rationalisoimaan työn rakennetta ja ymmärtämään käyttäjän tavoitteita paremmin (Beyer & Holtzblatt 1998). Esimerkiksi hakukonetta suunniteltaessa voi havainnoida myös miten ihmiset etsivät asioita tutussa ympäristössään, kuten ruokakaupassa. Tämä voi auttaa hahmottamaan geneerisen etsimisprosessin rakennetta, jolle tietysti myös online-hakukoneen rakenteen tulisi perustua.

Usein CI-tilanteet nauhoitetaan videolle. Tämä varmistaa sen, että tutkija pystyy keskittymään interaktioonsa käyttäjän kanssa, eikä hänen tarvitse koko ajan samalla keskustella ja kirjoittaa muistiinpanoja. Näin tilanne on myös luonnollisempi käyttäjän kannalta sillä hän voi selostaa asioita sitä mukaa, kun ne tulevat eteen, eikä hänen tarvitse odotella, että tutkija saa asiat kirjoitetuksi muistiin. (Beyer & Holtzblatt 1998)

CI alkaa siten, että käyttäjä suorittaa tavallisia työtehtäviään samalla tavalla kuin normaalistikin. Häntä voidaan myös pyytää selittämään tekemisiään, sillä tutkijan on vaikea ymmärtää tekojen motiiveja ilman selittämistä. Tällä pyritään sii-

hen, että käyttäjä ei selosta suunnittelijalle sitä, miten työtehtävät pitäisi yrityksen määrittelemien ohjeiden mukaan tehdä, vaan nimenomaan sitä, miten hän itse ne tekee. Tutkijan kannattaa kiinnittää huomiota siihen, että käyttäjä kertoo tekemisistään yksikön ensimmäisessä persoonassa. Tällöin hän kertoo omista toimistaan. Jos hän alkaa käyttää puheessaan passiivia tai monikon ensimmäistä persoonaa, viittaa tämä siihen, että hän kertoo yrityksen toimintatavoista yleisellä tasolla, ja esittää siis yleistyksiä. Tämä on sellaista tietoa, jonka tutkija voisi saada esiin työtehtävien määrittelyistä tai järjestelmien käyttöohjeistakin. (Beyer & Holtzblatt 1995)

Aina kun kaksi ihmistä on vuorovaikutuksessa keskenään siten, että ensimmäinen selittää toiselle tekemisensä ja ajatuksiaan, tekee kuunteleva henkilö mielesään erilaisia tulkintoja kuulemastaan. CI:ssa on tärkeä varmistaa, että tulkinnot ovat oikeita, ja käyttäjä on niistä samaa mieltä tutkijan kanssa. Tutkija voi esimerkiksi ajatellessaan ymmärtäneensä jonkin asian, varmistaa sen käyttäjältä yksinkertaisesti kysymällä. (Beyer & Holtzblatt 1998)

Contextual Inquiry:lla saadaan kerättyä valtava määrä dataa käyttäjän toiminnasta. Tämän tiedon purkaminen ja välittäminen eteenpäin koko suunnittelutiimille on haastava tehtävä, mutta siihenkin on CD-metodologiassa kehitetty menetelmiä. Näitä selostetaan tämän julkaisun luvussa 4.

3.2.2.5 Simulaatiot

Simulaatioita voidaan käyttää käyttäjätutkimuksen menetelmänä silloin, kun tutkittava käyttäjän toiminta on luonteeltaan sellaista, että todellisia käyttötilanteita on vaikea tutkijan päästä analysoimaan. Tällaisia ovat esimerkiksi monet turvallisuuskriittisiin järjestelmiin liittyvät työtehtävät, sekä tehtävät jotka liittyvät tapahtumiin, joiden käyttäytymistä ei voida ennustaa. Esimerkiksi tutkittaessa laivan komentosillalla tapahtuvaa toimintaa erilaisissa sääolosuhteissa on tehokkaampaa tehdä tutkimus simuloimalla kuin odottaa säätilan muuttuvan juuri halutun kaltaiseksi.

Lignell ym. (2003) ovat tutkineet simulaatioita mm. käyttäjävaatimusten esiin nostamisen menetelmänä. He toteavat, että simulointien avulla voidaan aktivoida käyttäjiä myös itse tuottamaan uusia ideoita informaatiojärjestelmäkehityksen

tueksi. Tutkimuksessa käytetty valmis simulaatiopeli myös mahdollisti suunnitteluprosessin läpiviemisen erittäin lyhyessä ajassa.

Simulointeja voidaan tehdä monella eri tasolla. Läheskään aina ei simulointiin tarvita valmista järjestelmää, vaan se voidaan toteuttaa myös leikkinä tai pelinä, jossa käyttäjä(t) kuvittelevat käyttötilanteen joidenkin artefaktien avulla ja näin esittävät miten tietyissä tilanteissa toimisivat. Samalla tavoin toimivat myös ns. kynä-paperisimulaatiot.

Toisaalta jos käyttäjätutkimusta tekevän tutkijan käytössä on tutkittavaa ilmiötä tarkkaan mallintava simulaattori, on koko tutkimuksen merkitys erilainen. Tällöin pystytään tutkimaan käyttäjän nykyisiä toimintamalleja todellisessa käyttötilanteessa aiheuttamatta prosessiin tahallisia poikkeustilanteita. Mielenkiintoisissa tilanteissa simulointi voidaan keskeyttää, ja tilanteesta keskustella kaikkien osapuolten kesken.

Käytettäessä simulointeja vaatimustenhankintamenetelmänä on aina kuitenkin muistettava, että simuloitu tilanne ei ole sama, kuin luonnollinen tilanne. Käyttäjät ovat aina tietoisia tilanteen aitoudesta, eikä simuloimalla voida koskaan luoda esimerkiksi samaa stressitasoa, mikä aidossa tilanteessa vaikeuttaa käyttäjän toimintaa.

3.3 Käyttäjätutkimusmenetelmän valinta monimutkaisen järjestelmän suunnittelussa

Tämän tutkimuksen tarkoitus on kartoittaa monimutkaiseen työhön liittyvien käyttäjävaatimusten määrittelyä. Siten myös menetelmän valintaa käsitellään samasta näkökulmasta. Pohdinnan aiheena on, mikä tai mitkä edellä esitellyistä käyttäjätutkimuksen menetelmistä soveltuvat luonteeltaan monimutkaisen työn tutkimiseen.

3.3.1 Tutkimusmenetelmän valinnan perusteita

Menetelmän valinnassa on otettava huomioon se, mihin koko tutkimuksella pyritään, ja mikä on kerätyn tiedon käyttötarkoitus. Esimerkiksi jonkin järjestelmän

uusimisen ja kokonaan uuden järjestelmän suunnittelun yhteydessä suoritetuilla käyttäjätutkimusmenetelmillä on eroa. Ensimmäisessä pyritään löytämään puutteita ja kehityskohteita jo olemassa olevassa järjestelmässä, jolloin nykyisen työn havainnointi näyttää tärkeää roolia. Uuden järjestelmän suunnittelussa taas tärkeää on saada tieto käyttäjän kokonaisvaltaisista tavoitteista ja selvittää mikä on hänen toimintansa päätarkoitus. Tällöin nykyisin käytössä olevan järjestelmän ei saa antaa vaikuttaa liikaa.

Myös suunnitteluprosessin vaihe vaikuttaa menetelmän valintaan. Mitä pitemmällä siinä ollaan, sitä enemmän tutkimusmenetelmä painottuu käytön havainnointiin ja sen perusteella käytäviin keskusteluihin. Suunnitteluprosessin vaihe vaikuttaa myös tarvittavan tiedon yksityiskohtaisuuden tasoon. Jos tuote on vasta idean tasolla, ei kerätyn tiedon tarvitse olla luonteeltaan täydellisen yksityiskohtaista. Myös resurssien käyttömahdollisuus on tietysti aina otettava huomioon. Tässä on pohdittava kuitenkin myös käyttäjän ajankäyttöä, sillä ei voida olettaa käyttäjien osallistuvan esimerkiksi työjärjestelmien kehitystoimiin vapaaajallaan. (Hackos & Redish 1998)

Menetelmän valinnassa on pohdittava myös käyttäjien osallistumista tutkimukseen. Saatavissa olevalla käyttäjien määrällä on merkitystä. Esimerkiksi kyselytutkimuksia ei kannata tehdä kovin pienelle käyttäjämäärälle. Toisaalta myös saatavissa olevien käyttäjien ammattitaidolla on merkitystä. Asiantuntijakäyttäjät tekevät päätelmiä työstään ja toimivat muutenkin ennakoivammin kuin noviisi-käyttäjät (Leppänen & Norros 2002). Tästä voi päätellä, että asiantuntijakäyttäjät pystyvät uusien järjestelmien kehittämisessä toimimaan paremmin suunnittelun tukena kuin noviisit.

3.3.2 Monimutkaisen työn tutkimukseen sopivat menetelmät

Monimutkaisen työn erityispiirteet on esitelty tämän työn johdannossa. Näistä tutkimusmenetelmän valintaan vaikuttavat eniten järjestelmän dynaamisuus, sen käyttöön sisältyvät suuret riskitekijät sekä työn sosiaalinen, yhteistoimintaa korostava luonne.

Taulukossa 3 on yhteenveto aikaisemmin esitellyistä käyttäjätiedon keruumenetelmistä. Kunkin menetelmän kohdalla eritelty tutkimusmenetelmän valintaan

vaikuttavia tekijöitä ja näistä on esitetty kirjoittajan subjektiiviset arviot. Tutkimuksen orientaatio kuvaa sitä, miten paljon käyttäjä pystyy vaikuttamaan tutkimuksen kulkuun. Tutkijaorientoituneissa menetelmissä tutkimuksen kulku on melko pitkälle lukittu suunnitteluvaiheessa. Tällöin joudutaan usein suorittamaan jonkinlaisia esitutkimuksia, jotta esimerkiksi kyselytutkimuksen kysymykset pystytään kohdentamaan oikein. ”Toimintatutkimuksellisuus” kuvaa sitä, miten helposti tutkimusmenetelmällä saa tietoa käyttäjistä toimijana omassa ympäristössään. Varsinkin monimutkaisen järjestelmän yhteydessä on tärkeää, että tutkimusmenetelmä tuo esiin käyttäjien työn yhteisöllisiä piirteitä ja että tutkija voi tehdä havainnoja käyttäjän tilannesidonnaisesta toiminnasta. Työläyden arvioinnissa on otettu huomioon sekä tutkimuksen valmisteluun, että kerätyn tiedon analysointiin mahdollisesti kuluva aika. Osa menetelmistä on muita työlämpiä yksinkertaisesti sen takia, että tietoa kertyy valtavan paljon, jolloin sen analyysi ja jäsentely muodostuu työlääksi prosessiksi.

Contextual Inquiry on perusteiltaan menetelmänä vahva, mutta siihen sisältyy joitain puutteita suhteessa monimutkaisten sosioteknisten järjestelmien tutkimiseen. Reaaliaikaisen turvallisuuskriittisen työn havainnointi ei salli sitä, että tutkija keskeyttää käyttäjän työn kysymyksillään. Menetelmän kehittäjät esittävät tähän ratkaisuksi käyttäjän toiminnan videointia ja videon analysointia jälkeenpäin yhdessä käyttäjän kanssa (Beyer & Holtzblatt 1998, s. 74). Mielestäni tämä kuitenkin muuttaa menetelmän luonnetta siinä määrin, että myös joitain hyötynäkökulmia on punnittava uudestaan. Videon läpikäyminen muuttaa menetelmää erityisesti käyttäjän kannalta. Aikaa kuluu enemmän ja omaa analysointia joutuu käyttämään täysin eri näkökulmasta. Tällöin tilanne muuttuu enemmänkin strukturoimatonta havainnointia tai haastattelua muistuttavaksi. Toisaalta nämä kolme ovat muutenkin menetelminä lähellä toisiaan ja voivat sopivalla tavalla käytettyinä täydentää toisiaan.

Taulukko 3. Yhteenveto käyttäjätutkimusmenetelmistä.

Menetelmä	Orientaatio	Tutkimuk- sen kesto	Tulosten laatu	Toiminta- tutkimuk- sellisuus	Työläys
Haastattelu	Tutkija	Lyhyt	Laadullinen	Mahdollinen	Raskas
Kysely	Tutkija	Lyhyt	Numeeri- nen	Heikko	Kevyt
Ryhmäme- netelmät	Tutkija/ käyttäjä	Lyhyt	Laadullinen	Mahdollinen	Kevyt
Etnografinen tutkimus	Käyttäjä	Erittäin pitkä	Laadullinen	Hyvä	Erittäin raskas
Havainnointi	Käyttäjä	Pitkä	Laadullinen	Hyvä	Raskas
Ääneen ajattelu	Käyttäjä	Vaihtelee	Laadullinen	Hyvä	Raskas
Contextual Inquiry	Käyttäjä	Pitkä	Laadullinen	Hyvä	Raskas
Simulaatio	Tutkija/ käyttäjä	Pitkä	Laadullinen	Hyvä	Raskas

Simuloinnin käyttömahdollisuudet tämän tyyppisiin järjestelmiin liittyvän työn tutkimuksessa ovat mielenkiintoiset. Usein järjestelmiin liittyy jo koulutuksenkin takia täysimittaisia simulaattoreita. Suunnitteluun tarkoitettulla kehityssimulaattorilla työskentely taas antaa käyttäjäkeskeiselle suunnittelulle täysin uusia mahdollisuuksia. Tällaisen simulaattorin avulla voidaan iteratiivisen suunnittelun periaatteita todella käyttää hyväksi. Simulointi sopii monimutkaisen työn tutkimukseen, sillä sen avulla voidaan välttää tilanteet, joissa käyttäjän toiminnan tutkinta aiheuttaisi turvallisuusriskejä käyttäjän huomion herpaantumisen vuoksi.

4. Vaatimusten muodostaminen ja esittäminen

Tässä luvussa pohditaan, mitä eri tapoja vaatimusten määrittelyyn ja esittämiseen on yleisesti olemassa, ja mitkä niistä sopisivat erityisesti monimutkaisen sosioteknisen järjestelmän vaatimusten esittämiseen. Vaatimus on määritely siten, että sillä tarkoitetaan suunnitteluprosessin aikaisessa vaiheessa järjestelmästä tehtyjä määrittelyjä, jotka rajaavat sen toteutusta jotenkin. Monenlaiset järjestelmästä luodut mallit voidaan siis tulkita vaatimuksiksi. Mallinnustekniikat on tässä luvussa jaoteltu niiden formaaliusasteen mukaan. Kunkin tekniikan formaaliusluokka perustuu Sutcliffen (2002) luokitteluun.

4.1 Suunniteltava järjestelmä kuvataan vaatimusten avulla

Vaatimusten määrittely tehdään järjestelmäsuunnittelun alkuvaiheessa. Vaatimusmäärittely voi olla ensimmäinen kirjoitettu kokonaisvaltainen kuvaus tulevan järjestelmän toiminnallisista ja muista ominaisuuksista. Määrittely on siis dokumenttina tärkeä, joten sen selkeyteen ja yksiselitteisyyteen on kiinnitettävä huomiota. Vaatimusten määrittely prosessina vaatii kommunikoinnilta paljon. Myös tähän on kiinnitettävä huomiota mallinnustekniikkaa valittaessa.

4.1.1 Millainen on hyvä kuvaus?

Kun jokin asia tai ilmiö kuvataan tai mallinnetaan jollain formaalilla menetelmällä, on kysymys reaali maailman ilmiön kuvaamisesta. Representaatiolla eli kuvauksella on systemaattinen yhteys esittämäänsä asiaan tai ilmiöön, mutta se voi toisaalta myös korostaa joitain puolia ja jättää joitain kokonaan pois. Kuvaus on aina tehty jottain tiettyä tarkoitusta varten. (Johnson ym. 1999.) Kuvaustekniikasta päätettäessä on kiinnitettävä huomiota siihen, mikä on kuvauksen tarkoitus, eli mitä tavoitteita kuvauksen avulla pyritään saavuttamaan. Kuvauksen on kyettävä nostamaan esiin mallinnettavan kohteen kannalta merkityksellinen informaatio.

Kuvauksen tekijän on mallinnustapaa valitessaan kiinnitettävä huomioitava kuvauksen joihinkin yleisiin ominaisuuksiin. Tällaisia tekijöitä ovat mm. (Johnson ym. 1999):

- **Kuvaustekniikan ilmaisuvoima:** Voiko mallinnustekniikalla mallintaa kaikki järjestelmän merkitykselliset ulottuvuudet? Se, mikä järjestelmän kuvauksessa on merkityksellistä ja tärkeää, on tietysti tapauskohtaista. Monimutkaisten sosioteknisten järjestelmien relevantteja ulottuvuuksia ovat tämän työn johdannossa luetellut piirteet.
- **Kuvauksen tarkkuus ja oikeellisuus:** Voidaanko tekniikalla esittää järjestelmän merkitykselliset ominaisuudet riittävän tarkasti? Vaatimusten määrittelyvaiheessa on otettava kantaa siihen, millä tarkkuudella järjestelmän toiminta määritellään. Samalla määritellään milloin järjestelmän voidaan sanoa olevan vaatimukset täyttävä.
- **Kuvauksen muunneltavuus:** Jo valmiita kuvauksia on jossain tapauksessa voitava jälkeensä muuttaa. Tällöin on eduksi, jos on valittu kuvaustapa, joka tukee muutoksien tekemistä. Muutokset voivat aiheutua esimerkiksi muuttuneista asiakasvaatimuksista tai jostain muusta vaihtuneesta ulkoisesta tai sisäisestä lähtökohdasta.
- **Kuvauksen sovellettavuus:** Onko kuvaus helppo hyödyntää teknistä määrittelyä tehtäessä? Kuvaus on monessa tapauksessa melko hyödytön, jos sitä ei voida teknisessä suunnittelussa käyttää hyväksi.
- **Kuvauksen ymmärtämisen helppous:** Kuvauksen hyöty syntyy sen käytöstä. Kuvaus on melko hyödytön, jos se on niin monimutkainen, että sitä ei kohtuullisella vaivalla pysty ymmärtämään. Kuvaustekniikka on valittava siten, että lopputulos on kuvauksen kohderyhmän ymmärrettävissä.
- **Kuvauksen tekemisen helppous:** Täydellinen kuvaus jonkin järjestelmän toiminnasta voi olla mahdollista tehdä, jos käytettävissä on rajattomasti aikaa ja resursseja. Tämä ei kuitenkaan ole käytännössä koskaan todellinen tilanne. Hyvä kuvaustekniikka on siinä mielessä helppokäyttöinen, että mallinnusprosessi ei vie kohtuuttomasti aikaa, ja silti lopputulos on kaikkea edellä luetelluista ominaisuuksista.

Kaikki edellä esitellyt kuvauksen ominaisuudet ovat tärkeitä mallinnettaessa monimutkaisia sosioteknisiä järjestelmiä ja mallinnusprosessin aikana ominaisuuksien välillä joudutaan tekemään kompromisseja. Johnson ym. (1999) määrittelee termin kuvaustekniikan käytettävyys, joka tarkoittaa kuvaustekniikan kykyä integroida tilanteeseen sopivalla tavalla erilaiset keskenään ristiriidassa olevat kuvausten ominaisuudet.

Kuvaustekniikkaa valitessa päädytään tyypillisesti ratkaisuun, jossa monia eri mallinnustapoja yhdistellään niin, että ne tukevat toisiaan. Siinä, missä järjestelmän jokin osa vaatii erittäin tarkkaa kuvausta, voidaan jokin toinen piirre mallintaa kevyemmällä ja nopeammin rakennettavissa ja muunneltavissa olevalla mallilla. Tämänkaltaiset kompromissit mallinnustavan eri ominaisuuksien välillä on syytä tehdä tietoisesti, jotta kaikki suunniteltavan järjestelmän merkittävät ominaisuudet tulevat mallinnetuiksi. (Johnson ym. 1999)

4.1.2 Vaatimusten määrittely on kommunikointia

Järjestelmien suunnittelu, ja etenkin suunnittelun vaatimusten määrittely, on pohjimmiltaan kommunikointia: käyttäjien ja suunnittelijoiden omaksuma tieto pyritään mahdollisimman tarkasti kommunikoimaan järjestelmän kehittäjille heille merkityksellisenä informaationa. Tavoitteena on päästä eri osapuolia sitovaan sopimukseen suunniteltavan järjestelmän vaadituista ominaisuuksista. Vaatimusmäärittely on tämän sopimuksen keskeinen osa. Eri osapuolet, jotka yrittävät päästä yhteisymmärrykseen sopimuksen sisällöstä tulevat erilaisista organisaatioista ja erilaisista taustoista. Kaikilla on kuitenkin yhteinen päämäärä. Kaikki haluavat ymmärtää ongelma-alueen mahdollisimman hyvin, jotta järjestelmän määrittelystä tulisi riittävän hyvä. Tavoitteena on myös, että kaikkien osapuolten näkemys ongelman erityispiirteistä olisi mahdollisimman samankaltainen, jotta yhteinen sopimus suunniteltavan järjestelmän sisällöstä voitaisiin tehdä. (Coughlan & Macredie 2002)

Kommunikoinnista tekee erityisen vaikeaa se, että vaatimukset ovat käsitteellisesti hankala määrittellä yksiselitteisesti. Vaatimusten luonteeseen kuuluu niiden monimutkaisuus, laaja-alaisuus ja epävakaus. Kommunikointia tarvitaan ylittämään suunnittelijan tiedon ja käyttäjän tiedon välinen semanttinen kuilu. Kuilun ylittäminen edellyttää, että kumpikin osapuoli ymmärtää sen, miten hyvin tai

huonosti toinen osapuoli ymmärtää, ja mitä hän tarkoittaa. Kommunikaation tehokkuus ja laatu onkin todettu keskeisiksi tekijöiksi suunnitteluprosessin onnistumisen kannalta. (Coughlan & Macredie 2002)

Vaatimusten kommunikointi asiakkaakke erilaisten mallien avulla on Beyerin ja Holtzblattin (1998) mukaan vaikeaa. Mallit ovat tavallaan kieltä, joka on käyttäjille ja asiakkaille vierasta, eivätkä he näin voi niiden avulla täysin ymmärtää järjestelmän suunniteltuja ominaisuuksia. Vieraan kielen lisäksi mallien ongelma on myös se, että ne esittelevät tyypillisesti joko nykyisen tai suunniteltavan järjestelmän ominaisuuksia. Käyttäjien tietämys ja kokemukset molemmista ovat hyvin vaikeasti artikuloitavissa, joten heidän on vaikea kommentoida malleja syvällisesti. Tutkijoiden mukaan käyttäjät reagoivat esimerkiksi käyttötapauksiin ainoastaan toteamalla pitävänsä niistä tai toteamalla niiden olevan totuuden vastaisia. Se, mikä nämä tunteet aiheuttaa ja miten niihin voitaisiin suunnitelmia muuttamalla vaikuttaa, jää kuitenkin aina suunnittelijoiden pohdittavaksi. Tällöin voidaan päätyä ikuiseen vaatimusten iterointisilmukkaan, jossa käyttäjä ei koskaan pysty artikuloimaan miksi jokin tietty suunnitteluratkaisu on hänen mielestään huono. On olemassa nimittäin vaara, että suunnittelija muuttaa määrittelystään juuri niitä ominaisuuksia, jotka olisivat vaikuttaneet vaatimuksen validointiin positiivisesti. (Beyer & Holtzblatt 1998)

4.1.3 Vaatimusten luokittelu

Suunniteltavalle järjestelmälle esitettäviä vaatimuksia voidaan kategorisoida moneen erilaiseen luokitteluun perustuen. Tässä työssä esitellyt vaatimusten kuvausmenetelmät on luokiteltu niiden formaaliusasteen perusteella. Kuvaustavasta riippumatta vaatimuksia kategorisoidaan kuitenkin myös muilla tavoilla.

4.1.3.1 Toiminnallisuuden perustuva luokittelu

Kun vaatimukset luokitellaan toiminnallisuutensa perusteella, jaotellaan ne toiminnallisiin ja ei-toiminnallisiin. Toiminnalliset vaatimukset käsittelevät suunniteltavan järjestelmän toimintoja ja eri toimintojen suhteita toisiinsa. Ei-toiminnalliset vaatimukset määrittelevät ominaisuuksia suunniteltavalle järjestelmälle. Nämä ominaisuudet voivat liittyä esimerkiksi ylläpidettävyyteen tai luotettavuuteen. Ei-toiminnalliset vaatimukset kuvailevat myös laitteisto- ja

ohjelmistovaatimuksia. (Schach 2002, s. 294) Ei-toiminnallisilla vaatimuksilla kuvataan tyypillisesti asiat, jotka voidaan ilmaista määrällisesti. Ei-toiminnalliset vaatimukset määrittävät toimintojen laatuominaisuuksia. (Douglas 2001)

4.1.3.2 Laatuominaisuuksien mukainen luokittelu

Standardi ISO/IEC 9126 (1991) käsittelee ohjelmistojen laatua. Se määrittelee mallin, jonka mukaan ohjelmistojen ominaisuudet voidaan jakaa kuuteen eri luokkaan, jotka sitten hierarkkisesti jakautuvat aina aliluokkiin. Ohjelmistojen yleiset laatuominaisuudet, joiden mukaan myös vaatimusmäärittelyssä esitettyjä vaatimuksia voidaan luokitella, ovat: toiminnallisuus, luotettavuus, käytettävyys, tehokkuus, ylläpidettävyys ja siirrettävyys.

4.1.3.3 Prioriteettiin perustuva luokittelu

Alustavan vaatimusten määrittelyn jälkeen vaatimusmäärittelydokumentti luovutetaan yleensä asiakkaalle priorisointia varten. Priorisointi voidaan tehdä myös loppukäyttäjän kanssa. Priorisoija arvottaa jokaisen alustavan vaatimuksen esimerkiksi asteikolla välttämätön, erittäin toivottu, toivottu ja niin edelleen. (Schach 2002.) Priorisoinnin jälkeen vaatimukset voidaan vaatimusmäärittelydokumentissa luokitella myös prioriteetin mukaan. Usein vaatimukset voivat olla keskenään ristiriitaisia. Esimerkiksi vaatimus hyvästä muistettavuudesta on usein ristiriidassa erilaisten tehokkuusvaatimusten kanssa. Varsinkin tällaisten vaatimusten toteuttamisen kannalta priorisointi on tärkeää.

4.1.3.4 Informaatiojärjestelmän rooleihin perustuva luokittelu

Vaatimuksia määriteltäessä käydään läpi tuotteen kaikki eri käyttäjäryhmät ja heille ominaiset tuotteen käyttötarkoitukset. Hannu Paunonen (1997) on väitöskirjassaan määritellyt tuotantoteollisuuden ohjausjärjestelmälle yhteensä seitsemän erilaista roolia. Nämä roolit antavat monimutkaisen sosioteknisen järjestelmän (Vicente 1999), jollainen esimerkiksi teollisuuden prosessinohjausjärjestelmä on, suunnittelijalle sopivat lähtönäkökulmat järjestelmän vaatimusten määrittelyyn.

Paunosen (1997) mukaan ohjausjärjestelmän erilaiset roolit ovat:

- **Prosessin valvonta- ja seurantatyökalu:** Prosessinohjaajat käyttävät ohjausjärjestelmää prosessin valvonnan ja seurannan välineenä. Tästä roolista johdetut vaatimukset määrittävät, mikä prosessista saatava informaatio mahdollistaa sen valvonnan ja seurannan.
- **Tukityökalu tehtävien suorittamiselle:** Ohjausjärjestelmä tukee myös muiden työtehtävien suorittamista. Järjestelmän tilasta voidaan päätellä, mitä tehtäviä on tehty, ja mitä on vielä tekemättä.
- **Häiriöiden hallinta- ja diagnosointityökalu:** Häiriötilanteissa ohjausjärjestelmän avulla voidaan selvittää häiriön syyt ja usein myös seuraukset. Ohjausjärjestelmää käytetään myös häiriötilanteen hallinnassa, jolloin sen avulla voidaan organisoida esimerkiksi häiriön eliminoimiseen tarvittavat toimenpiteet.
- **Organisaation informaationvälitystyökalu:** Ohjausjärjestelmällä on monia käyttäjiä, joiden normaaleihin työtehtäviin kuuluu seurata ohjattavan prosessin tilaa.
- **Tietämyksen hallinnan työkalu:** Ohjausjärjestelmä kerää itseensä tietoa prosessin historiasta. Samoin ohjausjärjestelmä määrää sen, miten sen käyttäjät jäsentävät tietämystään prosessin käyttäytymisestä.
- **Prosessin suorituskyvyn kehittämisen väline:** Koska ohjausjärjestelmä kertoo prosessista ja sen tilasta, voidaan sitä käyttää myös prosessin optimoinnin välineenä. Prosessin suorituskyvyn systemaattinen parantaminen vaatii aina mittausta, jotta voidaan havaita kehityskohteet, sekä kehitystoimenpiteiden jälkeen niiden vaikutukset. Ohjausjärjestelmä tarjoaa mittaukset, tiedon analysoinnin sekä visualisoinnin.
- **Oppimisen väline:** Ohjausjärjestelmä toimii monen eri työntekijän oppimista tukevana välineenä. Prosessin ohjaukseen liittyvä ammattitaito kehittyy ohjausjärjestelmän välityksellä.

Eri rooleihin liittyy osin myös eri käyttäjäryhmiä. Toisaalta sama käyttäjä voi eri tilanteissa toimia erilaisissa rooleissa. Esimerkiksi prosessin ohjaaja eli operattori, joka usein mielletään ohjausjärjestelmän pääkäyttäjäksi voi samanaikaisesti käyttää ohjausjärjestelmää lähes jokaisessa sen roolissa. Toisaalta prosessin kehittäjä eli insinööri, jonka tehtävänä on kehittää prosessin suorituskkyä, käyttää ohjausjärjestelmää lähinnä kehittämisen välineenä, sillä hän voi esimerkiksi kerätä prosessitietoa järjestelmän avulla.

Ekologisen suunnittelun teorian (Vicente 1999) mukaan juuri esitellyt järjestelmän erilaiset roolit ovat sellaisia seikkoja, joita työn analyysissa ja vaatimusten määrittelyssä tulisi pitää lähtökohtana Paunosen (1997) määrittelemät ohjausjärjestelmän roolit ovat reunaehtoja, jotka asettavat vaatimuksia järjestelmälle. Ne voidaan ymmärtää geneerisiksi vaatimuksiksi, jotka ohjausjärjestelmän tulisi aina täyttää. Ne syntyvät käyttäjän ja prosessin vuorovaikutuksesta välineen eli ohjausjärjestelmän avulla.

4.1.4 Vaatimukset validoinnin tukena

Standardi ISO/IEC 14598-1 (1999) käsittelee ohjelmistotuotteiden validointia. Validoinnilla tarkoitetaan testejä ja tarkastuksia, joiden tavoitteena on osoittaa suunnitellun tuotteen sopivuus käyttötarkoitukseensa. Validointi on tuotekehitysprosessin loppupään vaihe, joka tulee ottaa huomioon jo vaatimusten määrittelyssä, sillä vaatimusmäärittely toimii usein validoinnin pohjana. Standardi määrittelee sen, miten jokin ohjelmistotuote määritellään siten, että sen validiuden testaaminen on mahdollista.

Standardi käsittelee ISO 9126 (1991) mukaisia ohjelmistojen laatuvaatimuksia. Laatuvaatimukset ovat standardin mukaan järjestelmälle joko sisäisiä tai ulkoisia. Vaatimusten määrittelyssä lähdetään standardin mukaan liikkeelle ulkoisista vaatimuksista ja näistä pyritään johtamaan järjestelmän sisäisiä vaatimuksia. Tämä pyritään tekemään niin, että vaatimusten seuraaminen onnistuu myös toisinpäin, jolloin ulkoiset vaatimukset täytyisivät automaattisesti sisäisten täytyessä. Sisäisten vaatimusten merkitys ilmenee siinä, että niitä voidaan käyttää erilaisten välituotteiden laadun arviointiin. (ISO 1999)

ISO/IEC 14598-1:n (1999) mukaan lopputuotteen laatuun vaikuttavat vaatimukset eli järjestelmän ulkoiset vaatimukset riippuvat systeemin käyttötilanteesta. Jos kyseessä on interaktiivinen järjestelmä, ovat vaatimukset riippuvaisia loppukäyttäjän tarpeista ja tehtävistä.

4.2 Vaatimusten esittäminen

Seuraavissa kappaleissa esitetään erilaisia vaatimusten kuvaustekniikoita.

4.2.1 Epäformaalit mallit vaatimuksista

Suunnittelussa on usein perusteltua esittää järjestelmävaatimuksia jollain epäformaalilla menetelmällä. Epäformaali tarkoittaa tässä yhteydessä sitä, että esimerkiksi notaatiota ei ole määritelty, vaan esitystapa riippuu esittäjästä. Tällöin kahden eri järjestelmän vertailua ei vaatimusten perusteella voida tehdä kovin objektiivisesti. Epäformaaleja vaatimusten määrittelyjä käytetään varsinkin silloin, kun tuotekehitysympäristö on sellainen, että formaalin määrittelyn tekeminen katsotaan liian työlääksi tai aikaa vieväksi suhteessa mahdollisesti saavutettavaan etuun. Vaatimusten esittäminen epäformaalilla menetelmällä lähestyy joissain tapauksissa suunnitteluratkaisujen tuottamista. Usein mallit ottavat jo vahvasti kantaa uuden järjestelmän käyttötilanteeseen ja sen erilaisiin piirteisiin. Toisaalta näissäkään ei esitetä varsinaista ratkaisua tekniseltä kannalta, vaan ainoastaan se, miten käyttäjä ratkaisun vaikutukset kokee. Tämän takia tekniikat voidaan lukea käyttäjakeskeisen vaatimusmäärittelyn piiriin kuuluviksi.

4.2.1.1 Vaatimusluettelo

Vaatimukset suunniteltavan tuotteen käytettävyydelle voidaan esittää luettelona, jolloin sen keskeinen sisältö on järjestelmän toimintojen erittely sekä niiden vaadittava suoritustaso. Luettelon avulla esitetyille käytettävyyksivaatimuksille on ominaista, että niillä esitetään järjestelmän päätoiminnot käyttäjän näkökulmasta. Samoin määritellään metriikat toimintojen suorittamiselle. Nämä metriikat valitaan tyypillisesti Nielsenin (1993) käytettävyyden määritelmästä, jolloin ne ovat: opittavuus, tehokkuus, muistettavuus, virheiden määrä sekä käyttäjän subjektiivinen tyytyväisyys.

Kun järjestelmän toiminta analysoidaan näiden ominaisuuksien valossa, voidaan jokaiselle niistä määritellä tavoitetaso. Esimerkiksi opittavuutta voidaan kuvata vaatimuksella, joka määrittelee miten nopeasti uuden käyttäjän pitäisi oppia löytämään toimintoja. Virheiden määrän vaatimus taas voidaan kuvata luokittelemalla virheitä niiden vaikutuksen vakavuuden suhteen ja asettamalla eri luokkien virheille omat tavoitetasot. (Preece ym. 1994)

Kun vaatimukset esitetään listana, täytyy lista priorisoida. Priorisointi on tehtävä sen takia, että toteutusprojekteilla ei koskaan ole rajatonta budjettia. (Sutcliffe 2002) Joko vaatimukset esitetään tärkeysjärjestyksessä, tai jos ne on jäsennelty jotenkin muuten, voidaan niille esittää myös prioriteetti-arvo. Toisaalta jos vaatimukset koskevat laajaa järjestelmää, ei niitä voida missään tapauksessa esittää ainoastaan prioriteetin mukaan järjestettynä. Tällöin vaatimukset voidaan jäsentää järjestelmän kokonaisuuksien mukaan. Myös käyttäjäryhmiä voidaan käyttää vaatimusten jäsentelyn keinona.

Vaatimusten esittämisessä luettelon avulla on se huono puoli, että luettelo antaa järjestelmästä hyvin atomistisen kuvan (Kujala 2002). Listan avulla voidaan yksittäisiä asioita kuvata melko tarkkaan, mutta monimutkaisen järjestelmän vaatimusten määrittelyä vaatimuslistan avulla on erittäin vaikea tehdä. Vaikeus kasvaa suhteessa suunniteltavan järjestelmän laaja-alaisuuteen.

Beyer ja Holtzblatt (1998) toteavat perinteisten tekstuaalisten vaatimusmäärittelydokumenttien olevan erityisen huonoja suunnitteluratkaisujen kommunikoinnissa niin suunnittelutiimien sisällä kuin asiakkaiden ja käyttäjienkin suuntaan. Kun vaatimusmäärittely dokumentoidaan tiimin sisällä kirjoitettuna tekstinä, unohtuu siitä helposti käyttäjän näkökulma tuotteeseen. Tällöin jaotellaan koko systeemi sen teknisten osakokonaisuuksien mukaan ja näin muodostuu jopa suunnittelijoillekin vaikeaksi tehtäväksi tulkita eri vaatimusten suhteita ja vaikutuksia toisiinsa.

Myös se, että käytettävyyksivaatimukset ovat usein luonteeltaan ristiriitaisia, vaikeuttaa järjestelmän hahmottamista vaatimuslistan avulla. Tällaisia ristiriitaisia vaatimuksia ovat esimerkiksi käytettävyyden määritelmään sisältyvät käytön tehokkuus ja toisaalta järjestelmän helppo opittavuus. Usein tehokkuus saavutetaan esimerkiksi oikopolkujen avulla, jotka sinänsä ovat varsinkin noviisikäyttäjille vaikeita oppia, jolloin esitetyt vaatimukset joutuvat keskenään ristiriitaan.

4.2.1.2 Skenaariot

Skenaariot ovat tekstuaalisesti esitettyjä käyttötilanteiden kuvauksia. Skenaario on pieni tarina. Skenaarioita voidaan luokitella niiden pituuden ja tarkkuuden mukaan. Vaatimusmäärittelyvaiheen skenaariot kertovat suunniteltavan tuotteen tulevista käyttötilanteista. Ne ovat siis tulevaisuusskenaarioita. (Hackos & Redish 1998)

Lyhimmillään skenaario on yhden tapahtuman tai toimintasarjan yksiselitteinen kuvaus. Skenaario kasvaa ja sen tarkkuus lisääntyy, kun pituutta lisätään kuvailumalla käyttötilannetta tarkemmin. Nielsenin (1993) mukaan skenaario on tiivistetty kuvaus yksittäisestä käyttäjästä, joka käyttää tiettyä järjestelmää saavuttaakseen tietyn tavoitteen, tietyissä olosuhteissa tietyn ajanjakson aikana. Skenaario alkaa alkutilanteesta, kertoo käyttäjän interaktiosta välineen kanssa ja päättyy lopputilanteeseen. Väline tarkoittaa tässä tilanteessa suunniteltavaa järjestelmää. Skenaario sisältää kuvailua vain järjestelmän käytön kannalta oleellisista seikoista.

Vaikka skenaario kertoo vain yhden tapahtuman tai toimintasarjan suorittamisesta, antaa se kuitenkin vaatimuslistaa monitahoisemman kuvan tuotteen käytöstä. Skenaarion avulla on mahdollista myös määritellä eri toimintojen suhteita toisiinsa sekä sitä, minkälaista informaatiota käyttöliittymä eri tilanteissa antaa käyttäjälle. Skenaarion avulla pystytään siis myös ilmentämään joitain järjestelmän adaptiivisia ominaisuuksia. Toisaalta skenaariota on vaikea kirjoittaa täysin yksiselitteisesti.

Skenaario on melko tehokas kommunikoinnin väline. Sen avulla on mahdollista hahmottaa myös käyttötilanne ja siihen liittyvät taustavaikutteet. Skenaarion avulla on mahdollista välittää tietoa suunniteltavan järjestelmän käytöstä myös ei-asiantuntijaryhmille, joille esimerkiksi vaatimuslista antaa aivan liian suppean kuvan käyttötilanteista.

4.2.1.3 Storyboardit eli kuvakäsikirjoitukset

Storyboard (suom. kuvakäsikirjoitus) on kuvitettu kertomus suunniteltavan järjestelmän käyttötilanteesta. Se eroaa skenaarioista oikeastaan vain siinä, että se sisältää myös kuvamateriaalia. Näin voidaan tehostaa vaatimusten kommunikointia niin suunnittelutiimin sisällä kuin käyttäjienkin suuntaan. (Hackos &

Redish 1998) Yksi storyboard kuvaa aina yhden tietyn tehtävän suorittamistilannetta (Beyer & Holtzblatt 1998). Koska myös monimutkaisten järjestelmien käyttö perustuu useimmiten visuaaliseen käyttöön, voidaan storyboardin avulla esittää joitain juuri visuaalisia vaatimuksia tehokkaammin kuin pelkän tekstuaalisen tiedon avulla.

Storyboardin kuvat voivat olla kuvia käyttötilanteesta tai sitten karkeita luonnoksia käyttöliittymästä. Ne voidaan jakaa kahteen luokkaan sen mukaan, kumpia kuvia käytetään. Kun kuvat ovat käyttötilanteesta, puhutaan käyttö-storyboardista, ja kun kuvat ovat käyttöliittymästä, puhutaan tehtävä-storyboardista. Storyboardeja käytetään yleensä skenaarion yhteydessä, eli kuva ja teksti täydentävät toisiaan. Tällöin saatetaan myös käyttää termiä kehittynyt skenaario. (Hackos & Redish 1998)

Storyboardin avulla voidaan viestiä monimutkaisiin sosioteknisiin järjestelmiin kuuluvia käytön sosiaalisia näkökulmia. Kuvan avulla voidaan kertoa, kuinka käyttäjät toimivat yhteistyössä käytännön tilanteessa.

Storyboardin tekeminen vaatii jonkin verran aikaa ja vaivaa, mutta se on kuitenkin pelkkää skenaariota ilmaisuvoimaisempi tekniikka, sillä kuva kertoo useimmissa tapauksissa enemmän ja tarkemmin kuin pelkkä teksti. Toisaalta storyboard on vaatimusten esittämisen välineenä jo kaikkein epäformaaleimmasta päästä. Sitä on mahdotonta tehdä yksiselitteisesti. Eri ihmiset tulkitsevat kuvia aina eri tavoilla, joten storyboard toimii paremminkin vaatimusten täydentäjänä ja järjestelmän suunnittelun toiminnan viestittäjänä.

4.2.1.4 Prototyypit

Prototyyppi on nopeasti rakennettu ja helposti muutettavissa oleva luonnos tai simulaatio suunniteltavan järjestelmän käyttöliittymästä tai sen osasta. Se on käsitteenä yhtäläinen fyysisen mallin sekä mock-upin kanssa. (Hackos & Redish 1998)

Vaatimusten esittäminen prototyypin avulla on jo lähes yhtä paljon järjestelmän suunnittelua kuin sille asetettujen vaatimusten esittämistä. Joissain tilanteissa järjestelmän vaatimukset on helpompi kommunikoida eteenpäin prototyypin kuin jonkin formaalin vaatimusmäärittelyn avulla. Tilaus-toimitussuhteessa ole-

vat yritykset kommunikoivat toimitustarjouksen yhteydessä tyypillisesti prototyypin avulla. Irrallinen prototyyppi voidaan rakentaa nopeasti ja se toimii tarjoustilanteessa demonstrointitarkoituksessa. Sen avulla esitetään asiakkaalle, mitä ollaan valmiita tekemään. Samalla asetetaan itselle toimittajana vaatimus siitä, millainen järjestelmä tulee todellisuudessa olemaan. Jos prototyyppi hyväksytään vaatimukseksi tulevan järjestelmän toiminnalle, voidaan sitä kohdella järjestelmän vaatimusmäärittelynä.

Prototyypit voidaan luokitella niiden kypsyyssasteen mukaan. Kypsyyssastoltaan alhaisimmat prototyypit ovat karkeita kynällä piirrettyjä luonnoksia suunniteltavan järjestelmän käyttöliittymästä. Korkeimman tason prototyypit taas ovat interaktiivisia ja tietoteknisesti toteutettuja, jolloin niitä saattaa ulkonäön perusteella olla vaikea erottaa lopullisesta tuotteesta. Prototyyppi ei kuitenkaan sisällä mitään varsinaista toimintaa, eli jos ohjausjärjestelmän interaktiivista prototyyppiä käyttää, ei mitään ohjaustoimenpiteitä tapahdu oikeassa prosessissa. (Hackos & Redish 1998)

Prototyyppi on aina supistettu versio valmiista tuotteesta. Prototyypit voidaan jakaa kahteen luokkaan myös se perusteella, miten supistaminen on tehty (Nielsen 1993, Hackos & Redish 1998):

- **Horisontaalinen prototyyppi:** Jos prototyypillä halutaan esittää vaatimuksia esimerkiksi käyttöliittymän konsistenssille tai navigoinnille, rakennetaan se horisontaaliseksi. Tällöin prototypoidaan kokonaisuovellukseen vaikuttavat asiat. Horisontaaliseen prototyyppiin kuuluu järjestelmän koko käyttöliittymä, mutta mitään toiminnallisuutta ei ole tarkennettu.
- **Vertikaalinen prototyyppi:** Jos halutaan esittää vaatimuksia toiminnallisuudelle tai esimerkiksi toimintosarjojen loogisuudelle, voidaan rakentaa vertikaalinen prototyyppi. Tällöin prototypoidaan tapahtumaketjuja tai muita loogisia kokonaisuuksia kuten esimerkiksi käytön aloitusta, tuotetiedonhakua tai ostotapahtumaa.

Prototyypintä tukevat monet tekniset työkalut. Prototyyppien rakentamiseen voidaan käyttää ohjelmointikieliä, jolloin interaktiota ja toiminnallisuutta pystytään mallintamaan melko laajasti. Tällöin ovat kyseessä enimmäkseen vertikaali-

set prototyypit. Erilaiset multimediatyökalut taas tukevat hyvin horisontaalisten prototyyppien käyttöä. (Sutcliffe 2002)

Prototyypointi on tekniikka, joka mahdollistaa vaatimusten validoinnin käyttäjillä ja asiakkaalla. Käyttäjän kannalta se on luultavasti myös helpoimmin ymmärrettävissä oleva vaatimusten esittämisen väline. Prototyyppi on siinä mielessä kommunikatiivinen, että käyttäjän ei tarvitse kuvitella, miltä esiteltyt vaatimukset täyttävä tuote esimerkiksi näyttäisi tai tuntuisi. Käyttäjä kokeilee tuotetta itse, tai seuraa, kun joku muu esittelee sitä. Näin hän voi perustaa oman mielipiteensä tuotteen käytölle, eikä hänen tarvitse tehdä käsitteellistä loikkaa vaatimuksista suunnitteluratkaisuihin, sillä se on tehty hänen puolestaan prototyypin suunnitteluvaiheessa.

Prototyypoinnin huonot puolet liittyvät myös käyttäjän kanssa tehtävään validointiin. Sutcliffe (2002) toteaa, että loppukäyttäjä voi häkeltyä tuotteen näennäisestä valmiusasteesta, eikä näin pysty antamaan oikeaa arviota vaatimusten kelpoisuudesta. Samoin esimerkiksi visuaalinen ulkoasu voi hämätä käyttäjää niin, että hän ei pysty kiinnittämään huomiota toiminnallisiin näkökulmiin eikä arvioimaan niitä.

Huonoimmillaan vaatimusten esittäminen prototyypin avulla on resurssien tuhlausta. Prototyypin rakentaminen on kuitenkin aina suhteellisen työlästä ja aikaa vievää, joten vaatimusten ”kokeilu” prototyyppien avulla on jonkinasteista tuhlausta. Suunnittelijan on siis oltava melko varma siitä, että hänen omaksumansa vaatimukset ovat päteviä, ennen kuin prototyyppien rakentamista kannattaa aloittaa. (Sutcliffe 2002)

4.2.2 Puoliformaalit mallit vaatimuksista

Tässä kappaleessa esiteltäviä kaavioihin perustuvia vaatimusten esittämistekniikoita voidaan joissain asiayhteyksissä pitää hyvinkin formaaleina menetelminä. Tämä on yleistä varsinkin teollisessa ympäristössä. Koska akateemisissa piireissä nämä kuitenkin luetaan yleisimmin puoliformaaleiksi tekniikoiksi, tehdään sama jako myös tässä tutkimuksessa ja formaalit menetelmät esitellään lyhyesti omassa kappaleessaan.

Vaatimusten esittämisen menetelmät käydään tässä kappaleessa läpi esimerkkien avulla. Esimerkit ovat suurimmaksi osin peräisin alkuperäiskirjallisuudesta, mutta osa on tehty myös kirjoittajan toimesta alkuperäiskirjallisuuden esimerkkejä soveltamalla.

4.2.2.1 Contextual Designin kuvaustapoja

Contextual Design (CD) on suunnitteluprosessimalli, joka perustuu mahdollisimman tarkalle käyttäjän toiminnan tuntemiselle. CD on alun perin kehitetty erilaisten työn tukijärjestelmien kehittämiseen. Tämän takia malleissa puhutaan aina työntekijästä, työtehtävistä jne. Nämä vastaavat käsitteinä käyttäjää ja käyttäjän tehtäviä.

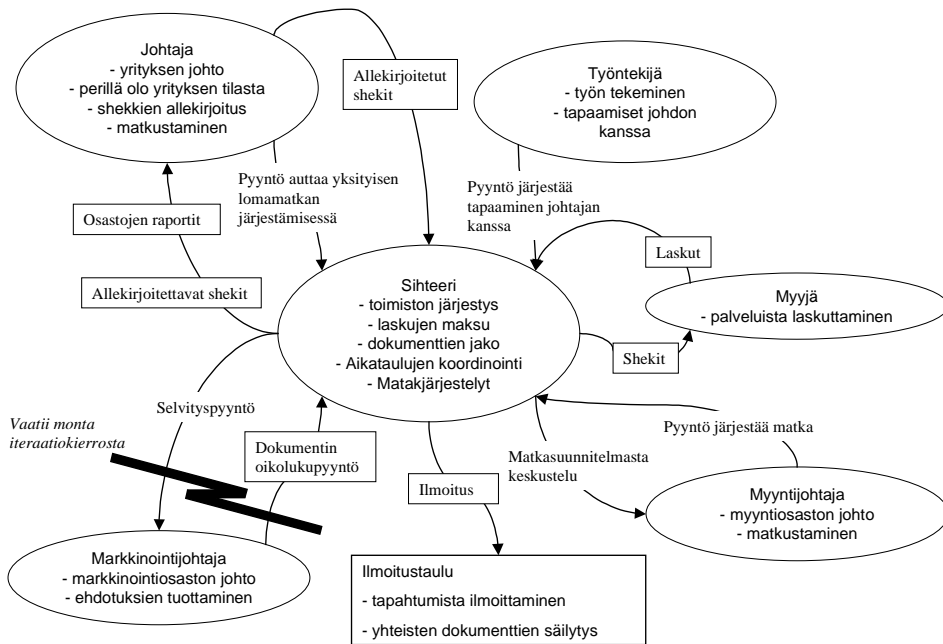
Seuraavia mallinnustekniikoita käytetään CD-suunnitteluprosessin niissä vaiheissa, joissa sekä mallinetaan asioiden nykyistä tilaa, että suunnitellaan tulevan järjestelmän toimintaa eli määritellään sille vaatimuksia. Käyttöympäristömalli esitellään viimeisenä. Se toimii vaatimusmäärittelynä siinä mielessä, että se on pohja suunnittelulle ja se validoidaan käyttäjillä ja siihen palataan aina, jos suunnittelussa huomataan tehdyn virheitä. Muut mallit toimivat tukena käyttöympäristömallin muodostamiselle ja muodostavat tärkeän osan suunniteltavan järjestelmän toiminnan mallinnuksesta.

Vuorovaikutusmallin (Kuva 5) avulla tunnistetaan yksilöt ja roolit, jotka liittyvät työn tekemiseen. Jokainen henkilöryhmä piirretään malliin kuplana, johon merkitään roolin työtehtävä. Samalla mallinetaan myös yksittäisten roolien vastuut. Tämä tehdään listana siitä, mitä kultakin roolilta odotetaan. Jokaiseen kuplana piirrettävään rooliin liitetään mallissa vastuulista. Malliin piirretään myös henkilöryhmät, joilla on yhteiset tavoitteet tai tehtävät. Tämän ryhmän ulkopuolella ryhmästä ei tunnisteta yksilöitä, vaan siitä puhutaan yhtenäisenä kokonaisuutena: ”Markkinoinnin mielestä tuotteeseen pitää lisätä ominaisuus x.” Tällöin markkinointi on henkilöstöryhmä, jolla on yhteiset tavoitteet ja tehtävät ja se mallinetaan vuorovaikutusmalliin. Vuorovaikutusmalli esittää ihmisten välisen kommunikaation, jonka tarkoituksena on auttaa suunnitellun työn loppuun viemisessä. Kommunikaation kulku voi olla epämuodollista keskustelua tai koordinoimista. Toisaalta se voi olla myös erilaisten artefaktien lähettämistä ja vastaanottamista. Kommunikaation kulku esitetään mallissa nuolina työntekijöiden välillä. (Beyer & Holzblatt 1998, s. 91)

Artefaktit ovat työhön liittyviä esineitä ja kappaleita. Artefakti voi olla fyysinen, esimerkiksi jokin muodollinen dokumentti tai epämuodollinen muistilappu. Toisaalta artefakti voi olla myös käsitteellinen. Jos esimerkiksi jollain keskustelulla voidaan ajatella olevan osallistujia, historia ja muita ominaisuuksia, ja sen ole-massaolo on riippumaton ulkopuolisista tekijöistä, voidaan sitä pitää artefaktina. Artefaktit esitetään vuorovaikutusmallissa pieninä kommunikaationuoliin liittyvinä laatikoina. Jos mahdollista, artefaktista esitetään myös sen välittämiseen liittyvä mekanismi. Kommunikaationuoleen voidaan liittää selvennyksiä myös ilman laatikkoa. Tällöin on kyse kommunikaation kulkuun liittyvästä kommuni-kaation aiheesta tai toiminnallisesta suorittamisesta. Tämä voisi esimerkiksi olla keskustelu kokouksen järjestämisestä tai jokin avunpyyntö oman tehtävän suorittamisessa. Fyysiset tilat, joita työntekijät tarvitsevat työnsä suorittamisessa mallinnetaan myös vuorovaikutusmallissa. Tällaisia ovat esimerkiksi neuvotte-luhuoneet ja jotkin julkiset tilat, kuten kahvihuoneet. Paikat piirretään isoina neliöinä, joihin merkitään tilan nimi. (Beyer & Holzblatt 1998, s. 91)

Katkokset ja ongelmat kommunikaatiossa mallinnetaan myös vuorovaikutus-mallissa. Ne piirretään kommunikaationuolet katkaisevina salamoina. Jos käy-tössä on värejä, esitetään salamat punaisina. (Beyer & Holzblatt 1998, s. 91)

Vuorovaikutusmallin avulla esitetään vaatimuksia työhön liittyville eri rooleille. Kuva 5 esittelee sihteerin työn nykytilaa, mutta samankaltaisen mallin avulla esitettäisiin vaatimukset työn tuleville toimintoille uuden järjestelmän puitteissa. Vuorovaikutusmallin avulla voidaan esittää vaatimuksia suunniteltavalle järjes-telmälle, sillä se määrittelee mitä tehtäviä, kommunikaatiota ja artefakteja jär-jestelmän suunnittelussa tulee ottaa huomioon.

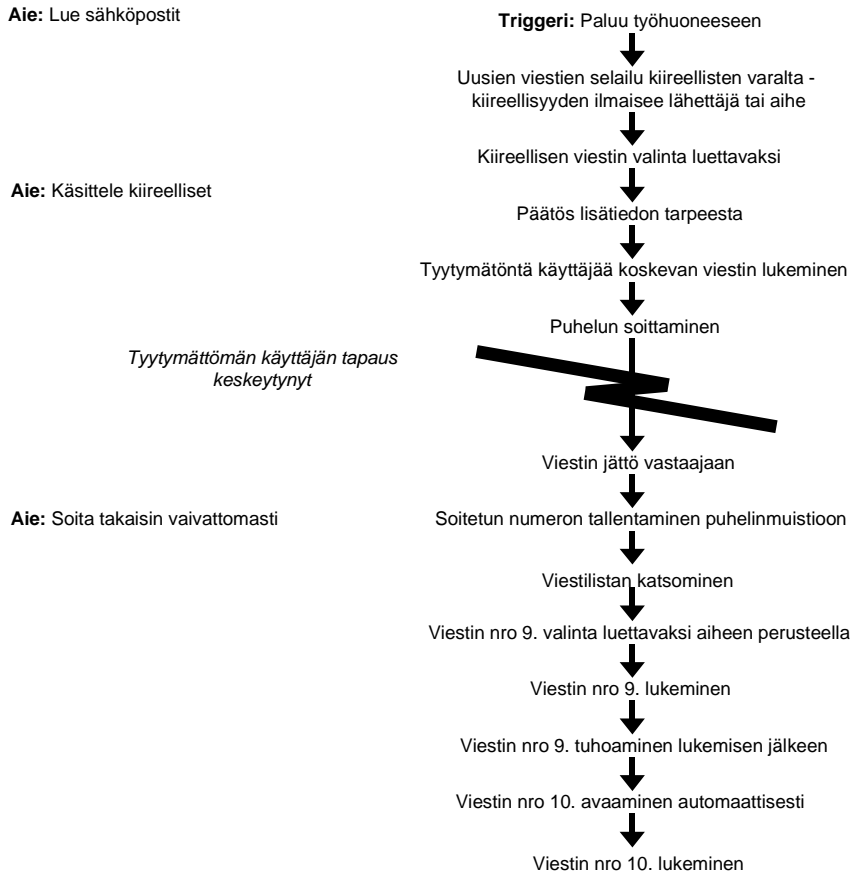


Kuva 5. Vuorovaikutusmalli sihteerin työstä. Sihteeri toimii koko osaston keskustyöntekijänä, jonka kautta hoidetaan paljon asioita. (Beyer & Holzblatt 1998, s. 92)

Sekvenssimallin (Kuva 6) avulla määritellään sen nimen mukaisesti työn suorittamiseen liittyviä peräkkäisiä toimintoja eli sekvenssejä. Sekvenssimallissa mallinnetaan yksittäisen työntekijän tai työntekijäryhmän rutiineja. Sekvenssimallissa mallinnetaan toiminnon tavoitteet aikeina. Aikeet voivat olla primäärejä tai sekundäärisiä. Sekvenssi alkaa aina primäärillä aikeella, mutta sen aikana voi ilmetä myös sekundäärisiä aikeita. Aikeen jälkeen tarvitaan jokin laukaiseva tekijä, joka saa henkilön aloittamaan sekvenssin. Nämä tekijät ovat sekvenssimallin laukaisijoita. Laukaisija on jokin käyttäjän ympäristöstään havainnoima tekijä, joka saa hänet toimimaan. Laukaisija voi olla esimerkiksi saapuvan sähköpostin merkkiäni. Askeleet ovat sekvenssin oletettu toteuttamistapa. Askeleet mallinnetaan kirjoitettuina lauseina. Askeleiden abstraktiuden taso riippuu siitä, missä vaiheessa mallinnusprosessia ollaan. Jos mallinnetaan työn nykyistä suoritusta, ovat askeleet työntekijän tekemiä konkreettisia ratkaisuja. Jos taas kyseessä on vaatimusmäärittely, mallinnetaan työtehtävän kannalta oleelliset askeleet, jolloin malli on abstraktimpi kuin edellä mainitussa tapauksessa. Askeleiden järjestys mallinnetaan pienten nuolten avulla. Nuolet paljastavat askelten suoritusjärjestyksen. Nuolet voivat olla myös haarautuvia. Katkokset ja ongel-

mat sekvensseissä mallinnetaan järjestysnuolet katkaisevina salamoina. Jos käytössä on värejä, esitetään salamat punaisina. (Beyer & Holzblatt 1998, s. 99)

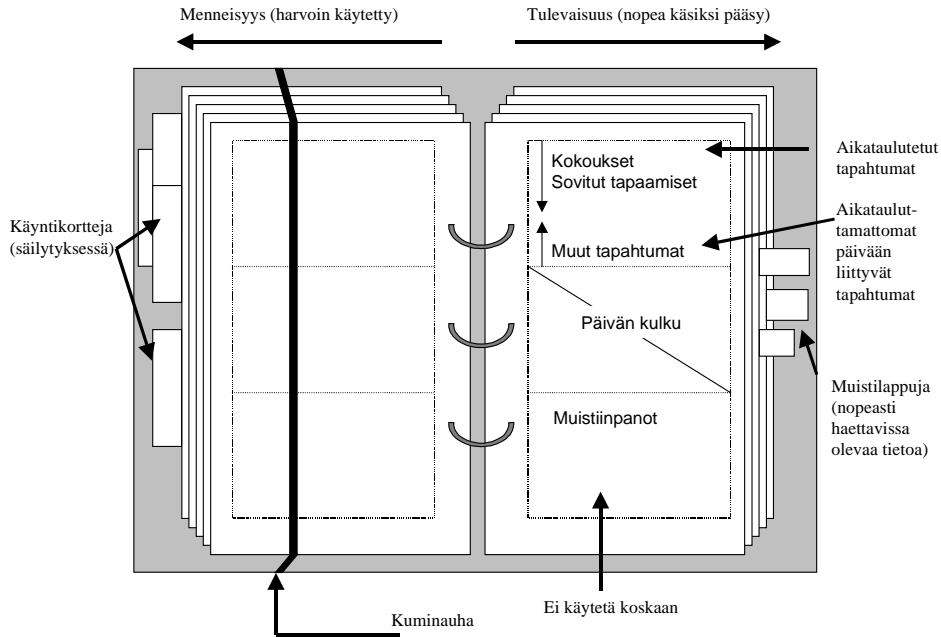
Vaatimusten määrittelyn kannalta sekvenssimalli on tehtäväanalyysin tapainen lähestymistapa ihmisten työhön. Sekvenssimallin avulla esitetyt vaatimukset kertovat siitä, mikä olisi rationaalinen tapa suorittaa tietty tehtävä. Suunniteltavan järjestelmän tulee tukea tätä rationaalista eli hyvää toimintatapaa.



Kuva 6. Sekvenssimalli sähköpostin käsittelemisestä (Beyer & Holzblatt 1998, s. 98).

Artefaktimallissa (Kuva 7) mallinnetaan se tieto, jonka jokin artefakti sisältää. Artefakti tarkoittaa sananmukaisesti ihmiskäden työtä. Se on siis esine tai asia, jonka käyttäjä eli tässä tapauksessa työntekijä on tuottanut. Artefaktit ovat rele-

vanteja järjestelmäsuunnittelussa, sillä ne paljastavat paljon sitä tietoa, mitä käyttäjät pitävät oleellisena päätöksentekonsa kannalta. Esimerkiksi prosessinohjaaja kirjoittaa muistilapulle tiedot, jotka kokee tarpeelliseksi esimerkiksi keskustella läpi vuoron vaihtuessa. Kun muistilapun sisältämä tieto ja siihen sisältyvä jäsen-tely otetaan huomioon uutta prosessinohjausjärjestelmää suunniteltaessa, voidaan systeemi suunnitella niin, että muistilappujen kirjoittamisen tarve vähenee.



Kuva 7. Artefaktimalli kalenterin käytöstä (Beyer & Holzblatt 1998, s. 104).

Artefakti on aina objekti, joten sen osat, jotka ovat käytölle ominaisia, mallinetaan myös. Osien suhteet toisiinsa eli artefaktin rakenne on myös tarpeen mallintaa. Samoin artefaktiin sisältyvät erilaiset huomiomerkinnot kuten alleviivaukset mallinetaan, kuten myös objektin ulkoasu. Usein käyttäjän omaksi hyödykseen tuottamiin esineisiin sisältyy jokin käsitteellinen erottelutapa. Esimerkiksi kalenterissa mennyttä aikaa kuvastavat käytetyt sivut, jotka on käännetty. Tämän tapaiset käsitteellistykset on myös tarpeen saada mallinnettua. Artefaktista sisällytetään malliin vielä sen käyttöä koskevaa tietoa kuten esimerkiksi se, missä tilanteessa artefakti on luotu ja miten sitä käytetään. Samoin kuin edellisissäkin malleissa artefaktiin sisältyvät ongelmat esitetään salamojen avulla. (Beyer & Holzblatt 1998, s. 105)

Artefaktimallissa mallinnustapa riippuu hyvin paljon mallinnuksen kohteesta. Reaalimaailman objekteille ei ole voitu luoda samanlaista symbolikirjastoa kuin aiemmin esitetyillä malleilla oli. Artefakti pyritään esittämään mahdollisimman hyvin sellaisena, kun käyttäjä sen näkee (Kuva 7).

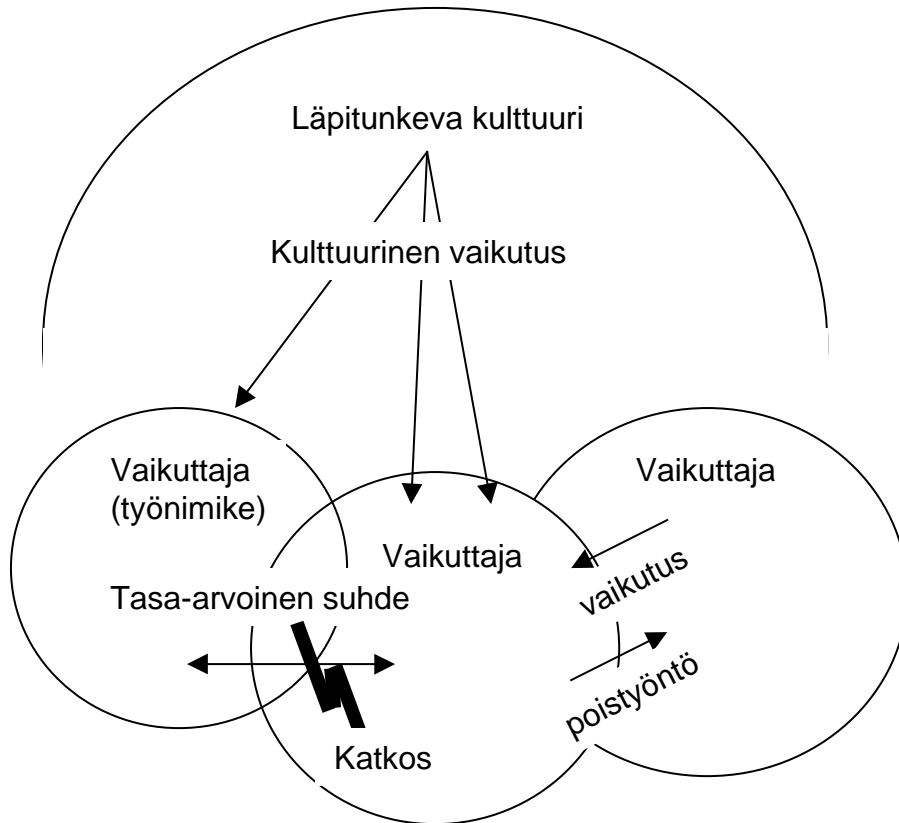
Artefaktimallit asettavat vaatimuksia suunniteltavalle järjestelmälle. Ne paljastavat, miten työntekijät jäsentävät työtehtäviään ja esimerkiksi ajankäyttöään. Artefaktit myös kertovat totutuista työtavoista. Esimerkiksi kuvan X kalenteri kertoo, että sen käyttäjä erottelee kalenterinsa avulla tulevaisuuden ja menneisyyden tapahtumat. Jos suunnittelun kohteena olisi tietokoneistettu kalenteri, esittäisi artefaktimalli vaatimuksen samantyyppisen jaottelun mahdollisuuden tärkeydestä suunniteltavassa järjestelmässä.

Kulttuurimallin (Kuva 8) avulla mallinnetaan työyhteisön kulttuuria. Kulttuuri määrää työpaikan odotukset, toiveet, käytännöt ja arvot. Kulttuuri on sitä, miten ihmiset eli työntekijät ylipäätään suhtautuvat työhönsä. (Beyer & Holtzblatt 1998, s. 107)

Kulttuurimallissa vaikuttajat mallinnetaan kuplina (Kuva 8). Vaikuttajat ovat joko yksilöitä tai muodollisia ryhmiä, jotka vaikuttavat organisaatiossa. Vaikuttaja voi olla myös epämuodollinen ryhmä, joka koostuu henkilöistä, jotka eivät periaatteessa toimi yhdessä, mutta jotka ihmiset mieltävät jostain syystä kuuluvan tiettyyn ryhmään. Vaikuttaja voi olla myös ulkoinen, esimerkiksi asiakas. Vaikuttajien kuplien päällekkäinen pinta-ala kertoo vaikutuksen merkityksestä. Jos vaikutussuhde on erittäin vahva, piirretään vaikutuksen alaisena oleva vaikuttaja kuplana kokonaan vaikuttajan kuplan sisään. (Beyer & Holtzblatt 1998, s. 109)

Vaikutus mallinnetaan kulttuurimaalissa nuolena (Kuva 8). Nuoli kuvaa vaikutuksen suuntaa. Nuolilla voidaan myös kuvata se, kohdistuuko vaikutus kattavasti läpi koko organisaation vai vain tiettyihin osiin sitä. Käytännössä vaikutus on harvoin yksisuuntaista, eli yksisuuntaiseenkin vaikutusnuoleen sisältyy jonkinlainen poistyyöntövaikutus. Katkoksia ja muita työtä häiritseviä tekijöitä kuvataan tässäkin mallissa salamalla. Koska vaikutteet ovat usein työtä jollain tavalla rajoittavia tekijöitä, käytetään salamaa ainoastaan silloin, kun vaikutus voidaan katsoa negatiiviseksi. (Beyer & Holtzblatt 1998, s. 109–110)

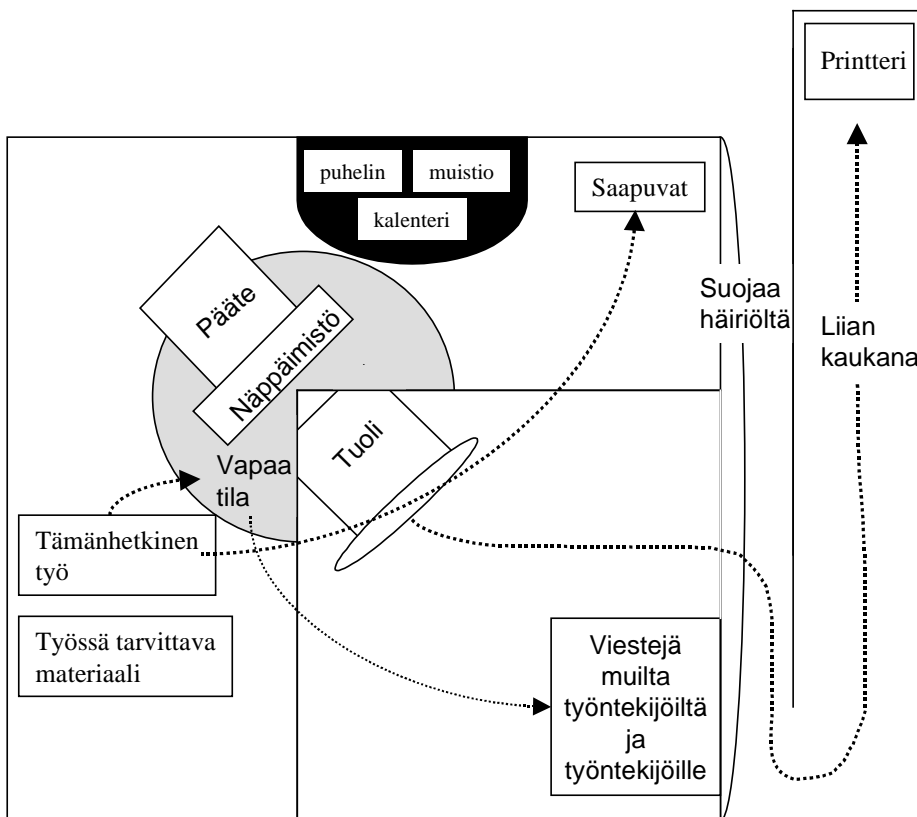
Ei ole epäselvää, että työkuulttuuri asettaa vaatimuksia työssä käytettäville järjestelmille. Beyerin ja Holtzblattin (1998) kulttuurimalli on kuitenkin hyvin abstrakti kuvaus työssä vaikuttavasta kulttuurista, joten sen ei voida ajatella esittävän näitä vaatimuksia kovin tarkasti.



Kuva 8. Yleinen malli kulttuurimallista (Beyer & Holtzblatt 1998, s. 109).

Fyysinen malli (Kuva 9) kuvaa työn fyysistä ympäristöä. Ympäristö voi joko tukea työtä tai muodostaa sille esteitä ja rajoitteita. (Beyer & Holtzblatt 1998, s. 115) Useimmiten voisi ajatella, että molemmat ilmiöt tapahtuvat yhtä aikaa. Ympäristön jotkut ominaisuudet tukevat työn tekemistä ja toiset taas asettavat reunaehdoja.

Fyysiseen malliin rakennetaan kuva ympäristön niistä seikoista, joilla on merkitystä työn suorittamisen kannalta. Tällaisia asioita ovat ensinnäkin fyysiset tilat eli työhuoneet ja -pisteet, toimistot ja kahvitilat. Malli kuvaa sitä, onko tila suuri vai pieni ja onko se sekundäärinen vai primääri työpiste. Malli kuvaa myös yksityisyyden tason ja vapaan tilan määrän. Samoin malliin kuuluu tilaa rajoittavat rakenteet eli seinät ja kaapit sekä muut isot esineet. Työntekijän liikkuminen työtilassa on myös oleellista. Työhön liittyvät tarpeelliset työntekijän siirtymiset eri tilojen välillä tai tilan sisällä mallinnetaan. Liikkuminen kuvataan katkovii-valla. Fyysiseen malliin kuuluvat myös työn tekemiseen tarvittavat työkalut. Nämä voivat olla tietokoneita, puhelimia ja mitä tahansa muita tarvittavia apuvälineitä. Fyysisessä mallissa työkalut sijoitetaan nykyistä tilaa vastaaville paikoilleen. Työntekijän käyttämät artefaktit sijoitetaan niille paikoille, jossa hän niitä säilyttää. Työtilan aiheuttamia rajoitteita mallinnetaan salamoilla. (Beyer & Holtzblatt 1998, s. 117)



Kuva 9. Fyysinen malli sihteerin työtilasta (Beyer & Holtzblatt 1998, s. 121).

Fyysinen malli viestii vaatimuksia, jotka johtuvat nimenomaan fyysisestä tilasta ja ympäristöstä, jossa järjestelmää käytetään. Suunniteltavasta järjestelmästä riippuen nämä voivat olla joko ympäristön asettamia vaatimuksia tai ympäristölle asetettavia vaatimuksia. Monimutkaisen sosioteknisen järjestelmän suunnittelussa voidaan ottaa kantaa myös käyttöympäristön suunnitteluun. Esimerkiksi suunniteltaessa teollisuuslaitoksen automaatiojärjestelmää voidaan asettaa vaatimuksia valvomotilalle käyttöympäristönä. Nämä vaatimukset voidaan esittää fyysisen mallin avulla, jolloin pystytään ottamaan kantaa esimerkiksi eri työpisteiden välisiin etäisyyksiin tai tarvittavien työpisteiden ja erilaisten työn apuvälineiden määrään.

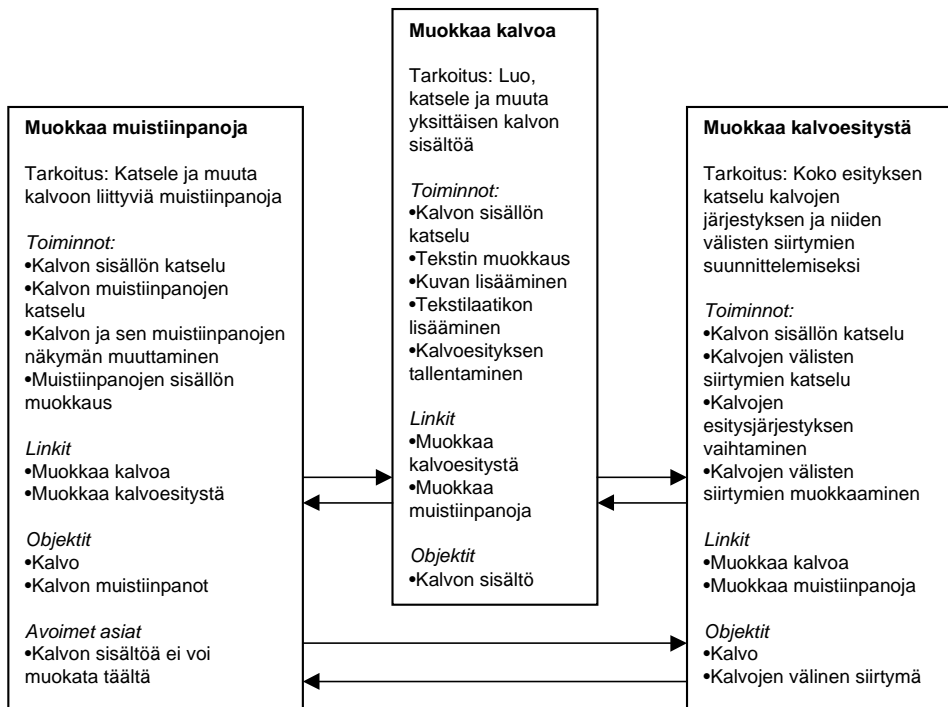
Malli, jota käytetään CD-metodologiassa yksityiskohtaisen järjestelmäsuunnittelun pohjana, on nimeltään *käyttöympäristömalli* (eng. user environment design) (Kuva 10). Siinä kuvataan uusi järjeistetty työtapa. Työtavan järjeistämällä tarkoitetaan sitä, että työtehtävien järjeistys on harkittu kokonaistavoitetta tukevaksi sekä samalla on karsittu turhat työvaiheet pois. (Beyer & Holtzblatt 1998)

Käyttöympäristömallissa käytetään fokusalueita kuvaamaan työn keskeistä sisältöä. Fokusalueet kuvataan mallissa laatikoina. Fokusalueelle kerätään toimintoja ja työhön liittyviä objekteja siten, että toisiinsa tai samaan tehtävään liittyvät yksiköt esitetään samalla fokusalueella, jolloin ne yhdessä muodostavat loogisen tehtäväkokonaisuuden. (Beyer & Holtzblatt 1998)

Fokusalueille määritellään tarkoitus, toiminnot, linkit, objektit, rajoitteet ja avoimet asiat (Kuva 10). Tarkoitus on lyhyt kuvaus siitä, mihin fokusalueen käytöllä pyritään. Toiminnot kuvataan lauseilla. Jokaisen toiminnon tulisi palvella kokonaistehtävän suorittamista. Toiminnot voivat olla joko suoraan käyttäjän suorittamia tai sitten automatisoituja, mutta kuitenkin sellaisia, joiden olemassaolon käyttäjät tiedostavat. Linkit ovat fokusalueiden välisiä tai keskinäisiä toimintoja. Linkit kuvataan fokusalueiden välisinä nuolina. Objektit ovat kohteita, joiden kautta käyttäjä tekee havaintoja fokusalueen tilasta tai joita manipuloimalla hän muuttaa tilaa. Rajoitukset ovat alueeseen liittyviä reunaehtoja. Rajoitteita voivat olla esimerkiksi nopeus ja luotettavuus. Avoimiin asioihin listataan fokusalueen toteutukseen liittyvät avoimet asiat. (Beyer & Holtzblatt 1998, s. 322)

Käyttöympäristömallin avulla pystytään hyvin erottelemaan suunniteltavan järjestelmän toimintoja toisistaan. Tämä on vaatimusten määrittelyn kannalta tärkeä toimenpide. Malli on myös selkeä ja ilmaisuvoimainen, jos kehitteillä oleva järjestelmä on suhteellisen pienikokoinen. Suuremmissa järjestelmissä fokusalueita voi olla valtava määrä. Tällöin ne joudutaan luultavasti järjestämään jontekin hierarkkisesti niin, että on olemassa eri tasoisia fokusalueita.

Toisaalta käyttöympäristömallin rakentaminen on jo hyvin lähellä yksityiskohtaisen suunnittelun tapaista toimintaa. Vaikka malli järjestelmästä on kuitenkin täysin käyttäjäkeskeinen. Siinä ei oteta kantaa siihen, miten kuvatut toiminnot teknisesti suoritetaan. Käyttöympäristömallia tarkastellessa käy kuitenkin selväksi, miten vaikeaa on vetää raja vaatimusten määrittelyn ja järjestelmäsuunnittelun välille. Samaa mallinnustekniikkaa voidaan usein käyttää molemmissa toiminnoissa. Mallin sisältö kuitenkin viime kädessä ratkaisee, kummasta on kyse.



Kuva 10. Esimerkki käyttöympäristömallista, jossa on kolme fokusaluetta (Beyer & Holtzblatt 1998, s. 320).

4.2.2.2 Unified Modelling Language kuvaukset

Unified Modelling Language (UML) on tietojärjestelmien olosuuntautuneeseen mallinnukseen tarkoitettu standardi notaatio. UML on Jim Rumbaughin, Grady Boochin ja Ivar Jacobsonin Rational Corporationin palveluksessa yhdessä kehittämä mallinnuskieli. Kielen ensimmäinen versio julkaistiin 1997. Tällä hetkellä UML on kansainvälinen standardi ja sen kehittämisestä vastaa Object Management Group (OMG), joka on maailmanlaajuinen olioparadigmaa käyttävistä yrityksistä koostuva organisaatio. (Schach 2002, s. 367)

UML käsittää yhteensä kaksitoista erilaista kaaviotyyppiä, joiden yhteiskäytöllä pyritään suunniteltavan järjestelmän mahdollisimman kokonaisvaltaiseen kuvaukseen. Kaaviotyypit voidaan luokitella niiden kuvaamien asioiden perusteella kolmeen alalajiin: rakenne-, käyttäytymis- ja mallinhallintakaavioihin. Vaatimusten määrittelyvaiheessa tärkein mallinnustapa on käyttäytymiskaavioista luultavasti tunnetuin, eli käyttötapauskaavio (engl. use case diagram).

Käyttötapauskaavioita (Kuva 11, Kuva 12) pidetään yleisesti hyvänä ratkaisuna erityisesti käyttäjävaatimusten kartoittamiseen. Käyttötapauskaavioita on melko yksinkertainen. Järjestelmän toiminnallisuus pyritään kuvaamaan joukkona järjestelmän käyttäjien sillä suorittamia tapahtumaketjuja. Käyttötapauskaavio kuvataan tyypillisesti tekstuaalisesti, ja järjestelmän eri käyttötapausten suhteet kuvataan käyttötapauskaaviolla. Jokaiseen käyttötapauskaavioon liittyy yksi tai useampi käyttäjärooli, joka toimii tapahtumaketjun suorittajana. (Haikala & Märijärvi 2000)

Käyttötapauskaavioita tekstuaalista kuvausta (Kuva 11) kutsutaan skenaarioksi. Käyttötapauskaavioilla on aina lähtötilanne ja lopputilanne. Näiden välillä on tapauksesta riippuva määrä käyttäjän tai järjestelmän toimintoja. Toimintasarja kuvataan mahdollisimman tarkasti ja siihen sisällytetään myös tyypillisimmät poikkeustilanteet. (Schach 2002)

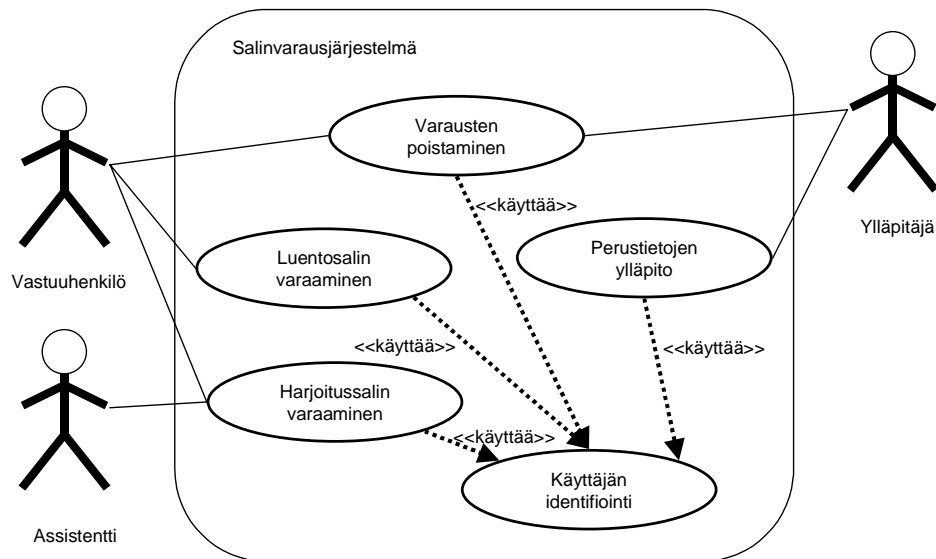
Käyttötapauskaavio (Kuva 12) koostuu monesta eri käyttötapauskaaviosta. Käyttötapauskaaviossa toimija (engl. actor) mallinnetaan tikku-ukkona. Yksittäiset käyttötapauskaavioita mallinnetaan soikioina, joilla kaikilla on oma nimensä. Järjestelmä kuvataan käyttötapausten taustalla olevana suorakulmiona. Käyttäjien ja käyttötapausten välillä on linkkejä (engl. association), joilla voi tarvittaessa olla myös

suunta. Käyttötapausten välistä suhdetta voidaan mallintaa myös yhteyksillä, joiden tyyppejä ovat esimerkiksi käyttö ja laajennus.

Käyttötapaukset pyritään aina kirjoittamaan käyttäjän näkökulmasta, jolloin järjestelmää kohdellaan ”black-box”-tyyppisesti, eli sen toimintaan liittyviä rakenteita ei eritellä tarkasti. Tämä mahdollistaa sen, että käyttötapauksilla pystytään kuvaamaan jonkin käyttäjän tarve sekä tilanne, jossa tarve on tyydytetty.

Nimi:	Luentosalin varaaminen, versio 1.0/ijh
Suorittajat:	Kurssin vastuuhenkilö
Alkuehdot:	Vastuuhenkilö ja kurssi on syötetty järjestelmään (KT henkilötietojen ylläpito)
Kuvaus:	Vastuuhenkilö seuraa WWW-linkkiä, joka johtaa järjestelmän pääsivulle. Hän syöttää järjestelmään käyttäjätunnuksensa ja salasanaansa (uses: KT käyttäjän identifiointi). Käyttäjä pyytää järjestelmää näyttämään salin varaustilanteen haluamaltaan aikaväliltä. Hän saa eteensä salin lukujärjestysnäytön (ks. Liite). Käyttäjä näkee näytöstä vapaat ajat sekä myös, mille kurseille sali on milloinkin varattu ja kuinka monelle viikolle. Käyttäjä tekee varauksen joltain vapaaksi havaitsemaltaan ajankohdalta. [Poikkeus: varaus ei onnistu].
Poikkeukset:	Varaus ei onnistu: Varaustilanne on voinut muuttua sillä aikaa, kun varaaja tekee varausta. Järjestelmä ilmoittaa tilanteesta käyttäjälle ja käyttäjä yrittää uudelleen.
Loppuehdot:	Varaukset kurssin luentoajoiksi on tehty.
Muut vaatimukset:	Päivittäin käsitellään kiireisimpänäkin aikana enintään n. 100 varausta. Vastausajan on oltava alle 1 sekuntia, lukujärjestysnäytön päivitys saa kestää 5 sekuntia.

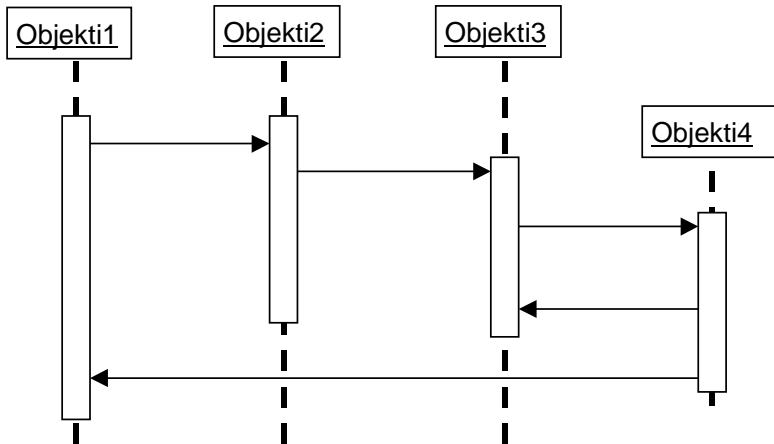
Kuva 11. Esimerkki käyttötapausskenaariosta (Haikala & Märijärvi 2000, s. 144).



Kuva 12. Käyttötapauskaavio luentosalinvarausjärjestelmästä (Haikala & Märijärvi 2000).

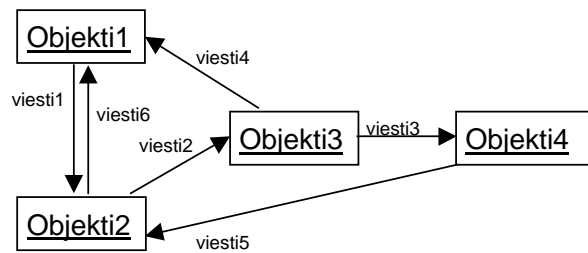
UML:n käyttäytymiskaavioista sekvenssi- ja yhteistyökaavioiden avulla voidaan kuvata objektien tai objektiryhmien välistä vuorovaikutusta. Tyypillisesti molemmilla interaktiokaavioilla kuvataan yhteen käyttötapaukseen sisältyvä vuorovaikutus. Kaaviossa kuvataan esimerkkitapaukseen liittyvät objektit sekä niiden välillä kulkevat viestit. (Fowler 1997)

Sekvenssikaaviossa (Kuva 13) aika kulkee ylhäältä alaspäin. Kaavio kuvaa eri toimintojen tapahtumisjärjestyksen. Toiminnot esitetään objektien välisinä viesteinä, jotka kuvataan kaaviossa nuolina. Objektin elinaika kuvataan ns. aktiivisuuspalkkina aikajanan päällä. Objektien väliset viestit voivat sisältää ehtoja ja parametreja.



Kuva 13. Geneerinen sekvenssikaavio.

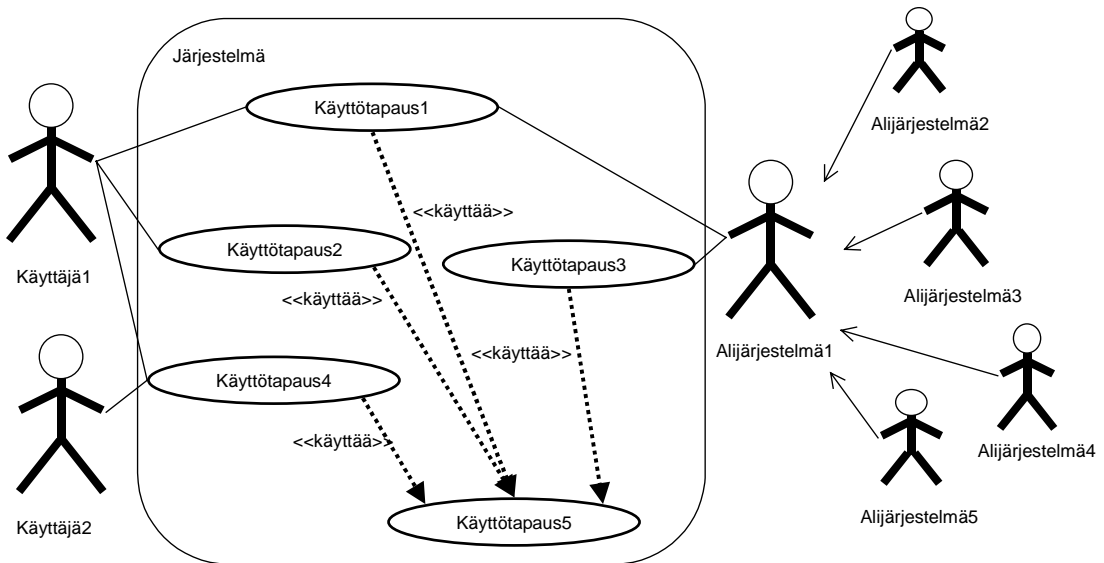
Myös yhteistyökaavio (Kuva 14) kuvaa objektien välisiä viestejä. Sen avulla voidaan ilmaista samat asiat kuin sekvenssikaaviollakin. Yhteistyökaavio on kuitenkin pätevämpi tilanteessa, jossa objektien välillä tapahtuvat toiminnot eivät tapahdu täysin sekvenssimäisesti.



Kuva 14. Geneerinen yhteistyökaavio.

UML:n kehityksestä vastaava organisaatio OMG on vuonna 2002 julkaissut spesifikaation (OMG 2002) siitä, miten UML:ää pitäisi käyttää järjestelmien reaaliaikaominaisuuksien mallintamiseen. Ohjeistus koskee reaaliaikaisten järjestelmien resurssien, aikaan liittyvien tekijöiden, rinnakkaisuuden, aikataulutuksen ja suorituskyvyn mallintamista. Spesifikaation mukaan siinä esitettyjä paradigmoja käytetään tehtäessä kvantitatiivisia ennustuksia järjestelmien käyttäytymisessä. Toisaalta samalla suunnitteluaikeiden kommunikointi järjestelmän kehittäjien välillä sekä erilaisten analyysi- ja suunnittelumenetelmien yhteentömmivuus ovat paradigmojen käytön hyötyjen listalla. (OMG 2002)

Reaaliaikaisten ja laajojen järjestelmien vaatimusten mallintaminen UML:n avulla tapahtuu Douglasin (2001) mukaan jakamalla järjestelmän toiminta eri tasoille. Tämän jälkeen koko toimintaa mallinnetaan top-down-menetelmällä jossa alemman tason järjestelmän suhde ylemmän tason järjestelmään on alijärjestelmä. Tällöin käyttötapauskaavion toimija (actor) mallinnetaan alijärjestelmäksi (Kuva 15), jolla on suhde sekä käsiteltävään järjestelmään, että mahdollisesti omiin alijärjestelmiinsä. Tässä mallinnustavassa on otettava huomioon, että järjestelmälle kertyy enemmän toimijoita kuin yhden hierarkiatason järjestelmän kuvauksessa olisi. (Douglas 2001)



Kuva 15. Käyttötapauskaavio, joka kuvaa järjestelmää, jolla on alijärjestelmiä (mukaillen Douglas 2001).

4.2.2.3 Kognitiivisen työn analyysin mallinnustekniikat

Kognitiivisen työn analyysin teoreettinen tausta ja prosessimalli on esitelty tämän työn luvussa 2. Analyysin kehittäjä Kim Vicente (1999) esittää, että monimutkaisten sosioteknisten järjestelmien suunnittelussa pitäisi harjoittaa ekologista suunnittelua, johon kuuluu formatiivinen mallinnus. Ekologisen suunnittelun pääajatus on se, että suunnittelun lähtökohtana ovat työkohteen ihmisen toiminnalle aiheuttamat reunaehdot. Työkohteella Vicente tarkoittaa esimerkiksi ohjausjärjestelmän avulla hallittavaa prosessia.

Yksi tärkeimmistä Vicenten (1999) huomioista on, että tehtävälähtöisellä lähestymistavalla on erittäin vaikea mallintaa monimutkaista järjestelmää tarvittavalla yksityiskohtaisuuden tasolla. Järjestelmien suunnittelu tehtävänälyysin avulla johtaa järjestelmiin, jotka tukevat nimenomaan suunnittelussa käytettyjen tehtävien suorittamista. Vicente argumentoi tämän olevan riittämätöntä, sillä suunnitteluvaiheessa ei monimutkaisille järjestelmille pystytä tekemään tarpeeksi kattavaa tehtävänälyysia. Tehtävänälyysin tulokset kertovat aina rutiinitehtävistä, kun erityisen merkityksellistä olisi löytää erilaisissa poikkeustilanteissa syntyviä tehtäviä. Vicente kertoo onnettomuuksien juontavan juurensa usein tilanteisiin, joissa käyttäjä on jonkin odottamattoman tapahtuman takia joutunut poikkeamaan rutiinitehtävästään, mutta järjestelmän tuki käyttäjän päätöksenteolle on ollut riittämätöntä. Tämän vuoksi Vicente kehottaa turvallisuuskriittisiä järjestelmiä suunniteltaessa käyttämään hyväksi suunnittelua, joka ei perustu perinteiselle sekventiaaliselle tehtävänälyysille.

Vicenten (1999) mukaan perinteisen tehtävänälyysin avulla suunnittelu johtaa proseduraalisiin käyttäjäliityntöihin. Hän tarkoittaa tällä sitä, että käyttöliittymä pakottaa käyttäjän toimimaan ennalta määritellyn toimintosarjan mukaan. Tällöin työssä oppiminen tarkoittaa lähinnä näiden toimintasarjojen oppimista ja erilaiset järjestelmään sisältyvät syy- ja seuraussuhteet jäävät käyttäjältä havainnoimatta. Tämä ei tue todenmukaisen ja oikean järjestelmäkuvan syntymistä, mikä taas on ongelma- ja poikkeustilanteissa alkuoletus tilanteen menestykselle ratkaisemiselle ja esimerkiksi onnettomuuksien välttämiseksi. Suurimmat uhkat järjestelmien turvallisuudelle aiheuttavat tilanteet, jotka ovat käyttäjille tuntemattomia, ja joita suunnittelijat eivät järjestelmiä laatiessaan ole osanneet ennakoita.

Abstraction-decomposition space eli ADS-malli (Kuva 16) on Kognitiivisen työn analyysin ensimmäisen vaiheen mallinnustekniikka. Vaihe on Kognitiivisen työn analyysin (CWA) tärkein, ja samalla se vaihe, joka eniten poikkeaa perinteisestä tehtävälähtöisestä suunnittelusta. (Vicente 1999)

ADS on alun perin Jens Rasmussenin kehittämä kuvaustapa valvonta- ja ope-
rintikäyttöliittymien mallintamiseen (Vicente 1999, Lind 1999). ADS on kaksi-
ulotteinen taulukko, jossa vaaka-akselilla kuvataan mallinnettavan järjestelmän
hajoitelma koko järjestelmästä osajärjestelmiin ja komponentteihin asti. Pysty-
akselilla kuvataan abstraktiohierarkiaa (AH) eli järjestelmän tavoitteiden eri

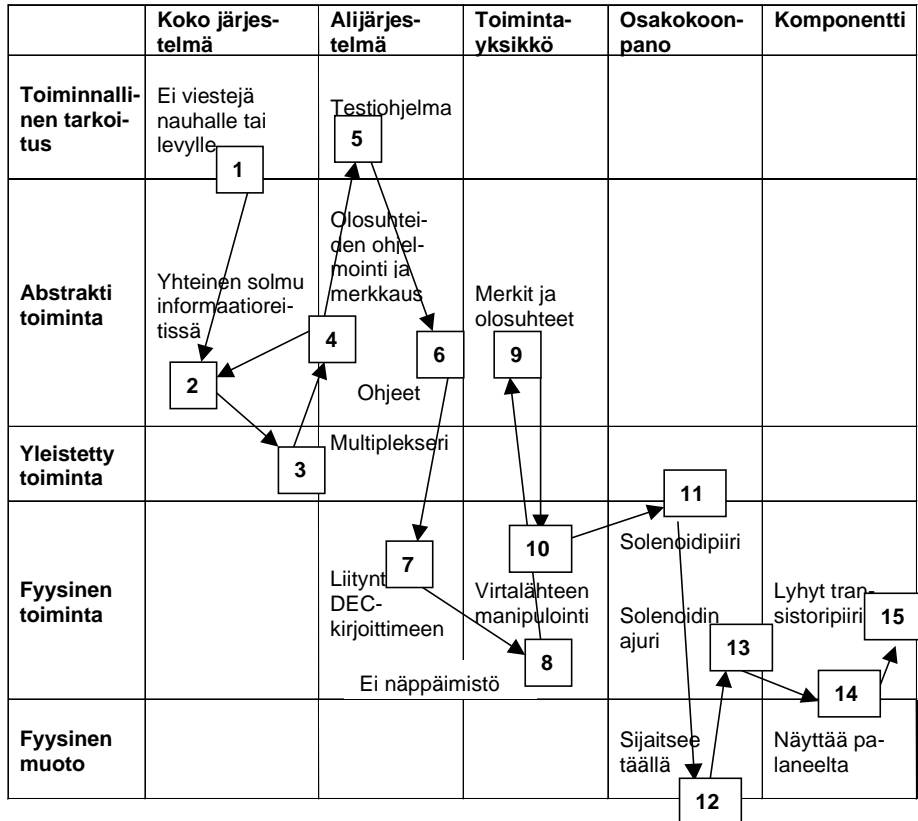
abstraktiotasoja korkeimmalta alkaen. AH:n korkeammat tasot kuvaavat järjestelmää sen toiminnallisten tavoitteiden kautta, kun taas alemmat tasot kuvaavat sen fyysisiä ominaisuuksia. AH siis pyrkii kuvaamaan kuilua järjestelmän taroituksen ja fyysisten ominaisuuksien välillä. (Vicente 1999, s. 158)

	Koko keho	Systeemi	Elin	Kudos	Solu
Tavoitteet	Toiminnan säilyttäminen entisen kaltaisena	Sopiva verenkierto, hengitys	Sopiva elinten nesteen ja veren saanti	Sopivan kudosten hapensaanti ja läpivirtaus	Sopivan solujen hapensaanti ja läpivirtaus
Tasapainot	Massan ja energian saanti, varastointi ja ulosvirtaus	Massan ja energian saanti, varastointi ja ulosvirtaus ja siirto	Massan ja energian saanti, varastointi ja ulosvirtaus ja siirto	Massan ja energian saanti, varastointi ja ulosvirtaus ja siirto	Massan ja energian saanti, varastointi ja ulosvirtaus ja siirto
Prosessit	Nesteiden kokonaisuus, ruumiinlämpö, hapen, nesteiden ja ravintoaineiden saanti	Verenkierto, hapensaanti, hengittäminen, kiertomäärät	Virtausten paineet, elinten verimäärä, verisuonten vastapaine	Kudoksen hapensaanti, hikoilu, metabolismi	Solun metabolismi, kemialliset reaktiot
Fysiologia		Systeemi-toiminnot	Elinten toiminnot	Kudoksen toiminnot	Solujen toiminnot
Anatomia			Elinten anatomia	Kudoksen anatomia	Solujen anatomia

Kuva 16. ADS-esimerkki ihmisen fysiologiasta (Hajdukiewicz ym. 1998).

Järjestelmien mallintaminen ADS:n avulla mahdollistaa käyttäjän päätöksenteon liittyvän ajatusketjun mallintamisen. Tämän ketjun ymmärtäminen on oleellisen tärkeää, kun suunnittelija päättää minkälaista informaatiota järjestelmän tulee missäkin tilanteessa tarjota käyttäjälle. (Vicente 1999)

Vicente (1999) esittää päätöksenteon mallintamisen esimerkin avulla (Kuva). Esimerkissä asiantuntija etsii vikaa tietokonelaitteesta. Ongelmanratkaisuketju lähtee liikkeelle ongelmatilanteesta, eli toiminnassa on havaittu häiriö. Jokainen ketjun solmukohta kuvaa asiantuntijan verbaalista analyysia tilanteesta.



Kuva 17. Esimerkki ongelmanratkaisuun liittyvän ajatusketjun mallintamisesta ADS:n avulla (Vicente 1999, s. 157).

ADS:ää on käytetty järjestelmien mallintamiseen ydinvoimateollisuudessa Yhdysvalloissa. Projekteissa on ollut tavoitteena järjestelmien luotettavuuden kasvattaminen valvomouudistusten yhteydessä. Sekä teollisuuden edustajat että akateemiset tutkijat tuntevat olevan yhtä mieltä siitä, että ADS:n avulla mallintaminen on järkevää ja hyödyllistä. Toisaalta on myös havaittu, että itse mallin laatiminen on suhteellisen vaikeaa. Vaikeudet ovat sekä käsitteellisiä, että metodologisia. Varsinkin pystyakselilla kuvattavan abstraktiohierarkian tasot ovat

vaikeasti määriteltäviä ja käytännössä eri ihmisten tekemät mallit samasta järjestelmästä muodostuvat toisistaan poikkeaviksi. (Lind 1999)

Lind (1999) toteaa osan ADS:n käytön hankaluuksista johtuvan siitä, että ihminen ongelmanratkaisutilanteissa joutuu tekemään eri tyyppisiä arvioita tilanteen selvittämiseksi. Tämän pitäisi heijastua myös ADS:n avulla mallinnettavassa päätöksentekoketjussa. Lind (1999) jatkaa eri tyyppisiä tilanteita olevan kolme. Ensimmäinen on järjestelmän tilan tunnistaminen. Tässä vaiheessa käyttäjä miettii esimerkiksi mikä mahdollinen vikatilanne on kyseessä. Tilanne vaihtuu toiseksi, kun järjestelmän tila on tunnistettu. Tällöin käyttäjä tekee päätöksen tavoitteestaan. Kolmas erityyppinen tilanne seuraa tavoitteen määrittelyn jälkeen. Tällöin käyttäjä tekee toimintasuunnitelman eli päättää siitä, mitä hänen on tehtävä järjestelmän ollessa määritellyssä tilassa, jotta hän pääsisi määrittelemäänsä tavoitteeseen. ADS ei tue näihin eri tilanteisiin liittyviä erilaisia näkökulmia järjestelmästä.

CWA:n toinen vaihe analysoi työkohteen hallintaan liittyviä tehtäviä. Toisen vaiheen mallinnustyökalu on päätöstikapuut (engl. decision ladder) (kuva 18). Myös päätöstikapuut on alunperin Jens Rasmussenin kehittämä mallinnustyökalu (Vicente 1999, s. 187). Sen avulla voidaan mallintaa tehtävien panos-tuotos-suhteita, eli sitä, mikä on edellytys suunnitteilla olevan tehtävän suorittamiselle ja toisaalta sitä, mitä suunniteltu tehtävä tuottaa panokseksi seuraavalle tehtävälle.

Mallissa laatikot kuvaavat toimintasarjaan liittyviä tehtäviä ja ympyrät tehtäviin liittyviä panoksia ja tuotoksia. Hypyt tikapuiden osien yli kuvaavat käyttäjän oivalluksia, jotka ovat tyyppisiä kokeneille käyttäjille. (Vicente 1999)

Myös päätöstikapuut mallintaa käyttäjän päätöksentekoa tilanteessa, jossa vaaditaan ongelmanratkaisua. Se eroaa esimerkiksi CD:n sekvenssimallista siinä, että se pureutuu enemmän käyttäjän päätöksenteon prosessiin. Kun edellä mainitut mallit kuvaavat sitä, mitä käyttäjä tekee, kertoo päätöstikapuut enemmän siitä, miksi hän toimii niin kuin hän toimii. Näin ollen voisi ajatella, että päätöstikapuut mallintaisi vaatimuksia informaatioisällölle ja muut sekvenssimallit tarvittaville toiminnoille.

Esimerkkejä tällaisista notaatioista ovat Z, Modal Action Logic ja LOTOS. (Sutcliffe 2002, s. 120)

Formaalien metodien esittely esimerkkien avulla on jätetty tämän työn rajauksen ulkopuolelle. Tähän on suurimpana syynä se, että formaalit metodit ovat käytännössä paremminkin järjestelmän spesifioinnin työkalu. Vaatimukset esitetään tyypillisesti jollain epäformaalisia piirteitä sisältävällä kuvaustavalla. Näitä menetelmiä käytettäessä joudutaan tinkimään semantiikan täsmällisyydestä, mutta samalla voitetaan kuitenkin määrittelytyöhön käytetyssä ajassa, sillä formaalin menetelmän käyttö on työläämpää. Vaatimusten esittäminen formaalilla notaatiolla tekisi myös niiden validoinnin käyttäjän kanssa vaikeaksi. Formaali kieli on useimmille loppukäyttäjille erittäin vaikea ymmärtää.

4.3 Vaatimusten muodostaminen käyttäjäkeskeisenä prosessina

Käyttäjäkeskeisen suunnittelun vaatimusmäärittelyvaihe on prosessi, joka alkaa käyttäjätiedon hankkimisella. Tässä vaiheessa alustava idea suunnittelun kohteesta on usein jo valmiiksi mietitty. Käyttäjätiedon avulla tuoteideaa kehitetään, jolloin samalla luodaan tietoa vaatimusmäärittelyn sekä tuotteen tarkan spesifiikaation tueksi. Vaatimusten määrittely on siis prosessi, joka alkaa alustavasta tuoteideasta ja päättyy tuotteen määrittelydokumentoinnin luomiseen. Toisaalta vaatimusprosessi jatkuu koko tuotekehityksen elinkaaren ajan vaatimusten hallintana. Tällä tarkoitetaan sitä, että vaatimusmäärittelyssä esitettyjä vaatimuksia käytetään hyväksi tuotekehitysprosessin elinkaaren eri vaiheissa. Samoin vaatimuksissa ilmenneisiin muutostarpeisiin voidaan vastata tuotekehitysprosessin elinkaaren myöhemmissäkin vaiheissa.

Suunnittelutoiminnan jäljitettävyydellä tarkoitetaan sitä, että kaikki tuotekehityksessä tehdyt tuotteen ominaisuuksia koskevat päätökset voidaan jäljittää taaksepäin aina vaatimusmäärittelyn vaatimuksiin asti (Schach 2002, s. 37–38). Jäljitettävyys on ominaisuus, joka mahdollistaa laadunvalvonnan sekä esimerkiksi muutospyyntöjen hallinnan sovelluskehitysprosessin aikana. Laadunvalvonnan on mahdotonta todeta tuotteen toteuttavan määrittelynsä ilman kehitysprosessin jäljitettävyyttä. Jos prosessi on jäljitettävä, voidaan myös jokainen ohjelmakom-

ponentti yhdistää osaan määrittelydokumenttia, ja samoin määrittely-yksikkö osaan vaatimusdokumentointia. (Schach, s. 278–279)

Hyvän jäljitettävyyden periaate on otettava huomioon jo vaatimusten määrittely-vaiheessa. Vaatimusten perusteena ovat tyypillisesti tuotteen käyttäjän, ostajan tai ostavan organisaation tarpeet. Näiden tarpeiden linkittäminen vaatimuksiin parantaa jäljitettävyyttä. Vaatimusten validointi tapahtuu esimerkiksi tarkastamalla, että kaikki dokumentoidut tarpeet on otettu huomioon vaatimuksissa.

4.3.1 Käyttäjätarpeiden dokumentointi

Sari Kujala (2002) on väitöskirjatyössään tutkinut käyttäjäkeskeisten suunnittelumenetelmien hyötyjä tuotekehitysprosessin aikaisissa vaiheissa. Käyttäjän tarpeiden dokumentoinnin merkitys on yksi hänen työnsä keskeisiä löydöksiä. Kujala on kehittänyt menetelmän, jossa käyttäjätutkimuksen aikana havaitut käyttäjätarpeet kirjataan ylös taulukoihin (kuva 19). Näihin taulukoihin kirjataan myös käyttökonteksti, eli tilanne, jossa kyseinen tarve heräsi. Näin tarpeen taustalla vaikuttavat tekijät tulevat myös dokumentoiduiksi. Taulukon avulla voidaan rakentaa silta käyttäjän tarpeiden ja käyttäjävaatimusten välille, jolloin jäljitettävyyden periaate toteutuu.

Tehtäväsekvenssi:	Ongelmat ja mahdollisuudet:
Askel 1: Käyttäjä tekee hälytyksen, koska on juuttunut hissiin.	<ul style="list-style-type: none"> • Ongelma: Käyttäjä haluaa päästä ulos hissistä mahdollisimman nopeasti. • Ongelma: Kaikkien käyttäjien pitää pystyä suorittamaan hälytys (sokeat, ulkomaalaiset jne.) • Ongelma: Joskus käyttäjät tekevät turhia hälytyksiä vahingossa. • Ongelma: Käyttäjät saattavat olla paniikitilassa. • Ongelma: Käyttäjät tarvitsevat välittömän palautteen yhteyden muodostumisen onnistumisesta ja siitä, että apu on tulossa.
Askel 2: Vapaa palvelukeskuksen työntekijä vastaanottaa hälytyksen ja kysyy lisätietoja.	<ul style="list-style-type: none"> • Ongelma: Kaukovalvontajärjestelmän eri versiot ja tyypit. • Ongelma: Hissin käyttäjä on ainoa informaation lähde. • Ongelma: Palvelukeskuksen työntekijä ei huomaa hälytystä.
Askel 3: Palvelukeskuksen operaattori siirtää saamansa tiedot järjestelmään ja lähettää tiedon alueen huoltomiehelle.	<ul style="list-style-type: none"> • Ongelma: Työläs vaihe palvelukeskuksen operaattorille. • Ongelma: Samanaikaiset soitot on eroteltava toisistaan. • Ongelma: Huoltomies ei näe tietoja. • Ongelma: Riittämättömät tiedot paikalliselta järjestelmältä. • Mahdollisuus: Ohjeet siitä, miten järjestelmää tulisi operoida. • Mahdollisuus: Mahdollisuus avata puhelinlinja palvelukeskuksesta hissiin.
Askel 4: Palvelukeskuksen operaattori soittaa huoltomiehelle ja lukee ongelman kuvauksen.	<ul style="list-style-type: none"> • Ongelma: Lisätyö palvelukeskuksen operaattorille.

Kuva 19. Esimerkki taulukosta, johon käyttäjän tarpeet on kirjattu. Kyseessä ovat hissin hälytystilanteeseen liittyvät käyttäjätarpeet. (Kujala 2002, s. 41)

Tarvetaulukon perusteella järjestelmän suunnittelijat muodostavat käyttötapaukset. Tällöin liikutaan kohti formaalimpaa määrittelyä, sillä käyttäjän tarpeet on vielä ilmaistu epäformaalisti, mutta käyttötapaukset ovat jo puoliformaali menetelmä vaatimusten kuvaamiseksi. Menettelytavassa kiinnitetään erityistä hu-

miota siihen, että linkki tietyn käyttötapausten ja käyttäjän tarpeen välillä on tuotu julki ja dokumentoitu. (Kujala 2002)

Käyttäjän tarpeet ovat käyttäjälähtöisen suunnitteluprosessin lähtökohta ja kantava voima. Tarpeiden tunnistaminen ja analyysi ovat tärkeitä suunnitteluprosessin alkupään vaiheita. Tarpeiden analyysin tekeminen eksplisiittiseksi auttaa vaatimusten muodostamisen prosessia varmasti, sillä harppaus tarpeista vaatimukseen on usein melko suuri ja sen onnistuminen riippuu pitkälti suunnittelijan kyvystä hahmottaa erilaisten ratkaisujen vaikutuksia käyttäjän näkökulmasta. Mitä pienemmiksi suunnittelijan suorittamat käsitteelliset loikat voidaan kutsua, sitä pienemmäksi muodostuu todennäköisyys, että ne tehtäisiin väärään suuntaan.

4.3.2 Contextual Designin prosessi

Contextual Design (CD) on prosessi, jossa järjestelmäsuunnittelu pyritään tekemään nojautuen todelliseen tietoon käyttäjän toiminnasta ja toiminnan kontekstista. CD on kokonaisvaltainen käyttäjäkeskeisen suunnittelu- tai tuotekehitysprosessin metodologia.

CD:ssa suunnitteluprosessi on jaettu seitsemään eri osavaiheeseen. Ensimmäinen vaihe eli käyttäjätutkimus tehdään Contextual Inquiry -menetelmällä. Toisessa vaiheessa kerätty tieto mallinnetaan erilaisilla kaaviotekniikoilla. Kolmannessa vaiheessa eri havainnointikerroilla saatu tieto yhdistetään yhdeksi malliksi. Neljännessä vaiheessa suunnittelutiimi ryhtyy pohtimaan ratkaisuja käyttäjien työn rakenteen parantamiseen rationalisoimalla työprosessin eri osavaiheiden suhteita. Viidennen vaiheen tuloksena tehdään työympäristömalli, jossa neljännen vaiheen erilaiset visiot konkretisoidaan työn luonnollisen kulun mukaiseksi, mahdollisimman virtaviivaiseksi malliksi. Työympäristömalli toimii pohjana järjestelmän toteutukselle. CD:n kuudenteen vaiheeseen kuuluu prototyyppi, jossa suhteellisen nopeassa tahdissa luodaan erilaisista suunnitteluratkaisuista paperiprototyyppejä, joita iteroidaan ja parannetaan yhdessä käyttäjien kanssa. Contextual Designin viimeiseen eli seitsemänteen vaiheeseen kuuluu järjestelmän tai tuotteen toteutuksen aloittaminen. Huomattavaa on, että järjestelmän toteutukseen osallistuvat samat henkilöt kuin suunnitteluunkin. Näin voidaan varmistaa, että visio kokonaistavoitteesta säilyy ehjänä myös toteutusprojektin ajan. (Beyer & Holtzblatt 1998)

CD-prosessin määrittelyssä on pyritty löytämään ja tunnistamaan sellaisia suunnittelun vaiheita, joita hyvät suunnittelijat jo käyvät läpi implisiittisesti, ja tekemään nämä vaiheet eksplisiittisiksi, jolloin suunnittelun hyvä lopputulos ei ole enää kiinni sattumasta tai suunnittelijan mahtavasta käsitteellisestä oivalluksesta. Jokainen CD:n seitsemästä vaiheesta joudutaan hyvässä suunnittelussa käymään läpi (Beyer & Holtzblatt 1998). Kyse on vain siitä, tapahtuuko tämä epäformaalisti suunnittelijan päässä, vai formaalisti yhtenä dokumentoituna suunnittelun vaiheena. Kun vaihe tehdään eksplisiittiseksi, mahdollistetaan samalla suunnittelutiimin kommunikointi ja synergia asian ympärillä. Beyer ja Holtzblatt uskovat tämän usein johtavan jopa vaiheeseen käytetyn ajan lyhentymiseen (Beyer & Holtzblatt 1998) ja samalla siis suunnitteluprosessin tehokkuuden kasvamiseen.

Contextual Design -prosessin käyttö helpottaa suunnitteluprosessin aikana tapahtuvien muutosvaatimusten hallintaa. Kun jokainen prosessin eri vaihe on mallinnettu ja dokumentoitu, mahdollistuu muutosten syiden ja vaikutusten seuraaminen. Samalla toteutuu myös vaatimus siitä, että määritellyt vaatimukset järjestelmälle ovat jäljitettävissä taaksepäin aina käyttökontekstin malleihin asti. (Beyer & Holtzblatt 1998)

Kaikkia CD-prosessin aikana tehtyjä malleja voidaan pitää osana vaatimusmäärittelyprosessia, sillä ne kuvaavat suunniteltavaa järjestelmää eli kertovat millainen sen tulisi olla. Kaikki uutta järjestelmää koskevat mallit voidaan siis tulkita vaatimuksiksi jollain tasolla.

4.3.3 Ihmiskeskeinen turvallisuuslähtöinen prosessi

Käyttäjakeskeisen ja turvallisuuslähtöisen suunnittelun yhdistäminen on käytännössä vaikeaa. Tämä on ollut pyrkimyksenä, kun ihmiskeskeinen turvallisuuslähtöinen suunnitteluprosessi (Leveson ym. 2001) on kehitetty. Prosessi on kehitetty isossa yhteistyöprojektissa, jonka tavoitteena on ollut Euroopan lennonvalvontajärjestelmän osittainen uusiminen.

Prosessimalli (kuva 20) esitetään kolmen rinnakkaisen aliprosessin kokonaisuutena. Keskellä kulkee yleinen järjestelmän suunnitteluprosessi ja siihen liittyvät eri aktiviteetit. Oikea sarake esittää turvallisuuslähtöiseen ja vasen sarake käyttäjakeskeiseen suunnitteluun liittyvät toiminnot. Malli on hyvin teoreettinen,

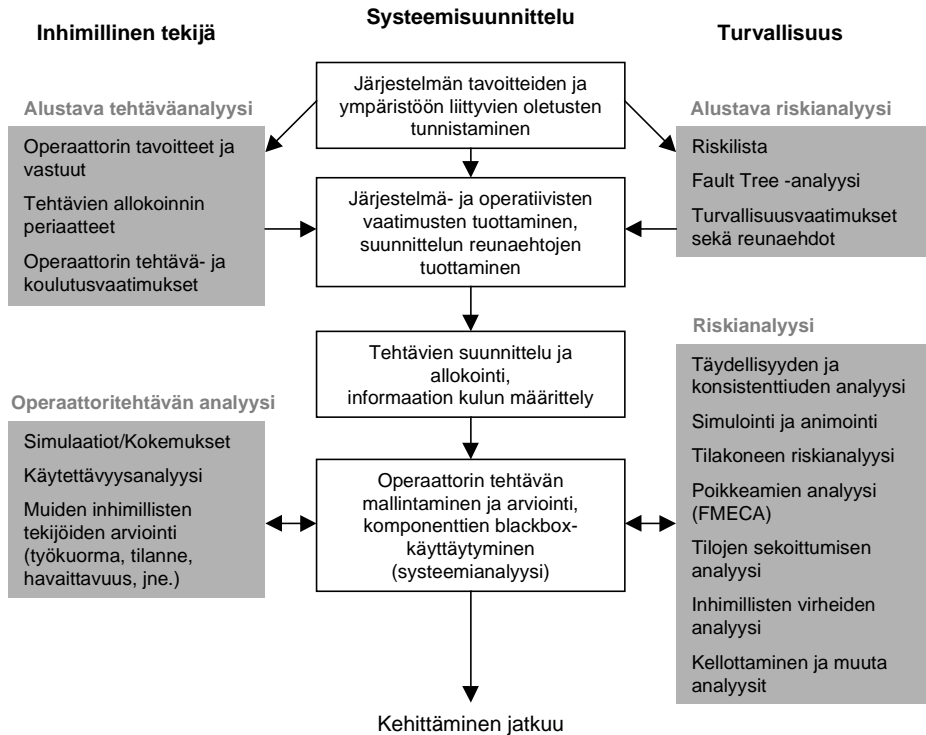
sillä generisen systeemisuunnitteluprosessin ja turvallisuus- ja käyttäjäprosessien integrointi tapahtuu läpi suunnittelutoiminnan ja järjestelmän käytön. Suunnittelutoimintaan liittyvää iterointia ja palautetta ei ole mallinnettu prosessin kuvaukseen. Samoin osa turvallisuusprosessiin kuuluvista aktiviteeteista (esimerkiksi inhimillisen virheen analyysi) on myös osa käyttäjäkeskeistä prosessia, joten ne voitaisiin yhtä hyvin esittää myös vasemmassa sarakkeessa. (Leveson ym. 2001)

Ihmiskeskeinen turvallisuuslähtöinen suunnitteluprosessi alkaa järjestelmän korkean tason toiminnallisten päämäärien määrittelyllä sekä järjestelmän toimintaympäristöön liittyvien reunaehtojen ja rajoitusten tunnistamisella. Alustava riskianalyysi tarkoittaa olemassa olevien ja tunnistettujen riskien dokumentoimista. Samalla järjestelmän käyttäytyminen, joka saattaa johtaa näihin riskitilanteisiin, dokumentoidaan. Alustavan riskianalyysin painopiste on uusiin toimintoihin liittyvien ”vanhojen” riskien tunnistamisessa. Alustavalla tehtävänälyysillä tarkoitetaan eri alojen asiantuntijoiden sekä käyttäjien yhdessä suorittamaa käyttäjien tehtävien ja vastuiden määrittelyä. Sama ryhmä määrittelee myös tehtävien allokoinnissa käytettävät periaatteet sekä käyttäjien koulutusvaatimukset. (Leveson ym. 2001)

Kolmen edellä esitellyn vaiheen tuotosten perusteella tuotetaan vaatimukset käyttäjän tehtäville, automatisoitaville toiminnoille ja käyttöliittymälle. Täydelliseen järjestelmän vaatimusmäärittelyyn kuuluvat toiminnallisuutta, ylläpitoa, hallintoa ja käyttöliittymää koskevat vaatimukset. Tämän lisäksi tuotetaan myös käyttövaatimukset ja suunnittelua rajoittavat reunaehdot. (Leveson ym. 2001)

Vaatimusten määrittelyn jälkeen suunnitteluprosessi jatkuu spesifikaation kirjoittamisella. Tämä tehdään formaalilla kielellä, joka voidaan testata virheiden ja epäloogisuuksien suhteen. Turvallisuusnäkökulmasta katsottuna määrittelyn formaalitus on tärkeää juuri sen takia, että koneellinen tarkistus on mahdollista ainoastaan formaalille notaatiolle. (Leveson ym. 2001)

Koko suunnitteluprosessi on iteratiivinen ja vaiheet osittain päällekkäisiä. Alussa, kun järjestelmä on vielä konseptitasolla, sekä riskianalyysi että tehtävänälyysi ovat vielä hyvin yleisiä. Kun siirrytään määrittelytasolle ja samalla formaaleihin mallinustekniikoihin siirrytään samalla sisällöltään yksityiskohtaisempiin malleihin. (Leveson ym. 2001)



Kuva 20. Ihmiskeskeisen turvallisuuslähtöisen suunnitteluprosessin alkuvaiheet (Leveson ym. 2001).

4.3.4 Vaatimusten määrittely ISO 13407:ssä

Käyttäjakeskeisessä tuotekehitysprosessissa perinteinen toiminnallinen vaatimusmäärittelyprosessi pitää laajentaa käsittämään myös käyttäjä- sekä organisaattoriset vaatimukset. Nämä on dokumentoitava eksplisiittisesti (taulukko 4), jolloin niiden suhde käyttökontekstin määrittelyyn on selkeä (ISO 1999). Toisin sanoen vaatimusten perusteet ovat käyttökontekstissa, eli jokainen vaatimus tulisi olla jäljitettävissä taaksepäin johonkin kontekstiin liittyvään ominaisuuteen.

Näin standardi siis toteaa, mitä erityyppisiä vaatimuksia käyttäjakeskeisessä vaatimusten määrittelyssä on otettava huomioon. Siihen, miten nämä vaatimukset tulisi määrittelydokumentissa ilmaista, standardi ei ota kantaa. Standardi ei ota kantaa vaatimusten esittämiseen sen enempää kuin että ne pitää olla doku-

mentoitu hyvin. Spesifikaation tulee kuitenkin täyttää seuraavat kriteerit (ISO 1999):

- Oleelliset käyttäjäryhmät on tunnistettava.
- Käytettävyystavoitteet on ilmaistava selkeästi.
- Eri vaatimusten prioriteetti on määriteltävä.
- Metriikat vaatimusten täyttymiseen on määriteltävä siten, että niitä voidaan testausvaiheessa käyttää hyväksi.
- Vaatimukset on todennettava käyttäjillä tai heidän edustajillaan.
- Vaatimusten tulee sisältää lakien ja asetusten asettamat reunaehdot.
- Vaatimusten tulee olla riittävällä tavalla dokumentoituja.

Standardi toteaa, että vaatimusten määrittelyssä on perusteet järjestelmän myöhempien toteutusvaiheiden tulosten testaustulle. Standardin mukaisen käyttäjäkeskeisen tuotekehityksen suunnitteluratkaisujen evaluointivaihe perustuu siten myös vaatimusmäärittelyvaiheen tuotoksiin. Jotta evaluointi voitaisiin tehdä, on vaatimusmäärittelyvaiheessa määriteltävä metriikat, joiden avulla vaatimusten voidaan sanoa täyttyvän. (ISO 1999)

Evaluointia tehdään koko tuotekehitysprosessin ajan. Mitä aikaisemmasta prosessin vaiheesta on kyse, sitä enemmän evaluoinnin tarkoituksena on ohjata suunnittelua. Kun suunnittelu on jo edennyt niin pitkälle, että evaluointia voidaan tehdä valmiiden prototyyppien avulla, voidaan testata myös vaatimusten toteutumista. (ISO 1999)

Evaluointitekniikat vaihtelevat niiden formaaliosasteen, tarvittavan käyttäjien osallistumisen ja evaluointiympäristön suhteen. Usein jokin tietty tekniikka valitaan evaluointiin ajan ja rahallisten syiden takia, mutta toisaalta myös suunniteltavan järjestelmän luonne ja valmiusaste vaikuttavat aina tekniikan valintaan. (ISO 1999)

Evaluointi toimii aina myös uusien vaatimusten esiin nostamisen keinona. Esimerkiksi käytettävyydestä voidaan huomata, että jokin tietty vaatimus täyttyy sille määritellyn metriikan mukaan, mutta tuotteen käytettävyydessä on silti ongelmia. Tässä tapauksessa järjestelmälle on syytä määritellä lisää vaatimuksia.

Vaatimusmäärittely on dokumenttina elävä, eli vaatimuksia voidaan kehitysprosessin aikana sekä lisätä että poistaa.

Taulukko 4. Standardin mukaiset käyttäjä- sekä organisatorisista vaatimuksista dokumentoitavat näkökulmat (ISO 1999).

Näkökulma	
Uuden järjestelmän tavoiteltu suorituskyky	<ul style="list-style-type: none"> • Taloudellinen • Operatiivinen
Käyttöliittymän ja työaseman suunnittelu	<ul style="list-style-type: none"> • Layout • Ergonomia
Relevantit lakisääteiset vaatimukset	<ul style="list-style-type: none"> • Turvallisuusmääräykset • Terveysmääräykset • Standardit
Käyttäjien ja muiden merkityksellisten osapuolten vuorovaikutus	<ul style="list-style-type: none"> • Yhteistyö ja sen muodot • Kommunikaatio
Käyttäjien työ	<ul style="list-style-type: none"> • Tehtävien allokointi • Käyttäjien hyvinvointi • Motivaatio
Tehtävien suoritus	<ul style="list-style-type: none"> • Suoritustapa
Työn suunnittelu ja organisaatio	
Muutoksen hallinta	<ul style="list-style-type: none"> • Koulutus • Osallistuva henkilökunta
Käytön ja kunnossapidon toteutettavuus	

4.4 Vaatimusten kuvausmenetelmän valinta

Tämän luvun alussa esitellyistä kuvaustekniikoiden ominaisuuksista tärkein on mielestäni ilmaisuvoima, jolla tarkoitetaan esimerkiksi mallinnustekniikan kykyä ilmaista suunniteltavan järjestelmän ominaisuuksia tekemättä laajoja yksinkertaistuksia. Tämä pätee erityisesti monimutkaisiin sosioteknisiin järjestelmiin, sillä niitä on voitava mallintaa erittäin monipuolisesti, jotta kokonaiskuva järjestelmän toiminnasta ja käyttötilanteesta välittyisi kohdeyleisölle. Tähän tarvitaan ilmaisuvoimaa. Kaikki muutkin ominaisuudet ovat kuitenkin tärkeitä, joten

selkeää valintaa esimerkiksi kuvaustavan tekemisen helppouden ja kuvauksen tarkkuuden suhteen ei monimutkaisten järjestelmien suhteen voida tehdä.

Tässä työssä on esitetty käyttäjäsuuntautuneen suunnittelun järjestelmien kuvaustekniikoita, jolloin kuvaustekniikan tärkeiksi ominaisuuksiksi nousevat myös lopputuloksen ymmärrettävyys sekä mallin muunneltavuuden helppous. Mallin ymmärrettävyys mahdollistaa sen, että ehdotetuista ratkaisuista voidaan keskustella, ja niiden toimintaa voidaan simuloida. Muunneltavuuden helppous taas takaa sen, että käytetty mallinnustekniikka on niin kevyt, että käyttäjän reaktiot malleista voidaan ja halutaan ottaa huomioon, eikä niiden perusteella tehtäviä muutoksia koeta liian raskaiksi toteuttaa. Monimutkaisista sosioteknisistä järjestelmistä on tarpeen kuvata näkökulmia, jotka liittyvät monimutkaisuuteen, järjestelmien sosiaaliseen käyttöön sekä toteutuksen teknisiin ratkaisuihin. Valittavan kuvausmenetelmän on siis tuettava ainakin erilaisten käyttäjäröolien, järjestelmän dynaamisen käyttäytymisen, järjestelmän laajuuden ja mahdollisten riskien mallintamista. Vaatimuksia tulisi myös pystyä mallintamaan nimenomaan käyttäjän näkökulmasta, sillä tämä on käyttäjakeskeisen suunnittelun lähtökohta. Lisäksi vaatimusten kuvausmenetelmän tulee olla mahdollisimman ilmaisuvoimainen, mutta samalla tarkka ja antaa tarvittavan yksityiskohtaista tietoa järjestelmän toiminnasta.

Ohessa on esitetty taulukoituna (taulukko 5) aiemmin läpi käytyjen vaatimusten kuvausmenetelmien ominaisuuksia. Taulukko perustuu kirjoittajan kirjallisuusselvityksen perusteella syntyneeseen näkemykseen kunkin kuvaustekniikan ominaisuuksista. Taulukkoon otettiin mukaan niitä ominaisuuksia, joita kirjoittajalla oli valmiuksia arvioida ja jotka on katsottu tärkeiksi monimutkaisten järjestelmien vaatimusten mallintamisessa. Formaalius kertoo mallinnuksen notaatioon liittyvien käytäntöjen yhtenäisyydestä. Eri menetelmien formaaliuden luokittelussa on käytetty Sutcliffen (2002) jaottelua. Käyttäjryhmien toiminta -kohdassa on arvioitu sitä, miten hyvin mallinnusmenetelmä tukee eri käyttäjryhmien toiminnan ja yhteistoiminnan esittämistä. Dynaamisen käyttäytymisen, järjestelmän laajuuden ja riskien mallintamisen mahdollisuutta on arvioitu asteikolla heikko – mahdollinen – melko hyvä – hyvä. Mallinnustapa-kohdassa pohditaan Vicenten (1999) esittämää teoriaa kolmesta eri mallinnussukupolvesta. Normatiivinen, deskriptiivinen ja formatiivinen mallinnus on esitelty tarkemmin tämän julkaisun luvussa 2. Ilmaisuvoima ja tarkkuus kuvaavat sitä, kuinka hyvin mallin avulla voidaan esittää järjestelmän monipuolisia ominaisuuksia tarkasti.

Kuvausmenetelmän työläydessä on arvioitu mallin muodostamisen vaikeutta ja mallin monimutkaisuutta. Samoin työläyteen vaikuttaa mallin muodostamiseen tarvittavan tiedon analysointiin kuluva aika.

UML:n käyttötapaukset ovat ilmaisuvoimaltaan vahva järjestelmän toiminnan kuvausmenetelmä. Käyttötapauksen avulla pystytään mallintamaan eri käyttäjäryhmien toimintaa tilannesidonnaisesti. Mallinnus voidaan tehdä joko normatiivisesti tai deskriptiivisesti. Mallinnus tehdään käyttäjän näkökulmasta. Tällöin voidaan tehdä malleja, joita voidaan simuloida käyttäjien kanssa, jolloin voidaan arvioida mallin hyvyttä käyttäjän kannalta. Vaikka mallinnus käyttötapauksen avulla on ilmaisuvoimaista, ovat mallit kuitenkin suhteellisen helppoja ymmärtää, joten tämäkin tukee käyttäjien kanssa yhdessä tehtävää mallien arviointia.

ADS on CWA-metodologian kulmakiviä. Sillä alkaa koko järjestelmän toiminnan mallinnusprosessi. ADS on myös puhtaasti formatiivinen ja ekologinen menetelmä työkohteen ominaisvaatimusten kuvaamiseen. Sen avulla kuvataan työkohteen suunniteltavalle järjestelmälle muodostamia ominaisvaatimuksia. ADS ei siis mallinna järjestelmää, vaan työkohdetta. Sen tarkoitus on kuitenkin selvittää järjestelmäsuunnittelijoille, mikä kohteesta saatava informaatio on järjestelmän käyttäjien kannalta merkityksellistä. Tähän perustuu ADS:n valinta monimutkaisen sosioteknisen järjestelmän vaatimusten mallintamiseen.

Taulukko 5. Yhteenveto kuvausmenetelmien ominaisuuksista. Lyhenteet: e-kirjain tarkoittaa epäformaalia, p-kirjain puoliformaalia, N-kirjain normatiivista, D-kirjain deskriptiivistä ja F-kirjain formatiivista.

Kuvausmenetelmä	Formaalius	Käyttäjär ryhmien toiminta	Dynaamisen käyttäytymisen kuvaaminen	Laajuuden kuvaaminen	Riskein mallintaminen	Mallinnustapa	Ilmaisuvoima	Tarkkuus	Työläys
Vaatusluettelo	e	mahdollinen	heikko	heikko	heikko	N	heikko	heikko	raskas
Skenaario	e	hyvä	hyvä	melko hyvä	hyvä	D	hyvä	heikko	kevyt
Storyboard	e	hyvä	melko hyvä	melko hyvä	melko hyvä	D	hyvä	heikko	kevyt
Prototyyppi	e	mahdollinen	hyvä	melko hyvä	melko hyvä	N	hyvä	hyvä	kevyt
CD: Vuorovaikutusmalli	p	hyvä	hyvä	hyvä	hyvä	N/D	hyvä	hyvä	melko raskas
CD: Sekvenssimalli	p	mahdollinen	hyvä	melko hyvä	hyvä	N/D	hyvä	mahdollinen	melko raskas
CD: Artefaktimalli	p	heikko	heikko	heikko	hyvä	D	hyvä	hyvä	kevyt
CD: Kulttuurimalli	p	hyvä	heikko	heikko	hyvä	D	melko hyvä	mahdollinen	raskas
CD: Fyysinen malli	p	heikko	heikko	melko hyvä	hyvä	D	melko hyvä	hyvä	kevyt
CD: Käyttöympäristöllinen malli	p	mahdollinen	hyvä	melko hyvä	hyvä	D/F	hyvä	hyvä	raskas
UML: Käyttötapaus	p	hyvä	hyvä	hyvä	hyvä	N/D	hyvä	hyvä	melko raskas
UML: Vuorovaikutusmallit	p	heikko	hyvä	melko hyvä	melko hyvä	N/D	hyvä	hyvä	melko raskas
CWA: ADS	p	heikko	hyvä	hyvä	hyvä	F	hyvä	mahdollinen	raskas
CWA: Päätöstikapuut	p	heikko	melko hyvä	heikko	hyvä	F	hyvä	mahdollinen	melko raskas

5. Tapaustutkimus: Alusten ilmoittautumisjärjestelmän vaatimusten määrittely

Tämä luku on tutkimuksen käytännön osa, jossa kahta kirjallisuusosuudessa esitellyistä vaatimusten kuvaustekniikoista on kokeiltu tapaustutkimuksessa. Kokeillut menetelmät valittiin kirjallisuustutkimuksen perusteella siten, että tämän työn johdannossa esitettyihin tutkimuskysymyksiin voitaisiin mahdollisimman hyvin pureutua. Menetelmien valintaa on esitelty tarkemmin tämän tutkimuksen kohdissa 3.3 ja 4.4.

Tapaustutkimus otettiin käytännön osuuden tutkimusstrategiaksi, sillä tarkoituksena oli tutkia, kuinka hyvin kirjallisuusselvityksen perusteella valitut menetelmät toimivat käytännön vaatimusten määrittelyssä. Tutkimus täyttää Hirsjärven ym. (1997, s. 123) tapaustutkimuksen määritelmän: Tutkimuksessa on käytetty useita metodeja ja tutkittavaa ilmiötä kuvaillaan laadullisin keinoin. Tutkimuksen kohde on yksittäinen suunnitteluprojekti, jota on tutkittu suhteessa ympäristöönsä.

Kohteena on Suomenlahdelle kesällä 2004 käyttöön otettavan alusten pakollisen ilmoittautumisjärjestelmän (engl. Ship Reporting System) (SRS) vaatimusten määrittely. Suomenlahdella alkava SRS-toiminta on kansainvälisesti ainutlaatuista, joten sille ei ole olemassa Kansainvälisen Merenkulkujärjestön (IMO) suosituksia vaan toiminnan määrittely on jätetty kolmen osallistujavaltion (Suomi, Viro, Venäjä) kolmikantayhteistyönä määriteltäväksi. Koska SRS-toiminta tulee osin pohjautumaan pidemmät perinteet omaavaan Vessel Traffic Services (VTS) -toimintaan, VTS:n periaatteet esitellään luvun alussa lyhyesti.

5.1 Taustaa: Vessel Traffic Services

Vessel Traffic Services eli VTS-järjestelmät toteuttavat rannalta käsin tapahtuvia toimintoja, joiden tarkoituksena on lisätä ympäristön ja merenkulun turvallisuutta. VTS tuottaa laivoille erilaisia palveluja, joiden kirjo vaihtelee sääennusteista muiden alusten paikkatietoihin ja tiukkaan satama-altaiden ja väylien liikenteen kontrollointiin. Suomessa VTS-toimintojen toteuttamisesta vastaa Me-

renkulkulaitos, joka käyttää VTS:stä suomenkielistä nimeä alusten ohjaus- ja tukipalvelu.

Tyypillisesti VTS toimii tarkkaan määritellyllä alueella. Kun alus saapuu VTS-alueelle, se ilmoittautuu maalla sijaitsevaan VTS-keskukseen. Ilmoittautuminen tehdään VHF-radion avulla paikalliselle VTS-operaattorille. Tämän jälkeen aluksen kulkua seurataan ja valvotaan VTS-keskuksesta käsin tutkatiedon avulla. Vastuu aluksen navigoinnista säilyy kuitenkin koko ajan aluksen kapteenilla, ja VTS-operaattori toimii neuvonantajan roolissa. Alukset pitävät koko ajan auki ennalta määritellyn VHF-kanavan, jonka kautta VTS-operaattorit kommunikoi-
vat suoraan komentosillalla työskentelevän laivan päällystön kanssa. Kaikki yhteydenpito keskuksen ja komentosillan välillä tapahtuu VHF-puhelinten avulla. VTS-keskuksessa on elektroniset kartat (merikortit) joihin alusten sijaintitiedot yhdistetään tietojärjestelmän avulla. Operaattorit tarkkailevat liikennettä mahdollisten riskitilanteiden kannalta ja antavat tarpeen mukaan varoituksia niistä, sekä kertovat minkä aluksen vuoro on edetä liikenneajoitetuilla väylillä. (Merenkulkulaitos 2002)

5.1.1 VTS:n tehtävät

VTS-järjestelmän päätavoite on edistää tehokasta, turvallista ja sujuvaa liikennettä satama-alueilla sekä niihin johtavilla väylillä. Wiersma & Heijer (1996) on tutkinut kansainvälistä VTS-toimintaa operaattoreiden tukijärjestelmien suunnittelua varten. Hän toteaa VTS-operaattoreiden ydintehtävien olevan (Wiersma & Heijer 1996):

- toimimista satamaviranomaisen ominaisuudessa
- systemaattista alusten liikkeiden seuranta
- meriliikennetiedon välittämistä; ohjeiden ja neuvojen välittämistä aluksille sekä kolmansille osapuolille
- yhteistyön koordinoimista ja neuvottelua alueellisen liikenteen hallinnan eri osapuolten kanssa
- operatiivisten toimintojen koordinoimista ja menettelytapojen muuntamista toiminnallisiksi ohjeiksi ja käytännön työjärjestykseksi.

Yleisesti voidaan sanoa, että VTS-operaattorit keräävät, prosessoivat ja arvioivat meriliikennetietoa ja tekevät sen perusteella päätelmiä. Oikean ja täsmällisen liikennekuvan muodostaminen on oleellinen osa heidän työtään. Havaintojensa perusteella VTS-operaattorit antavat toimintaohjeita aluksen navigoinnista vastaaville henkilöille.

5.1.2 VTS:n kehitys

Aluksen kapteeni on perinteisesti ollut se henkilö, joka määrää aluksen kurssin ja nopeuden. Tarpeen vaatiessa luotsi on vastannut samoista asioista yhteistyössä hänen kanssaan. Satamaa lähestyvät alukset ovat entisinä aikoina ilmoittaneet saapumisestaan ja etenemisestään lippumerkein. Radion keksiminen loi pohjan luotettavammalle yhteydelle laivojen ja sataman välille. Tutkajärjestelmien kehittyminen toisen maailmansodan aikana mahdollisti, laivojen navigoinnin turvallisuuden lisääntymisen ohella laivaliikenteen erittäin tarkan seuraamisen maista käsin. Rannikkoviranomaisten kyky seurata laivaliikennettä tutkan avulla yhdistettynä radion kautta lähetettäviin navigointia koskeviin viesteihin muodosti näin ensimmäiset muodolliset VTS-järjestelmät. (IMO 2002)

Nykyisin VTS-järjestelmiä on lähes kaikissa kansainvälisissä satamissa. Kaikkia merenkulkuun liittyviä standardeja säätelevä kansainvälinen merenkulkuorganisaatio (IMO) säätelee myös VTS:n kehitystä. Se on julkaissut standardin ja ohjeen VTS-järjestelmän käytöstä. Standardin taustalla oli tarve määritellä, missä tilanteissa VTS:n käyttöönotto on suotavaa, sekä lieventää pelkoa siitä, että VTS vaikuttaisi laivan kapteenin vastuuseen aluksensa navigoinnista. IMO:n mukaan VTS-järjestelmä on tarpeellinen satamien lähestymis- ja sisäänmenoväylillä sekä alueilla, joilla vesiliikennetiheys on suuri, tai joilla kuljetetaan vaarallisia aineita, tai joilla on paljon navigointiin liittyviä ongelmatekijöitä tai joiden ympäristö sekä meri- ja rannikkoluonto on erityisen herkkää. Päätökset, jotka koskevat aluksen navigointia ja hallintaa kuuluvat aina kuitenkin viime kädessä aluksen kapteenille. (IMO 2002)

5.2 Suomenlahden SRS-järjestelmä

Suomenlahden kansainvälisellä merialueella otetaan kesällä 2004 käyttöön alusten pakollinen ilmoittautumisjärjestelmä (SRS). SRS-toiminta tulee perustumaan osin jo VTS:n myötä hyväiksi havaituille toimintatavoille.

SRS mahdollistaa kaikkien merkittävien alusten seurannan koko Suomenlahden alueella, sillä se koskee kaikkia bruttovetoisuudeltaan yli 300 tonnia olevia aluksia. Seurannan tarkoituksena on hallita merenkulkuun liittyviä riskitekijöitä ja näin edistää yleistä meriturvallisuutta. (VTT 2002)

SRS:n toiminta on luonteeltaan hyvin samanlaista kuin VTS:n toiminta. Kun alus saapuu SRS-alueelle, se antaa ilmoituksen määrättyyn kansalliseen meriliikennekeskukseen, joka ottaa aluksen seurantaan ja informoi sitä tarpeen tullen merkityksellisistä asioista. SRS-operaattorit työskentelevät kansallisissa VTS-keskuksissa ja toimivat osin samoissa tehtävissä kuin VTS-operaattorit. (VTT 2002)

SRS-toiminta edellyttää käyttäjiltä eli SRS-operaattoreilta hajautettua yhteiskäyttöä. Tällä tarkoitetaan sitä, että käyttäjät ovat maantieteellisesti sijoittuneet kolmeen eri maahan (Suomi, Viro, Venäjä) ja samalla järjestelmän käyttöön liittyvä päätöksenteko on hajautunut. Varsinkin tilanteissa, joissa alus ylittää tietyn operaattorin hallitseman alueen rajan ja siirtyy samalla toisen operaattorin alueelle, tarvitaan operaattoreiden yhteistoimintaa.

5.3 Tapaustutkimuksen tavoitteet

Tapaustutkimuksen tavoitteena oli vastata kolmanteen tutkimuskysymykseen eli selvittää, minkälaisia vaatimuksia kognitiivisen työn analyysin kuvausmenetelmillä voidaan nostaa esiin. Tutkimuksessa selvitettiin myös löydettyjen vaatimusten merkityksellisyys suunniteltavan järjestelmän käyttäjän toiminnan kannalta.

5.4 Tutkimusmenetelmät ja tutkimuksen kulku

Tässä kappaleessa kuvataan käytetyt tutkimusmenetelmät. Niiden käyttö on jaettu tutkimuksen kulun mukaan neljään eri vaiheeseen: käyttäjätutkimukseen, vaatimusten kuvaukseen, kuvausten validointiin ja kuvausten arviointiin. Samalla tutkimuksen kulku tulee esitellyksi.

5.4.1 Käyttäjätutkimus

Tutkimuksen ensimmäinen osa oli käyttäjätutkimus, jonka avulla kerättiin empiirinen aineisto vaatimusten määrittelyä varten. Alla on esitetty käyttäjätutkimuksessa käytetyt menetelmät. Kohdan lopussa on listattuna kerätty empiirinen aineisto.

5.4.1.1 Contextual Inquiry -havainnointi

Käyttäjätutkimus suoritettiin osaksi Contextual Inquiry (CI) -tyyppisellä havainnoinnilla. Tutkija seurasi VTS-operaattoreiden työskentelyä Helsingin meriliikennekeskuksessa Merisatamassa yhteensä yhden päivän ajan.

Havainnoinnin aluksi tutkijalle esiteltiin keskuksen tilat sekä vuorossa olevat operaattorit. Samoin VTS-operaattorin työssään käyttämät välineet käytiin läpi. Tämän jälkeen tutkija seurasi yhden operaattorin työtä. Tänä aikana tutkija kirjasi ylös omia havaintojaan sekä esitti välillä tarkentavia kysymyksiä operaattorille hänen toimistaan. Tutkija pyrki muodostamaan havainnoitavaan käyttäjään mahdollisimman luonnollisen suhteen.

CI-havainnoin tuloksena tutkijalle muodostui kuva VTS-operaattorin rutiinitehtävistä sekä niihin käytettävistä työvälineistä.

VTS-operaattorit työskentelevät Helsingin meriliikennekeskuksessa kolmen hengen vuoroissa. Kaksi operaattoria hoitaa Helsingin alueen liikennettä ja yksi Hangon alueen. Usein operaattoreiden päätöksentekoon liittyy kuitenkin yhteistyöllisiä piirteitä. Varsinkin epäselvissä tilanteissa operaattorit tulkitsevat yhdessä elektroniselta kartalta havaitsemaansa liikennekuvaa sekä muodostavat sen perusteella yhteisiä päätöksiä tilanteisiin liittyvistä tarvittavista toimista.

5.4.1.2 Asiantuntijakeskustelujen havainnointi

Käyttäjätutkimus tehtiin osaksi VTT:n Tuotteet ja tuotanto -yksikön VTS-projektiin liittyvässä operatiivisessa harjoituksessa. Harjoituksen tarkoitus on määrittellä yhteiset toimintakäytännöt Suomen, Venäjän ja Viron yhdessä ylläpitämälle ja vuonna 2004 käyttöönotettavalle Suomenlahden SRS-järjestelmälle. SRS-toiminnan määrittely aloitettiin 19.–20.9.2002 järjestetyssä operatiivisessa harjoituksessa, johon osallistui VTS-henkilöstöä Suomesta, Venäjältä ja Viirosta. Tilaisuus järjestettiin Meriturvan simulaattoriyksikössä Otaniemessä ja sen suunnittelijoina ja vetäjinä toimivat VTT:n Tuotteet ja tuotanto -yksikön tutkijat.

Tilaisuuden keskeinen sisältö oli kokoneiden VTS-operaattoreiden keskustelu tulevan SRS:n tavoitteista, toimintaperiaatteista ja mahdollisista ongelmista. Keskusteluihin osallistui kolme VTS-operaattoria Suomesta, kaksi operaattoria sekä heidän esimiehensä Venäjältä sekä kaksi operaattoria Viirosta. Näiden lisäksi keskusteluissa olivat läsnä Suomen merenkulkulaitoksen sekä VTT:n edustajat. Tilaisuuden kulku oli etukäteen suunniteltu nelivaiheiseksi.

Tilaisuuden ensimmäisessä vaiheessa keskusteltiin SRS-toiminnan korkean tason tavoitteista. Asiantuntijat pohtivat ensin mitkä ovat SRS:n tavoitteet yleisen meri- ja alusturvallisuuden parantamisessa. Näin he pystyivät hahmottamaan suunniteltavan järjestelmän oleellisia toimintoja.

Asiantuntijakeskustelun toinen vaihe oli etukäteen valmisteltu siten, että SRS:n erilaiset liikennesidonnaiset tilanteet oli kirjattu keskustelun ohjelmaan. Tämä helpotti keskustelun siirtoa abstraktilta tasolta kohti käytännön toiminnan vaatimuksia. Keskustelijoiden lähettyvillä oli koko ajan Suomenlahden alueen merikartta, jota käytettiin erilaisten liikennetilanteiden keskustelun pohjana. Näin operaattorit pystyivät orientoitumaan eri tilanteisiin.

Asiantuntijat saavuttivat keskustelujen tuloksena yhteisymmärryksen osasta SRS:n käyttöönottoon liittyvistä vaatimuksista. Niitä toimintatapoja, joista keskustelussa oltiin päästy yhteisymmärrykseen, kokeiltiin tämän jälkeen simulointiharjoituksessa, joka muodosti tilaisuuden kolmannen vaiheen.

Järjestelmän kehittämisen kannalta asiantuntijakeskustelujen merkittävintä antia oli SRS-järjestelmän käyttöönottoon liittyvien erilaisten ongelmien eksplikointi.

Tämä tapahtui simulaatio-osuuksien purussa, joka muodosti operatiivisen harjoituksen neljännen ja viimeisen vaiheen.

Käyttäjätutkimuksena asiantuntijakeskustelujen havainnointi oli erittäin tehokas tapa tutustua kohdealueella vallitsevaan toimintaan ja kulttuuriin. Samalla jo CI:n perusteella muodostunut kuva VTS- tai SRS-operaattorin tehtävästä ja siihen liittyvistä vaatimuksista vahvistui. Operaattorin toiminnan kokonaistavoitteiden selkiytyminen oli asiantuntijakeskustelujen havainnoinnin suurin anti käyttäjätutkimuksen kannalta. Merenkulkuun liittyvässä työssä toimintakulttuurilla on suuri merkitys. Kulttuurin kautta toimintaan periytyvät vahvat käytännöt ja jännitteet selkiytyivät tutkijalle asiantuntijakeskustelujen havainnoinnin kautta.

5.4.1.3 Simulointi

Kolmas käyttäjätutkimuksen menetelmä oli asiantuntijakeskusteluissa kehitettyjen toimintatapojen simulointi. Tämä toteutettiin samassa yhteydessä kuin asiantuntijakeskustelutkin. Simuloinnit liittyivät aina samoihin liikennetilanteisiin, jotka olivat aiemmin olleet asiantuntijakeskusteluiden aiheena.

Simuloinnit suoritettiin samoissa Meriturvan tiloissa kuin keskustelutkin. Väliinena simuloinnissa käytettiin VHF-puhelimia, merikarttoja sekä puhelinyhteyttä. Näin liikenneilmoitusten vastaanottotilanne muodostui mahdollisimman todenmukaiseksi. Operaattorit suorittivat normaaleja työtehtäviään, kun laivoja simuloivat avustajat antoivat ilmoituksiaan. Kunkin yksittäiseen liikennetilanteeseen liittyvän simulointiharjoituksen kesto oli n. 15–20 minuuttia, jonka jälkeen harjoitus purettiin yhdessä osallistujien kanssa. Purkutilanteessa operaattorit pohtivat kuinka toimivia kokeillut toimintatavat olivat ja minkälaisia muutoksia niihin kenties tarvittaisiin.

Simuloinnin hyöty operatiivisen harjoituksen kannalta oli lähinnä kehitettyjen toimintatapojen arviointi. Osa tavoista havaittiin hyviksi, mutta jotkut käytännöt vaativat selkeästi vielä kehittämistä. Tilaisuuden tiukasta aikataulusta johtuen käytiin simuloinnissa läpi ainoastaan rutiinitilanteita. Erilaisissa poikkeus- ja hätätilanteissa käytetyt toimintatavat jäivät kokonaan vielä määrittelemättä.

Käyttäjätutkimuksen kannalta simulointi vahvisti edelleen SRS-operaattorin työnkuvaa. Simuloinnissa pystytään huomattavan tehokkaasti käymään läpi eri-

laisia liikennetilanteita, joita varsinaisen työn havainnoinnissa saattaisi joutua odottamaan erittäin pitkäänkin. Tämän tapaista korkean abstraktiotason simulointia voi siis pitää kustannustehokkaana tapana tehdä käyttäjätutkimusta. Toisaalta käyttäjät tuntuivat jännittävän simulointitilannetta. Tämä on aina otettava huomioon simuloinnin avulla saavutettavia tuloksia arvioitaessa, jolloin saatuihin tietoihin esimerkiksi tehtävien kestoista on suhtauduttava varauksella.

Empiirinen aineisto

Edellä esitellyillä menetelmillä kerätty tieto muodostaa tapaustutkimuksen empiirisen aineiston. Koska käyttäjätutkimus tehtiin vaatimusmäärittelyn tarkoituksia varten, aineistoa ole tässä tutkimuksessa kuvattu erillisinä, vaan menetelmän yhteydessä on selostettu myös tutkimuksen kulkua sekä kohteesta tehtyjä havaintoja.

Tutkimuksen empiirinen aineisto muodostuu seuraavista osista:

- CI-havainnoinnin muistiinpanot
- CI:n yhteydessä tehdyn haastattelun muistiinpanot
- asiantuntijakeskusteluissa käytetyt materiaalit (ohjelma, tehtäväkuvaukset)
- asiantuntijakeskustelujen muistiinpanot
- asiantuntijakeskusteluiden videonauhoitteet
- simuloinneissa käytettyjen tilannesidonnaisten tehtävien kuvaukset
- simulointien videonauhoitteet
- simulointien purkutilanteiden videonauhoitteet.

5.4.2 Vaatimusten kuvaaminen

Vaatimusten kuvaamisen menetelmänä käytettiin CWA:n ADS-taulukkoa, sekä siihen mallinnettavia päätöksentekoketjuja. ADS-taulukot tehtiin SRS:n kokonaistavoitteiden perusteella. Päätöksentekoketjut mallinnettiin muutamalle tyypilliselle järjestelmän käyttötilanteelle.

Mallinnustavan vertailukohtana käytettiin käyttötapausmalleja, jotka rakennettiin kuvaamaan samoja tilanteita kuin ADS:ien päätöksentekoketjut. Käyttöpaukset esitettiin skenaarioiden ja kaavioiden avulla.

5.4.3 Kuvausten validointi

Tutkijan tuottamat alustavat kuvaukset (Liite D) validoitiin asiantuntijakäyttäjän kanssa haastattelutilanteessa. Validoinnin tavoitteena oli selvittää olivatko vaatimusten kuvaukset tehty käyttäjän kannalta oikealla tavalla ja merkityksellisistä toiminnoista. Samalla havainnoitiin sopivatko käytetyt vaatimusten kuvausmenetelmät käyttäjäkeskeisen suunnittelun menetelmäksi. Edellytyksenä tälle on kuvaustavan ymmärrettävyys käyttäjän kannalta.

Validointi tehtiin strukturoimattomana haastatteluna, jossa tutkija ja asiantuntijakäyttäjä kävivät yhdessä läpi kuvattuja vaatimuksia, sekä niiden taustalla olevia SRS:n käyttötilanteita. Asiantuntijahaastatteluun osallistunut käyttäjä on kohdealueen asiantuntija, joka on osallistunut VTS-toiminnan kehittämiseen, toiminut itse VTS-operaattorina ja jolla on merikapteenin koulutus. Haastattelussa käytetyt tutkijan tuottamat mallit SRS-toiminnan vaatimuksista on esitetty tämän työn liitteissä ja lopulliset mallit kohdassa 5.4.

5.4.4 Kuvausten vertailu

Valmiiden kuvausten vertailun tavoitteena oli selvittää minkä tyyppisiä vaatimuksia ADS nostaa esiin. Vertailukohtana käytettiin käyttötapausten avulla ilmaistuja vaatimuksia. Vaatimusten tyypittely ja vertailu tehtiin tutkijan toimesta.

5.5 Tulokset

Tutkimuksen tulokset ovat ADS-taulukoina sekä UML:n käyttötapauksina esitettyjä kuvauksia järjestelmän vaatimuksista. Mallit olivat alun perin tutkijan tekemiä, mutta niihin tehtiin muutoksia validointitilanteessa.

Myös vaatimusten kuvausmenetelmien arvioinnit ovat osa tutkimuksen tuloksia, sillä tutkimustavoitteena oli tyypitellä ja vertailla kahdella eri menetelmällä esitettyjä vaatimuksia.

5.5.1 Järjestelmävaatimukset

Järjestelmävaatimuksista analyysin kohteeksi otettiin kolme käyttäjätutkimusvaiheessa merkityksellisiksi havaittua erilaista tilannetta. Ensimmäinen käsittelee rutiininomaista aluksen ilmoittautumistilannetta, toinen tilannetta, jossa operaattorit joutuvat selvittämään tunnistamattoman aluksen tiedot ja kolmas potentiaalista riskitilannetta, jossa kaksi alusta on risteävällä kurssilla. Näiden kaikkien tilanteiden tilannesidonnaiset vaatimukset mallinnettiin sekä ADS:n että UML:n käyttötapauksen avulla.

5.5.1.1 Vaatimukset ADS:lla kuvattuna

Kun vaatimuksia mallinnettiin ADS:lla, oli ensin pohdittava, mikä on se kohde, jota ollaan kuvaamassa. Tämä ei aina ole itsestään selvää. SRS:n ollessa suunnittelun kohteena mielletään työkohteeksi kaikki Suomenlahdella seilaavat alukset ja niiden ohjausjärjestelmät. Toisaalta myös ympäristön pinnanmuodot ovat osa järjestelmää samoin kuin VTS-keskukset ja niissä olevat järjestelmät. SRS:n käyttäjiä ovat VTS-operaattorit sekä alusten henkilöstö. Alla esiteltävät ADS-mallit kuvaavat siis Vicenten (1999) mukaan määriteltyä järjestelmän ympäristöä. Tämä on koko fyysinen todellisuus, jonka laivat merellä seilatessaan muodostavat.

Alustavien ADS:ien muodostamisessa sovellettiin Lindin (1999) neuvoa ja käytettiin ainoastaan neljää abstraktio-hierarkian tasoa. Alun perin ADS on kehitetty erilaisten kemiallisten prosessien mallintamiseen, ja hierarkiatasojen määrä on säädetty tälle käyttötarkoitukselle sopivaksi. Lind (1999) huomaa tasojen määrän olevan riippuvainen mallinnettavasta sovellusalueesta ja suosittelee yleisesti käytettävän 3–5 tasoa. Tutkijan tuottamissa malleissa oli järjestelmän fyysinen hierarkia esitetty viitenä eri tasona. Asiantuntijahaastattelun perusteella tämä kuitenkin muutettiin kolmeksi tasoksi, jolloin malli kuvaa kohdetta paremmin nimenomaan SRS-operaattorin näkökulmasta.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät
Ympäristön ja ihmishenkien suojele, meriliikenneturvallisuus			
Alusten häiriötön navigointi			
Liikennetilannetiedon välittyminen osapuolille			
Kurssit, nopeudet, sää, ympäristö, alustihentymien välttäminen			

Kuva 21. ADS-malli SRS-toiminnasta.

ADS:n muodostamisessa (kuva 21) lähdetään liikkeelle hajottelemalla mallinnettava kohde kokonaisuudesta mahdollisimman pieniksi osiksi. Näin muodostuu ADS:n yläriivi. SRS:n tapauksessa kokonaisuus on usean liikenteessä olevan aluksen sekä niiden ympäristön muodostama. Kun tätä lähdetään jaottelemaan pienempiin osiin päästään ensin yksittäiseen alukseen ja sen tiettyyn ympäristöön. Tämän jälkeen SRS:n kannalta seuraava merkittävä hierarkiataso on laivan komentosillalla sijaitsevat erilaiset sen ohjaukseen ja hallintaan tarkoitetut järjestelmät. Näillä tarkoitetaan myös yhteydenpitoon ja ympäristön havainnointiin tarkoitettuja laitteita ja välineitä.

ADS:n pystysarake on niin sanottu tavoite-keino hierarkia. Siinä on määritelty kokonaisjärjestelmän tavoite ja tämän jälkeen hierarkkisesti alaspäin ne keinot, jolla tavoitteeseen voidaan päästä. SRS:n kokonaistavoite on alus- ja meriturvallisuus, joka tähtää yksinkertaisesti ympäristön ja ihmishenkien suojelemaan. Tähän voidaan päästä edistämällä alusten turvallista navigointia. Turvallisen navigoinnin edellytys taas on SRS:n kannalta oikean liikennetilannetiedon välittyminen kaikille turvallisuudesta vastuussa oleville osapuolille. Näitä osapuolia ovat eri

alusten komentosilloilla työskentelevä henkilöstö sekä eri maiden SRS-operaattorit. Jotta liikennetilanteesta välittynyt kuva olisi mahdollisimman todennukainen, on siinä otettava huomioon alusten todelliset kurssit ja nopeudet sekä ympäristön tila ja esimerkiksi sääolosuhteet. Tärkeää on myös välttää alustihentymien muodostamista, sillä ”ruuhkat” lisäävät liikennetilanteeseen liittyvän tiedon epävarmuutta. Näin aikaansaata ADS:ää SRS:n kohteesta voidaan käyttää yleisluontoisena pohjana, johon kuvataan erilaisia järjestelmän käyttöön liittyviä tilanteita.

ADS:ään mallinnettiin erilaisia tilanteita tapahtumaketjuina. Kuvassa 22 esitetään rutiinitilanne, jossa alus on saapumassa SRS-alueelle. Tapahtuma etenee niin, että alus ilmoittaa alueelle saapumisestaan SRS-keskukseen VHF-puhelimen avulla. Tämän jälkeen operaattori paikantaa aluksen tutkakuvastaan ja tarkistaa onko ennakkoraportti annettu, eli ovat kaikki alukseen liittyvät tiedot kunnossa. Tämän jälkeen operaattori ottaa aluksen seurantaan eli merkitsee seurantanäyttöön aluksen tiedot. Jos aluksen reitillä ei tunnu olevan ongelmia, alkaa operaattori passiivisen aluksen kulun monitoroinnin.

SRS-operaattorin päätöksenteossaan tarvitsema tieto vaihtelee kaikkien alusten muodostaman kokonaisliikennekuvan ja aluksen komentosillan järjestelmien sekä kokonaistavoitteiden ja oikean liikennekuvan välittymisen välillä.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosil- lan järjes- telmät
Ympäristön ja ihmishenkien suo- jelu, meriliikenne- turvallisuus	Alus ilmoittaa saapuvansa SRS-alueelle Monitoring		
Alusten häiriötön navigointi		"Reitti selvä"	
Liikennetilannetiedon välittyminen osapuolille	Alus näkyy tutkakuvassa	Alus seurantaan	Ennakkoraportti annettu?
Kurssit, nopeudet, sää, ympäristö, alustihentymien välttäminen		"Onko kaikki kunnossa"?	

Kuva 22. ADS-malli tilanteesta, jossa alus on saapumassa SRS-alueelle.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät
Ympäristön ja ihmishenkien suojele, meriliikenneturvallisuus	Monitorointi	Alueen rajaa ylittämässä operaattorin järjestelmien avulla tunnistamaton alus	Nimen tiedustelu VHF-puhelimen avulla, aluksen kutsuminen paikkatiedon, suunnan ja nopeuden perusteella
Alusten häiriötön navigointi		Huomautus alukselle	
Liikennetilannetiedon välittyminen osapuolille	Soitto toiseen SRS-keskukseen ja aluksen tietojen tiedustelu	Aluksen kutsu nimellä VHF-puhelinta käyttäen	Saapumisasi- moituksen anto SRS-keskukselle
Kurssit, nopeudet, sää, ympäristö, alustihentymien välttäminen			

Kuva 23. ADS-malli tilanteesta, jossa operaattori havaitsee tunnistamattoman aluksen alueella.

Seuraavassa mallinnetussa tilanteessa (kuva 23) operaattori havaitsee aluksen, joka ei ole ottanut yhteyttä keskukseseen ilmoittaakseen alueelle saapumisestaan. Tässä tilanteessa operaattorin on yritettävä tunnistaa alus jollain keinolla. Hän voi yrittää tiedustella VHF-puhelimen avulla laivan nimeä kutsuen tätä paikan, nopeuden ja kurssin perusteella kutsukanavalla. Tämä on kuitenkin suhteellisen tehoton toimenpide. Toinen vaihtoehto on selvittää edellisen alueen (toinen SRS-keskus tai VTS-keskus) kautta aluksen nimi. Nimen selvitettyään SRS-operaattori voi kutsua alusta sen oikealla nimellä, jolloin yhteyden saaminen suurella todennäköisyydellä onnistuu. Yhteyden saamisen jälkeen operaattori pyytää alusta antamaan ilmoittautumisessa tarvittavat tiedot. Tässäkin tilanteessa

operaattorin tarvitsema tieto liittyy lähinnä oikeaan liikennekuvaan ja yksittäisen aluksen sijaintiin ja komentosillan järjestelmiin.

Kolmas mallinnettu tilanne (Kuva 24) on potentiaalinen riskitilanne, jossa kaksi alusta on risteävällä kurssilla ja SRS-operaattori kokee tarpeelliseksi puuttua tilanteeseen. Havainnon tehtyään operaattorin tulee varmistaa, että se on oikea, eli tällöin hän tutkii liikennekuvan epävarmuuteen vaikuttavia tekijöitä. Tällaisia ovat esimerkiksi sääolosuhteet. Todellisen törmäysriskin selvittämiseksi operaattorin on tiedettävä myös alusten todelliset suunnat ja nopeudet, jolloin hän pohtii myös tilanteen kiireellisyyttä. Jos näiden perusteella näyttää siltä, että törmäysriski on olemassa, operaattori ottaa yhteyden väistämisvelvolliseen alukseen ja antaa toimintaohjeita.

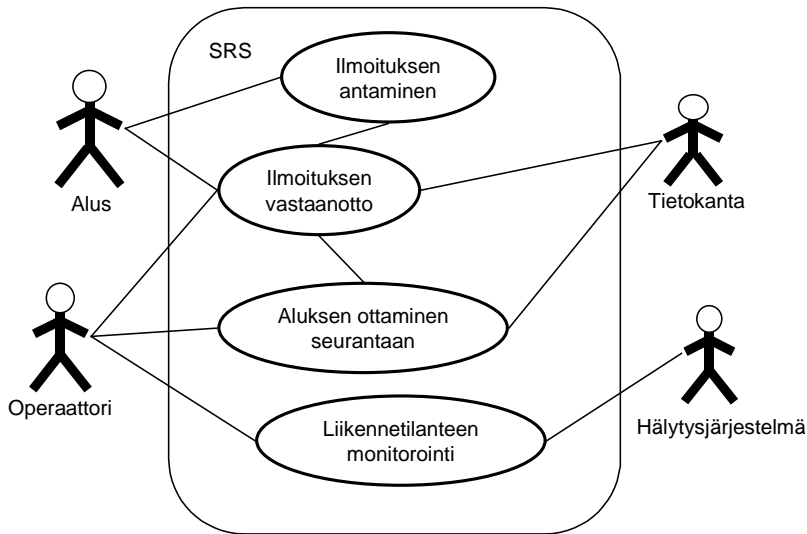
	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät
Ympäristön ja ihmishenkien suojeleminen, meriliikenneturvallisuus	Tutkakuva: kaksi alusta risteävällä kurssilla		Korjaavat toimenpiteet
Alusten häiriötön navigointi		Tilanteen kiireellisyys, alusten nopeudet?	
Liikennetilantiedon välittyminen osapuolille			Yhteys aluksiin: Tilanne huomattu?
Kurssit, sää, ympäristö, alustihentymien välttäminen	Onko liikennekuva oikea?		

Kuva 24. ADS-malli SRS-järjestelmästä monitorointitilanteessa, jossa alukset risteävällä kurssilla.

5.5.1.2 Vaatimukset käyttötapauksina

Samat tilanteet, jotka edellä on esitetty ADS:ien avulla on alla esitetty UML:n käyttötapauksina. Jokaista ADS:ää vastaavasta tilanteesta muodostettiin käyttötapauskaavio, jossa näkyy käyttötapauksen keskinäiset suhteet sekä niihin liittyvät toimijat. Niille käyttötapauksille jotka sisältävät selkeästi operaattorin tai jonkin muun toimijan aktiivisia toimintoja muodostettiin myös skenaariot.

Käyttötapaukset muodostettiin melko pienistä järjestelmän käyttöön liittyvistä tapahtumista. Näin myös niiden perusteella kirjoitetut skenaariot ovat hyvin lyhyitä. Näin pystyttiin tekemään skenaarioista tarpeeksi yksinkertaisia ja yksityiskohtaisia. Juuri yksityiskohtaisuus vaikuttaa positiivisesti siihen, että vaatimukset on mahdollista arvioida käyttäjien kanssa.



Kuva 25. Käyttötapauskaavio aluksen saapumisesta SRS-alueelle.

Ensimmäinen käyttötapauskaavio (kuva 25) kuvaa suurin piirtein samaa tilannetta kuin ensimmäinen ADS-mallikin. Kyseessä on rutiininomainen tilanne, jossa alus saapuu SRS-alueelle, antaa ilmoituksen keskuksessa työskentelevälle operaattorille, joka ottaa aluksen seurantaan ja valvoo sen kulkua yleisellä liikennetilanteen monitoroinnilla. Alla on esitetty myös käyttötapauskaavio kolmeen tärkeimpään käyttötapaukseen liittyvät skenaariot (kuvat 26, 27 ja 28).

Nimi:	Ilmoituksen antaminen
Suorittajat:	Alus, Officer on Watch
Alkuehdot:	Alus ylittämässä SRS-alueen rajan, ennakoilmoitus annettu
Kuvaus:	Alus kutsuu VHF-puhelimen kutsukanavalla SRS-keskusta. Operaattori vastaa. Molemmat siirtyvät VHF:n työkanavalle. Alus antaa lyhyen ilmoituksen. Operaattori kuittaa ja kehottaa palaamaan kutsukanavalle.
Poikkeukset:	Ennakoilmoitusta ei ole annettu. Tällöin alus antaa koko ilmoituksen VHF:n työkanavalla. Työskentelykanava on varattu. Tällöin alus antaa ilmoituksen kutsukanavalla
Loppuehdot:	Operaattori ottanut vastaan ilmoituksen ja tunnistanut aluksen seurantanäytöltään.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

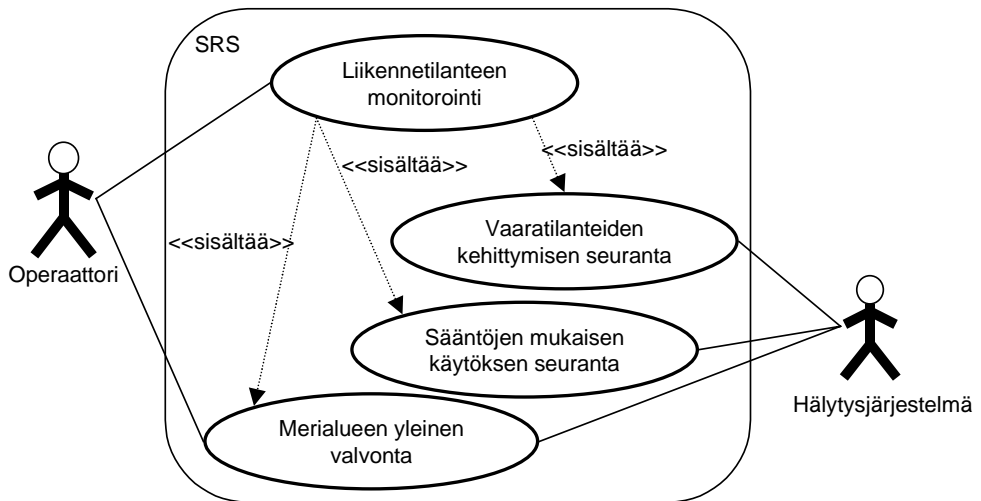
Kuva 26. Käyttötapausmalliin liittyvä skenaario ilmoituksen antamisesta.

Nimi:	Ilmoituksen vastaanotto
Suorittajat:	SRS-keskuksen operaattori, tietokanta
Alkuehdot:	Alus ylittämässä SRS-alueen rajan, ennakoilmoitus annettu
Kuvaus:	Operaattori vastaa aluksen kapteenin kutsuun VHF-puhelimen kutsukanavalla. Molemmat siirtyvät VHF:n työkanavalle. Operaattori vastaanottaa lyhyen alueelle saapumisilmoituksen ja päivittää tietokantaansa. Operaattori kuittaa ja kehottaa alusta palaamaan kutsukanavalle.
Poikkeukset:	Ennakoilmoitusta ei ole annettu. Operaattori vastaanottaa pitkän raportin ja päivittää samalla tietokantaansa kaikki tiedot.
Loppuehdot:	Operaattori ottanut vastaan ilmoituksen ja tunnistanut aluksen seurantanäytöltään.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Kuva 27. Käyttötapausmalliin liittyvä skenaario ilmoituksen vastaanotosta.

Nimi:	Aluksen ottaminen seurantaan
Suorittajat:	SRS-keskuksen operaattori, tietokanta
Alkuehdot:	Alus tunnistettu, ilmoitukset annettu
Kuvaus:	Operaattori merkitsee tutkakuvaansa ilmoituksen antaneen aluksen seurantatiedot. Muiden operaattoreiden tutkakuvat päivittyvät samalla.
Poikkeukset:	-
Loppuehdot:	Aluksen tiedot näkyvät tutkakuvassa.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

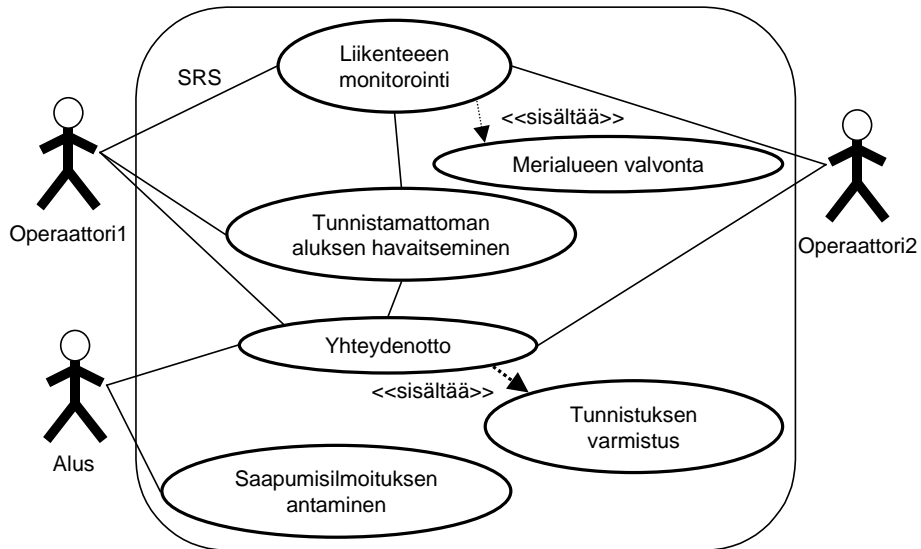
Kuva 28. Käyttötapaumalliin liittyvä skenaario aluksen ottamisesta seurantaan.



Kuva 29. Käyttötapauskaavio liikennetilanteen monitoroinnista.

Toinen käyttötapauskaavio (kuva 29) on yleinen kuvaus liikennetilanteen monitoroinnista. Käyttötapaukseen on otettu toiseksi toimijaksi suunnitella oleva hälytysjärjestelmä, joka tarvittaessa kiinnittää operaattorin huomion potentiaaliin vaaratilanteisiin. Toisen kaavion käyttötapauksista ei kirjoitettu skenaarioita, sillä ne sisältävät lähinnä operaattorin ja hälytysjärjestelmän passiivisia valvontatehtäviä.

Kaksi viimeistä käyttötapauskaaviota (kuvat 30 ja 32) tarkentavat edellistä ja esittävät kaksi erillistä mahdollista riskitilannetta, joihin SRS-operaattori voi halutessaan puuttua. Ensimmäinen näistä koskee tilannetta (kuva 31), jossa operaattori havaitsee tunnistamattoman aluksen saapuvan alueelle. Tyypillisesti tällainen tilanne voi muodostua, kun alus on esimerkiksi vain unohtanut antaa saapumisilmoituksen tai ei ole tietoinen alueen rajan ylityksestä. Tällöin operaattorin on otettava yhteys alukseen jollain keinolla. Hän voi kutsua sitä sen paikkakoordinaattien, nopeuden ja kurssin perusteella. Toinen vaihtoehto on selvittää edellisen VTS- tai SRS-alueen operaattorilta aluksen nimi ja kutsua sitä nimellä. Tämä tietysti toimii vain silloin, kun alus on saapumassa jonkin järjestelmän alaiselta alueelta. Tähän käyttötapauskaavioon liittyen kirjoitettiin yksi skenaario. Tämä kuvaa toimenpiteitä, jotka liittyvät tunnistamattoman aluksen tietojen selvittämiseen ja kattaa näin muutaman kaaviossa esitetyistä käyttötapauksista.

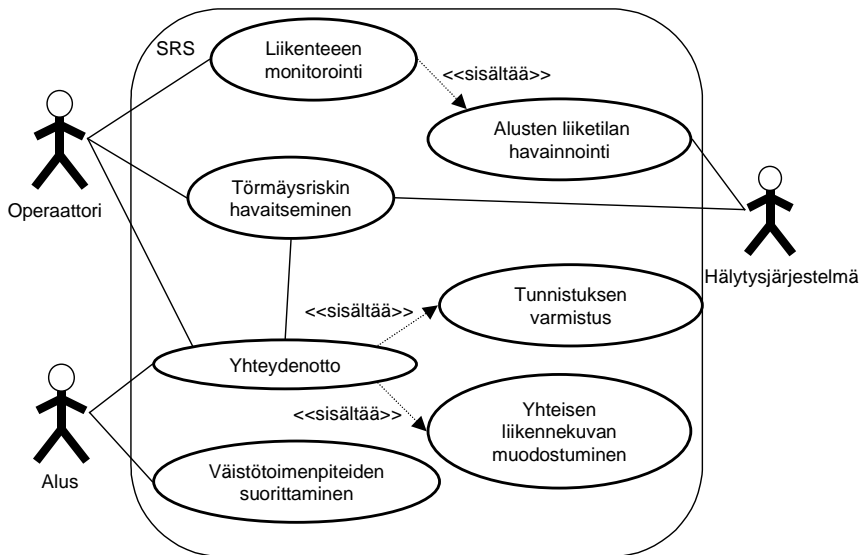


Kuva 30. Tarkennettu käyttötapauskaavio liikennetilanteen monitoroinnista tilanteessa, jossa alus ei ole antanut ilmoitusta saapumisestaan SRS-alueelle.

Nimi:	Tunnistamattoman aluksen havaitseminen
Suorittajat:	SRS-keskuksen operaattori, alus, toisen keskuksen operaattori
Alkuehdot:	Alus saapumassa SRS-alueelle toiselta valvotulta alueelta
Kuvaus:	Operaattori havaitsee aluksen olevan saapumassa SRS-alueelle. Kyseisen aluksen tietoja ei näy operaattorin tutkakuvassa, joten se ei ole seurannassa, eikä siten antanut saapumisilmoitusta. Operaattori kutsuu alusta sen paikkakoordinaattien mukaan VHF-puhelimen kutsukanavalla. Alus huomaa kutsun ja vastaa siihen. Molemmat siirtyvät työkanavalle ja alus antaa saapumisilmoituksen.
Poikkeukset:	Alus ei huomaa kutsua kutsukanavalla. Operaattori ottaa yhteyttä toiseen keskukseseen ja kysyy aluksen tietoja. Toinen keskus antaa aluksen nimen. Operaattori kutsuu alusta nimellä kutsukanavalla. Alus vastaa kutsuun. Molemmat siirtyvät työkanavalle ja alus antaa saapumisilmoituksen.
Loppuehdot:	Alus antaa saapumisilmoituksen
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja, alus tulossa toiselta meriliikenteenohjausalueelta

Kuva 31. Käyttötapausmalliin liittyvä skenaario tunnistamattoman aluksen saapumisesta SRS-alueelle.

Viimeinen käyttötapauskaavio (kuva 32) kuvaa tilannetta, jossa kaksi alusta on risteävällä kurssilla, ja operaattori kokee tarpeelliseksi puuttua tilanteeseen. Kaavio ei ota kantaa niihin syihin, joiden perusteella operaattori haluaa puuttua tilanteeseen. Näitä voi kuitenkin olla monia. Esimerkiksi huonoista sääolosuhteista johtuva huono näkyvyys, tai jokin muu tunne siitä, että alukset eivät välttämättä ole havainneet toisiaan, eivätkä tunne kyseistä merialuetta tarpeeksi hyvin tehdäkseen väistöliikkeitä. SRS-operaattori ei kuitenkaan anna varsinaisia ohjeita siitä, miten alusten tulee tilanteessa toimia. Hän vain tiedustelee onko tilanne havaittu komentosilloilla ja näin samalla hienovaraisesti huomauttaa aluksille tulkinneensa mahdollisen riskitilanteen olevan muodostumassa. Lopulliset toimenpiteet tilanteen välttämiseksi suorittavat alukset.



Kuva 32. Tarkennettu käyttötapauskaavio tilanteesta, jossa kaksi alusta on ristteävällä kurssilla.

Käyttötapauskaavion perusteella kirjoitettiin yksi skenaario (kuva 33), joka kuvaa törmäysriskin havaitsemiseen liittyviä toimenpiteitä.

Nimi:	Törmäysriskin havaitseminen
Suorittajat:	SRS-keskuksen operaattori, Alus
Alkuehdot:	Alukset tunnistettu, ilmoitukset annettu jne.
Kuvaus:	Operaattori havaitsee tutkakuvastaan kahden aluksen olevan lähestymässä toisiaan. Alusten liiketilan perusteella operaattori havaitsee törmäysvaaran olevan syntymässä. Operaattori kutsuu väistämisvelvollista alusta ja tiedustelee onko tilanne havaittu komentosillalla. Alus suorittaa tarvittavat toimenpiteet törmäysuhan välttämiseksi.
Poikkeukset:	–
Loppuehdot:	Alukset ohittavat toisensa turvallisesti
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Kuva 33. Käyttötapausmalliin liittyvä skenaario törmäysvaaran välttämisestä.

5.5.2 Arviot vaatimuksista ja kuvausmenetelmistä

Kohdealueen asiantuntija on haastattelutilanteessa arvioinut sitä, miten hyvin vaatimukset kuvaavat kohdealueen problematiikkaa. Asiantuntijahaastattelun yhteydessä kirjoittaja on myös kiinnittänyt huomiota siihen, miten hyvin vaatimusten esitystapa sopii käyttäjän kanssa validoitavaksi.

Eri menetelmien avulla kuvattuja vaatimuksia on vertailtu kirjoittajan toimesta. Tässä on lähinnä kiinnitetty huomiota siihen, minkä tyyppisiä vaatimuksia eri tekniikat nostavat esiin ja mikä mahdollisesti voisi olla erityyppisten vaatimusten rooli järjestelmän suunnittelussa.

5.5.2.1 Asiantuntijahaastattelu: vaatimusten validointi

Asiantuntijahaastattelussa tutkijan ADS:lla ja käyttötapauksilla kuvaamat vaatimukset todettiin hyväksi työkaluksi saada käyttäjä ja suunnittelija keskustelemaan suunniteltavan järjestelmän ominaisuuksista. Käyttäjä totesi piirrettyjen mallien helpottavan oman työn reflektointia ja abstrahointia.

Selvää eroa kahden eri mallinnustavan välille ei kuitenkaan muodostunut. Käyttäjä totesi ADS:ien ja UML:n käyttötapauksien olevan yhtä helposti omaksuttavia tapoja esittää SRS-toimintaa. Hänen mukaansa käyttötapauksien kautta on kuitenkin hieman helpompi hahmottaa kokonaistoimintaa.

Tutkijan tuottamiin vaatimuksiin tehtiin muutamia muutoksia validoinnin yhteydessä. Suurin muutos oli ADS:n fyysisen hajotelman tasojen vähentäminen kolmeen. Asiantuntijan mukaan SRS-operaattorin toiminnan kohde ei ulotu tätä syvemmälle laivojen ohjailuun liittyviin järjestelmiin. Tämän lisäksi korjattiin termistöä ja sanamuotoja.

Asiantuntijahaastattelun antia arvioitaessa on muistettava, että siihen osallistunut käyttäjä on osallistunut SRS-toiminnan kehittämiseen, ja siten ehkä tavallista riviikäyttäjää tottuneempi mallien lukija ja tulkitsija. Hänellä on myös erittäin selkeä kuva VTS-operaattorin tehtävistä ja niihin liittyvistä ongelmakohtista.

5.5.2.2 Vaatimusten kuvaustapojen vertailu

Edellä esitetyt, CWA:n ADS:ien ja UML:n käyttötapausten avulla mallinnetut SRS:n vaatimukset ovat jo lähtökohdiltaan hyvin erilaisia. ADS:lla mallinnetaan kohdetta, kun taas käyttötapauksilla mallinnetaan kohteeseen liittyvää toimintaa. Tämä muodostaa perustavaa laatua olevan eron tekniikoiden välille. Tämän lisäksi kuvaustavat eroavat toisistaan kolmella eri tavalla:

Ensimmäinen ero liittyy mallintamisessa käytettyyn näkökulmaan. Lähdeettäessä muodostamaan ADS:aa on ensin valittava näkökulma kohteeseen. Esimerkiksi aluksen kapteenilla ja SRS-operaattorilla on molemmilla oma käsityksensä siitä, mikä on SRS-toiminnan kohde. Tutkimuksessa havainnoiduissa asiantuntijakeskusteluissa kävi ilmi, että myös eri maiden SRS-operaattoreilla voi olla erilainen näkemys toiminnan tavoitteista. Valittava näkökulma on tärkeä, sillä ADS:n muodostaminen perustuu toiminnan kohteen hierarkkiseen osittamiseen. Varsinkin pystysarakkeen keino-tarkoitushierarkiaa jaoteltaessa joudutaan tekemään mallin kannalta ratkaisevia valintoja siitä, mikä on tarkoituksen saavuttamisen kannalta merkittävin keino. Tämä on näkökulman valinnan kannalta erittäin merkittävä kohta. Tässä mallintaja joutuu tekemään oletuksia siitä, mihin keinoihin mallinnettavalla järjestelmällä on mahdollista vaikuttaa. Vicente (1999) ei ole käsitellyt näkökulman valitsemiseen liittyvää problematiikkaa. Hänen mukaansa ADS kuvaa kohteen muuttumattomina pysyviä ominaisuuksia, mutta tällöin ei huomioida ollenkaan käyttäjän suhdetta kohteeseen. Eri käyttäjien kannalta kohde on erilainen, eikä minkäänlaista objektiivista mallia siten pystytä muodostamaan. UML:n käyttötapauksissa vastaavaa näkökulman valintaan liittyvää ongelmaa ei ole, sillä käyttötapaukset kirjoitetaan aina käyttäjän näkökulmasta.

Toinen mallinnusmenetelmien ero liittyy mallinnustapaan. ADS:ssa mallinnustapa on formatiivinen, kun taas käyttötapauksissa se on joko normatiivinen tai deskriptiivinen. Tämä näkyy jo siinä, että ADS mallintaa kohdetta ja käyttötapaukset taas toimintaa sellaisena kuin sen pitäisi tapahtua tai se tapahtuu tällä hetkellä. Formatiiivisuuden käsite on helpommin sovellettavissa kohdetta mallinnettaessa. Tässä tapauksessa formatiivinen mallinnus tarkoittaa sitä, että ADS:n avulla määritellään, millä järjestelmän osa-alueilla kulloinkin liikutaan. Näin saadaan selville, minkä tyyppistä tietoa operaattori käyttää ja tarvitsee päätöksenteossaan. ADS:ssa ei kuitenkaan ilmaista, mikä on se yksityiskohtainen tieto,

minkä operaattori tarvitsee, vaan malli ilmaisee, mitä kriittistä keinoa tämä tieto koskee. Kriittiset keinot ovat pystysarakkeen keino-tarkoitushierarkian eri tasot.

Kolmas ero vaatimusten kuvaustapojen välillä on se, että ADS-malli on luonteeltaan staattisempi kuin käyttötapaukset. ADS kuvaa järjestelmän käyttöliittymän informaation sisältöä, ja käyttötapaukset kuvaavat järjestelmän toimintoja. Käyttöliittymän informaation sisältöön ei käyttötapausmallinnuksessa oteta kantaa. Sama ero näkyy siinä, että käyttötapaukset ovat aina tilannesidonnaisia ja niihin sisältyy dynaamisia muutoksia. ADS:ään tulee tilannesidonnaisuutta mukaan päätöksentekoketjujen avulla. Ilman päätöksentekoketjua ADS esittää kohteen hallintaan liittyviä keinoja, sekä sitä miten ne vaikuttavat kohteen eri osiin.

5.6 Tapaustutkimuksen johtopäätökset

Sekä CWA:n ADS-kaaviot että UML:n käyttötapauskaaviot ovat toimivia menetelmiä SRS:n vaatimusten käyttäjälähtöiseen kuvaamiseen. Ne esittävät kuitenkin järjestelmää hyvin erilaisista näkökulmista, joten myös niiden avulla esitetyt vaatimukset ovat luonteeltaan erityyppisiä.

Käyttötapaukset sopivat järjestelmäsuunnittelun vaiheeseen, jossa pohditaan, minkälaisia toimintoja suunniteltavan systeemin pitäisi sisältää. Käyttötapauksen avulla voidaan selkeästi esittää järjestelmän eri käyttäjät ja heidän tarvitsemansa toiminnot. Samoin tehtävien ja toimintojen jakautuminen käyttäjien, automaation ja osajärjestelmien kesken voidaan esittää käyttötapauksen avulla. Käyttötapaukset välittävät hyvin järjestelmien dynaamisen luonteen. Samoin niillä on mahdollista kuvata järjestelmään liittyviä riskejä tai poikkeamia rutiinisuoritus-tavoissa.

ADS:t kuvaavat suunniteltavan järjestelmän käyttöliittymän informaation sisältöä. Tämä tehdään järjestelmän käyttäjän tarpeisiin perustuen. Siten ne sopivat käytettäväksi silloin, kun ollaan suunnittelemassa yksittäisiä käyttöliittymiä. ADS:n avulla voidaan mallintaa, mikä tieto kohteesta on oleellista käyttäjälle hänen operoidessaan järjestelmää.

ADS-mallit sopivat käyttäjän kanssa tehtävään validointiin. Tämä kuitenkin edellyttää, että mallin muodostaminen selitetään käyttäjälle perusteellisesti. Va-

lidointiin osallistuvan käyttäjän on myös kyettävä pohtimaan omaa toimintaansa sen tavoitteiden ja erilaisten osatehtävien ja kokonaisuuksien näkökulmasta.

ADS:t ja käyttötapaukset ovat toisiaan täydentäviä järjestelmien suunnittelussa käytettäviä mallinnusmenetelmiä. Niiden käyttötarkoitukset ovat erilaisia, eikä niitä siten voi vertailussa asettaa täysin samalle viivalle. Suunnitteluprosessin oikeassa vaiheessa käytettynä voidaan molempia mallinnustekniikoita käyttää hyödyksi toteutettaessa käyttäjäkeskeisen suunnittelun periaatteita.

6. Yhteenveto ja pohdinta

Tässä luvussa esitetään tutkimuskysymysten vastaukset sekä pohditaan työn onnistumista ja tuloksia sekä mahdollisia jatkotutkimuksen aiheita. Ensin esitetään työn tutkimusongelman alakysymysten vastaukset. Ylätason tutkimusongelman vastaus on kohdassa 6.2, jossa pohditaan työtä kokonaisuutena.

6.1 Vastaukset tutkimuskysymyksiin

Tämän työn johdannossa määriteltiin kolme erillistä tutkimuskysymystä. Tutkimuskysymysten vastaukset on esitetty alla. Ensimmäisen ja toisen tutkimuskysymyksen vastaukset perustuvat enemmän työn kirjallisuusosaan ja kolmannen kysymyksen vastaus enemmän käytännön osassa tehtyihin havaintoihin ja päätelmiin.

6.1.1 Ensimmäinen tutkimuskysymys

Ensimmäinen tutkimuskysymys pohtii, mitä erityisvaatimuksia monimutkaisen järjestelmän suunnittelu aiheuttaa käyttäjätutkimukselle. Käyttäjätutkimuksia käsiteltiin kirjallisuuden perusteella luvussa 3, jossa todettiin, että käyttäjätutkimuksen ensisijainen tarkoitus on selvittää suunniteltavan järjestelmän käyttäjien tarpeet järjestelmän suhteen. Käyttäjätutkimuksen tavoitteet saavutetaan määrittelemällä suunniteltavan järjestelmän käyttökonteksti.

Ensimmäinen tutkimuskysymys:

Mitkä käyttäjätutkimusmenetelmät soveltuvat monimutkaisen järjestelmän tiedonkeräysmenetelmiksi?

Luvussa 3 todettiin, että monimutkaisen järjestelmän suunnittelun vaatimusten määrittelyä varten tehtävän käyttäjätutkimuksen on nostettava esiin tulevan järjestelmän käytön kannalta merkityksellisiä ominaisuuksia. Tällaisia ominaisuuksia ovat esimerkiksi järjestelmän dynaaminen käyttäytyminen, sen käyttöön sisältyvät suuret riskitekijät sekä työn yhteistoiminnallinen luonne.

Contextual Inquiry (CI) on menetelmä, joka sopii monimutkaisen järjestelmän käyttäjätiedon keräysmenetelmäksi. CI perustuu tutkijan tekemään havainnointiin, joka suoritetaan aidossa käyttöympäristössä. Kun havainnointi tehdään oikeassa ympäristössä ja avoimuuden periaatteita noudattaen, pystyy tutkija muodostamaan aidon kuvan monimutkaisen järjestelmän käyttötilanteesta. Koska käyttäjät havainnointitilanteessa käyttävät järjestelmää oikeasti, tutkijalle muodostuu totuudenmukainen kuva järjestelmän dynaamisesta käyttäytymisestä. CI:hin kuuluvien kysymysten avulla tutkija pystyy myös selvittämään järjestelmään liittyviä riskejä ja uhkatekijöitä. Tässä on tosin otettava huomioon mahdollinen käyttäjien häiriintyminen kysymysten vaikutuksesta. Viettämillä aikaa aidossa käyttöympäristössä CI:tä tekevä tutkija pystyy myös havainnoimaan monimutkaisten järjestelmien käyttöön liittyvää yhteistoimintaa.

Eritasoiset simuloinnit ovat toinen monimutkaisten järjestelmien suunnitteluun sopiva käyttäjätiedon keräysmenetelmä. Simulointi mahdollistaa erityisesti järjestelmien käyttöön liittyvien uhka- ja riskitekijöiden havainnoinnin, sillä simulaattoreiden avulla voidaan tutkia, miten käyttäjät toimivat tietyissä poikkeustilanteissa. Turvallisuuskriittisten järjestelmien hätätilannekäytön tutkiminen on oikeastaan mahdollista vain simuloinnilla ja onnettomuustutkinnalla.

Usein simulaattoreiden käyttö käyttäjätutkimusmielessä on mahdollista, sillä monelle monimutkaiselle järjestelmälle on koulutustarkoituksia varten olemassa korkean tason simulaattoreita. Näiden avulla käyttäjien toimintaa voidaan tutkia hyvin monissa erilaisissa käyttötilanteissa.

6.1.2 Toinen tutkimuskysymys

Toinen tutkimuskysymys pohtii, mitä erityisvaatimuksia monimutkaisen järjestelmän suunnittelu aiheuttaa vaatimusten kuvaustavalle. Vaatimusten määrittelyn prosessia esiteltiin kirjallisuuden perusteella luvussa 4.

Toinen tutkimuskysymys:

Mitkä vaatimusten kuvausmenetelmät tukevat monimutkaisten järjestelmien ominaispiirteiden mallintamista?

Tämän työn neljännessä luvussa todettiin, että monimutkaisen järjestelmän suunnittelun vaatimusten määrittelyssä on nostettava esiin tulevan järjestelmän käytön kannalta merkityksellisiä ominaisuuksia. Tällaisia ominaisuuksia ovat esimerkiksi järjestelmän dynaaminen käyttäytyminen, sen käyttöön sisältyvät suuret riskitekijät sekä työn yhteistoiminnallinen luonne.

Vaatimusten kuvaustavan on oltava ilmaisuvoimainen jotta edellä mainitut monimutkaisuutta ilmentävät ulottuvuudet voitaisiin kattavasti kuvata. Myös kuvaustavan ymmärrettävyyteen on kiinnitettävä huomiota, sillä käyttäjakeskeisessä prosessissa jo vaatimusvaihe voidaan validoida käyttäjillä.

UML:n erilaiset kuvausmenetelmät sopivat monimutkaisten järjestelmien vaatimusten kuvaamiseen. Käyttötapaukset ovat ilmaisuvoimaltaan vahva menetelmä, joka sopii erityisen hyvin toiminnan kuvaamiseen. Käyttötapausten avulla vaatimuksista voidaan tehdä kontekstiriippuvaisia, ja samalla eri käyttäjäryhmien toimintaa voidaan eritellä. Käyttötapaukset ovat myös ymmärrettävyydeltään hyviä, joten validointi käyttäjien kanssa on toteutettavissa.

UML:n kuvausmenetelmät eivät kuitenkaan ole formatiivisia menetelmiä, joita pitäisi monimutkaisten järjestelmien vaatimusten mallintamisessa käyttää. Formatiivisuuden vaatimuksen täyttää CWA:n ADS-kaaviot, joiden vahvuus on nimenomaan tässä ominaisuudessa. ADS on myös ekologinen tapa lähestyä monimutkaisen järjestelmän vaatimuksia, sillä menetelmällä kuvataan järjestelmän kohdetta. ADS on kuitenkin melko työläs menetelmä. Se vaatii kohdealueen asiantuntemusta, jota vain kokeneet käyttäjät pystyvät tuomaan suunnitteluprosessiin.

6.1.3 Kolmas tutkimuskysymys

Kolmas tutkimuskysymys pohtii tarkemmin Kim Vicenten (1999) kehittämän Kognitiivisen työn analyysin (CWA) mallinnustapaa. CWA on työn analyysimenetelmän, jonka tähtäimenä on suunniteltavan järjestelmän formatiivinen ja ekologinen mallintaminen. Kognitiivisen työn analyysin teoriaa on käsitelty tämän julkaisun toisessa luvussa, ja analyysiin liittyviä mallinnustekniikoita neljännessä ja viidennessä luvussa.

Kolmas tutkimuskysymys:

Millaisia vaatimuksia CWA:n mallinnustekniikat nostavat esiin?

CWA:n ADS:llä voidaan mallintaa käyttäjän päätöksentekoprosessia. ADS on taulukko, johon on kuvattu järjestelmän hierarkkinen hajotelma sen osien ja tarkoitusten suhteen. Käyttäjätutkimuksissa tehtyjen havaintojen perusteella taulukkoon mallinnetaan käyttäjien ajatusten kulkua erillisinä päätöksentekoketjuina.

ADS ei ole toimintalähtöinen tapa mallintaa järjestelmiä. ADS on malli monimutkaisesta kohteesta, jota hallitaan monimutkaisen järjestelmän avulla. Toisin sanoen se ei kuvaa hallintaan liittyviä tehtäviä tai toimintoja, vaan ainoastaan itse toiminnan kohdetta.

Monimutkaisen järjestelmän suunnitteluvaiheen vaatimukset, joita esitetään ADS:llä, koskevat käyttöliittymän informaatioisisältöä. Huolellisesti toteutetun ADS:n avulla voidaan tehdä päätelmiä siitä, millaista järjestelmätietoa käyttäjä tarvitsee päätöksenteossaan. ADS auttaa informaation abstraktiotasojen hahmottelemisessa, sillä se konkretisoi käyttäjän tavoitteiden hierarkian sekä keinot, joilla tavoitteita voidaan täyttää. ADS-tilukon solut kuvaavat hallittavaan kohteeseen liittyvän tiedon eri muotoja ja hierarkkisia tasoja.

Nykyaikana monimutkaisilla järjestelmillä hallittavista prosesseista on kerättävissä kehittyneen analysointitekniikan johdosta valtava määrä erilaista tietoa. Tätä kaikkea tietoa ei missään tapauksessa ole järkevä tuoda prosessin operaattorin päätteelle. ADS:n merkitys on siinä, että sen avulla pystytään mallintamaan, mikä prosessia koskeva tieto on operaattorin päätöksenteon kannalta merkityksellistä. Tällä tiedolla ei ole merkitystä ainoastaan järjestelmän käyttöliittymän suunnittelussa, vaan jo aikaisemminkin, jolloin esimerkiksi pohditaan, minkälaista tietoa prosessista ylipäätään on tarpeen kerätä. ADS-mallit tukevat siis käyttäjäkeskeistä suunnitteluprosessia sen monessa eri vaiheessa.

Kun ADS:ään lisätään päätöksentekoketju, voidaan malliin tuoda myös tilannesidonnaisuutta. Päätöksentekoketjun avulla esitetään, minkä abstraktiotason tietoa kohteen tilasta käyttäjä tarvitsee. ADS:n avulla on siis mahdollista mal-

lintaa järjestelmän eri tilanteissa tuottamaa informaatiota ja tätä voidaan käyttää hyväksi niin järjestelmän kuin sen käyttöliittymänkin suunnittelussa.

ADS-kaavio ja siihen mallinnettu päätöksentekoketju määrittävät vaatimuksia monimutkaista kohdetta hallitsevan järjestelmän käyttöliittymälle. Niillä ei kuitenkaan ole mahdollista suoraan mallintaa järjestelmän ominaisuuksiin ja toimintoihin liittyviä vaatimuksia. Tässä suhteessa ADS-mallia on siis täydennettävä muilla kuvaustavoilla, jotka mahdollistavat dynaamisemman ja esimerkiksi erilaisia käyttäjärooleja paremmin tukevan lähestymistavan.

ADS-mallit ovat ymmärrettävyydeltään ja ilmaisuvoimaltaan riittäviä, jotta niitä voidaan käyttää käyttäjälähtöisen suunnittelun välineenä. Vaatimuksia validoitaessa kuvaustapa on kuitenkin selitettävä käyttäjille hyvin, sillä se on lähestymistavaltaan melko teoreettinen.

6.2 Monimutkaisen järjestelmän suunnitteluprosessin käyttäjäkeskeisyys

Monimutkaisen järjestelmän suunnittelu on aina monimutkainen toimenpide. Koska laajan kokonaisuuden hallinta sisältää paljon erilaista problematiikkaa, ei käyttäjäkeskeisten menetelmien käyttö suunnitteluprosessissa ole aivan yksinkertaista. Käyttäjäkeskeinen suunnittelu on kuitenkin paljon enemmän kuin järjestelmän käyttöliittymän suunnittelu tai projektin alussa tehtävä käytettävyyssuunnittelun vaatimusten määrittely. Monimutkaisen järjestelmän käyttäjäkeskeisessä suunnittelussa pyritään tuottamaan järjestelmä, joka tukee käyttäjän tehtäviä kokonaistavoitteiden täyttymisen näkökulmasta. Tässä otetaan huomioon myös työn tehokkuus- ja tuottavuusnäkökulmat.

Kun suunnitteluprosessin alussa luodaan kokonaiskuva toteutettavasta projektista, on rutiinimaiseen työn etenemiseen sisällytettävä käyttäjäkeskeisten menetelmien käyttö. Näitä menetelmiä on olemassa runsaasti, joten niiden tarkoituksenmukainen käyttö on määriteltävä.

6.2.1 CWA monimutkaisen työn analyysina

Tässä työssä on tarkasteltu Kim Vicenten (1999) kehittämää Kognitiivista työn analyysia (CWA) käyttäjäkeskeisenä suunnittelumetodologiana. Kognitiivinen työn analyysi tarjoaa formatiivisen ja ekologisen tavan mallintaa monimutkaisen järjestelmän kohdetta. Näitä periaatteita noudattamalla pyritään toteuttamaan järjestelmä, joka tukee käyttäjien yksilöllisiä tapoja tehtävien suorittamisessa. Tämä järjestelmä on myös mukautuva ja se osaa antaa käyttäjälle juuri tarvittavan tasoista tietoa järjestelmän avulla hallittavasta prosessista.

CWA on työn analysointimetodologia, jossa on viisi vaihetta. Nämä vaiheet tähtäävät kokonaisvaltaiseen työkohteen, sen hallintaan liittyvien tavoitteiden, näiden saavuttamisstrategioiden, organisaation erityispiirteiden ja työntekijöiden osaamisen malliin. Tämä analyysi ja siihen liittyvä mallinnus on tarkoitettu tehtäväksi käyttäjätutkimuksen tapaan suunnitteluprosessin alkuvaiheessa, jolloin luoduilla malleilla voidaan esittää vaatimuksia suunniteltavan järjestelmän toiminnalle ja sen käyttäjärajapinnalle.

CWA:n mukaisessa työn analyysissa mallinnettu tieto ei mielestäni yksin riitä järjestelmän suunnittelun perustaksi. CWA ottaa kantaa järjestelmän toiminnallisuuteen ainoastaan välillisesti. Sen avulla ei suoraan voida esittää vaatimuksia sille, miten käyttäjä käyttää järjestelmää tai esimerkiksi miten kohteen hallinta-tehtävät on jaoteltu käyttäjän ja automaation välillä.

CWA ja siihen sisältyvät vaatimusten kuvaustekniikat ovat kuitenkin merkittävä tapa ymmärtää ja siten mallintaa monimutkaista kokonaisuutta. Mielestäni CWA:n suurin anti on mallintamisen formatiivisuuden käsitteen soveltaminen monimutkaisen sosioteknisen järjestelmän suunnitteluun. Samoin CWA:n ADS-mallien avulla määriteltävissä oleva informaation eri abstraktiotasojen erittely on varsinkin nykymailman informaatiotulvassa merkittävää, sillä sen avulla voidaan määrittellä erilaisen informaation tilannesidonnaisia tarpeita.

6.2.2 Käyttäjätutkimuksen näkökulmat

Käyttäjäkeskeisten suunnittelumenetelmien käytöllä pyritään järjestelmien käytettävyyden parantamiseen. Käytettävyys on järjestelmän ominaisuus, jota voi-

daan kehittää toteuttamalla suunnittelu käyttäjakeskeisellä prosessimallilla, ja jota voidaan mitata käytettävyydestillä. Käytettävyys kuitenkin ilmenee ihmisten, eli käyttäjien tarkoituksenmukaisena toimintana. Käytettävyys on siis järjestelmän ominaisuus, mutta ilman järjestelmään liittyvää käyttäjien toimintaa on vaikea määritellä käytettävyyden tasoa. Käytettävyys ei siis ole itseisarvo vaan se määritellään esimerkiksi toiminnan hyödyllisyyden, tehokkuuden ja miellyttävyyden kautta.

Toiminnan tutkimus käyttäjätutkimuksen näkökulmana on erityisen merkittävä monimutkaisilla järjestelmillä, sillä niiden käytettävyysominaisuuksien tutkimisen luonnollinen lähestymistapa on käyttäjien toiminnan havainnointi. Tällöin ei tutkita niinkään yksittäisiä käyttäjiä ja heidän persoonallisia ominaisuuksiaan, vaan järjestelmän käyttöön liittyvää toimintaa ja sitä, mikä tilannesidonnaisesti on hyvää ja mikä ei niin hyvää toimintaa. Tärkeää on havaita, milloin toiminta palvelee järjestelmän kokonaistavoitteita, sillä tämän voidaan katsoa olevan yksi hyvän toiminnan kriteereistä.

Käyttäjätutkimuksen tarkoitus on selvittää, mitä ollaan suunnittelemassa ja millainen lopputuotteen tulisi olla. Tutkimuksessa tuotetaan usein tuloksena käyttäjien tarpeita, jotka siten suunniteltavalla järjestelmällä pyritään täyttämään. Monimutkaisten järjestelmien käyttäjien tarpeet on ekologisen suunnittelun periaatteiden mukaan määriteltävä järjestelmän kokonaistavoitteiden näkökulmasta. Tämäkin tukee ajatusta siitä, että käyttäjätutkimuksissa tulisi tällöin tutkia niidenomaan käyttäjien ja käyttäjäryhmien toimintaa.

6.2.3 Vaatimusten määrittely

Monimutkaisen järjestelmän vaatimusten määrittely on monimutkainen prosessi, jonka problematiikkaa voidaan kontrolloida toteuttamalla prosessi hallitusti ja suunnitelmallisesti. Prosessissa käyttäjien toimintaa tutkimalla muodostetut käyttäjien tarpeet jalostetaan järjestelmävaatimuksiksi.

Vaatimusten hallintaa voidaan helpottaa niiden tarkoituksenmukaisella luokittelulla. Monimutkainen järjestelmä on tyypillisesti jonkin prosessin ohjausjärjestelmä, jolloin vaatimusten luokittelussa voisi käyttää hyväksi Paunosen (1997) erittelemiä informaatiojärjestelmän rooleja. Eri rooleissa informaatiojärjestel-

mällä on ainakin alemmalla tasolla eri tavoitteet, vaikka kokonaistavoite pysyisi aina samana. Tällöin esimerkiksi CWA:n ADS-kaavioita olisi syytä rakentaa järjestelmän eri rooleja silmällä pitäen.

Vaatimusten määrittelyn tärkeä osa on niiden kuvaaminen. Oikeaoppinen vaatimus kertoo *mitä* suunniteltavalta tuotteelta halutaan, mutta ei sitä, *miten* se käytännössä toteutetaan. Tässä on oltava suhteellisen tarkkana joidenkin vaatimusten kuvausmenetelmien kanssa, sillä ne lähestyvät helposti suunnitteluratkaisuja kaikessa monipuolisuudessaan. Esimerkiksi UML:n monet kuvaustavat voidaan lukea helposti järjestelmän toteutuksen spesifiointiin kuuluviksi, sillä ne kuvaavat järjestelmän osien toimintaa erittäin yksityiskohtaisesti. Toisaalta, jos UML-malleissa pysytään käsitetasolla, eikä aleta eritellä tekniseen toteutukseen liittyvien osien merkityksiä, voidaan sanoa, että ollaan vielä vaatimusten määrittelyyn kuuluvan toiminnan piirissä.

6.3 Tutkimuksen puutteet

Tutkimus muodostui osin hyvin teoriapainotteiseksi. Ensimmäisen ja toisen tutkimuskysymyksen vastaukset on saatu kirjallisuustutkimuksen perusteella melko teoreettisena pohdiskeluna. Toki käytännön osuudessa toteutettiin käyttäjätutkimus, mutta tähän ei kuitenkaan analyysissa paneuduttu kovin syvästi, sillä kokeellisen osan paino oli kolmannen tutkimuskysymyksen ympärillä. Ensimmäisen ja toisen tutkimuskysymyksen vastausten pätevyyttä tulisi tutkia lisää empiirisesti, jotta niiden käytännön arvo saataisiin selville.

Tutkimuksen kokeellista osaa olisi suopeiden käytännön olosuhteiden vallitessa voinut laajentaa. Jos Suomenlahden SRS:n toteutusprojekti olisi ollut sopivassa vaiheessa, olisi muodostettuja vaatimuksia pitänyt ehdottomasti käyttää myös järjestelmän suunnittelijalla validointitarkoituksessa. Ilman tätä tutkimusvaihetta on epäselvää, voisiko kehitettyjä vaatimuksia käyttää hyväksi oikeassa suunnitteluprosessissa. Muutenkin suunnitteluprosessin nykytilan tunteminen olisi ollut hyödyllistä, jotta olisi voitu tehdä vertailuja tällä hetkellä käytössä olevien ja esiteltyjen vaatimusten kuvaustekniikoiden välillä.

6.4 Jatkotutkimusmahdollisuuksia

Ensimmäisen ja toisen tutkimuskysymyksen vastauksiin voisi jatkotutkimusmielessä suhtautua hypoteeseina ja tutkia empiirisesti niiden paikkansa pitävyyttä. Tässä tulisi kokeilla sitä, miten monimutkaisuuden erilaiset piirteet nousevat esiin erilaisilla käyttäjien toiminnan tutkimuksen menetelmillä ja miten näitä piirteitä pystytään parhaiten kuvaamaan erilaisilla mallinnustekniikoilla.

Tässä tutkimuksessa on esitetty monimutkaisen järjestelmän käyttäjakeskeisen vaatimusmäärittelyn menetelmiä. Nämä on esitetty yksittäisinä, johonkin tiettyyn suunnittelukohteen monimutkaisuudesta kumpuavaan ongelmaan vastaavina ratkaisuina. Menetelmien nivominen yhteen jonkinlaiseksi käyttäjävaatimusten kartoituksen ja esittelyn menetelmäviitekehikseksi muodostaa uuden laajemman tutkimuskysymyksen. Tämän työn perusteella voidaan sanoa, että tämänkaltaisen metodologisen kehikon kehittäminen voisi olla kiinnostavaa. Menetelmäkehikon avulla jokaista erillistä menetelmää voitaisiin käyttää juuri siinä vaatimusten määrittelyn vaiheessa, johon se parhaiten sopii.

Monimutkaisen järjestelmän käyttäjakeskeisessä suunnittelussa käyttäjien ominaisuudet on otettava huomioon jokaisessa suunnitteluprosessin erillisessä vaiheessa. Esimerkiksi monimutkaisen järjestelmän automaatioasteen määrittelyn tulisi osaksi perustua käyttäjävaatimuksille, jolloin valintoja ei perustettaisi sille, mitä voidaan automatisoida, vaan sille, mikä on tarpeen automatisoida. Toisaalta olisi tässä otettava huomioon myös, mitä tietoa prosessista on tarpeen kerätä järkevän operoinnin mahdollistamiseksi. Tästä muodostuu tutkimuskysymys menetelmästä, jota voidaan käyttää tämänkaltaisten valintojen tekemisessä.

Monimutkaisen järjestelmän käytettävyys on hyvin laaja käsite, ja sen sisällön määrittelemine perusteellisesti on ehdottomasti mielekästä. Tällä tarkoitan ajatusta siitä, että lopputuloksen käytettävyys ilmenee monimutkaisen toiminnan onnistuneisuutena, jota voidaan mitata ainoastaan korkean tason tavoitteiden täytymisen kautta. Tällöin esimerkiksi monimutkaisen järjestelmän käytettävyyden arvioinnissa on keskitettävä huomio erilaisiin asioihin, kuin jonkin yksinkertaisemman järjestelmän tapauksessa. Tämäntapainen teoreettinen näkökulma monimutkaisen järjestelmän käytettävyyteen on yksi tutkimuksen aikana nousseista käsitteellisistä tutkimusongelmista.

Lähdeluettelo

- Anttila, P. 1998. Tutkimisen taito ja tiedon hankinta. Taito-, taide- ja muotoilu-alojen tutkimuksen työvälaineet. 2. p. Helsinki: Akatiimi.
- Anttila, S. 1999. Etnografinen ilmiön kuvaus. [Viitattu 20.10.2002]. Saatavilla on-line: http://www.metodix.com/metodi/pirkko/etnografinen_ilmion_kuvaus.htm
- Beyer, H. & Holtzblatt, K. 1995. Apprenticing with the customer. *Communications of the ACM*, 38, 5, s. 45–52.
- Beyer, H. & Holtzblatt, K. 1998. *Contextual Design: Defining Customer-Centered Systems*. San Francisco: Morgan Kaufmann.
- Coughlan, J. & Macredie, R. D. 2002. Effective communication in requirements elicitation: A comparison of methodologies. *Requirements Engineering*, 7, s. 47–60.
- Douglas, B. P. 2001. Capturing Real-Time requirements. *Embedded Systems Programming*, November 2001.
- Ede, M. R. 1998. Focus Groups to Study Work Practice. *Usability Interface*, Vol 5, No. 2, October 1998. Saatavissa on-line: <http://www.stcsig.org/usability/newsletter/9810-focusgroups.html>
- Faulkner, X. 2000. *Usability Engineering*. New York: Palgrave.
- Fowler, M. 1997. *UML Distilled: Applying the Standard Object Modeling Language*. Reading, Massachusetts: Addison-Wesley.
- Hackos, J. & Redish, J. 1998. *User and Task Analysis for Interface Design*. New York: John Wiley & Sons.
- Haikala, I. & Märijärvi, J. 2002. *Ohjelmistotuotanto*. Pieksämäki: Satku.
- Hajdukiewicz, J. R., Doyle, J., Milgram, P., Vicente, K. J. & Burns, C. 1998. A Work Domain Analysis of Patient Monitoring in the Operating Room. *Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting 1998*. Saatavissa on-line: <http://acad88.sahs.uth.tmc.edu/courses/hi6301/vincente 1.pdf>

Heath, C. & Luff, P. 1992. Crisis Management and Multimedia Technology in London Underground Line control Rooms. *Journal of Computer Supported Cooperative Work (CSCW)* Vol. 1, No 1, s. 24–48.

Hirsjärvi, S., Remes, P. & Sajavaara, P. 1997. *Tutki ja kirjoita*. Helsinki: Kustannusosakeyhtiö Tammi.

IMO [Viitattu 2.9.2002] Vessel Traffic Services. Saatavissa on-line: http://www.imo.org/Safety/mainframe.asp?topic_id=387

ISO 11064. 2000. Ergonomic design of control centres – Part 1, Principles for the design of control centres. ISO Standard.

ISO 13407. 1999. Human-centred design processes for interactive systems. ISO Standard.

ISO 9126. 1991. Information technology – Software product evaluation – Quality characteristics and guidelines for their use. ISO Standard.

ISO 9241-11. 1998. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11, Guidance on usability. ISO Standard.

ISO/IEC 14598-1. 1999. Information technology – Software product evaluation Part 1, General overview. ISO/IEC Standard.

Johnson, M. 2002. *Towards Web-Based Requirements Management. A Case Study in a Distributed Development Organisation*. Diplomityö. Teknillinen Korkeakoulu, Tietotekniikan osasto. Espoo. 95 s.

Johnson, P., O’Neill, E. & Johnson, H. 1999. Introduction to This Special Issue on Representations in Interactive Systems Development. *Human–Computer Interaction*, Volume 14, s. 1–7. Lawrence Erlbaum Associates, Inc.

Jokela, T. 2001. Assessment of user-centred design process as a basis for improvement action – An experimental study in industrial settings. Department of Information Processing Science, University of Oulu.

Kallela, J. 1996. Automaation paradigmat, Käyttäjien osallistuminen automaation suunnitteluun. Espoo: Valtion teknillinen tutkimuskeskus, VTT Julkaisuja – Publikationer 817. 148 s. + liitt. 74 s.

Keinonen, T. 2000. Yksiulotteinen käytettävyys. Teoksessa: Keinonen, T. (toim.). Miten käytettävyys muotoillaan? Helsinki: Taideteollinen korkeakoulu. S. 92–105. (Taik:n julkaisu B61.)

Kreitzberg, C. 1999. The LUCID Design Framework. Cognetics Corporation. Saatavissa on-line: <http://www.cognetics.com/lucid/>.

Kujala, S. 2002. User Studies: A Practical Approach to User Involvement for Gathering User Needs and Requirements. Acta Polytechnica Scandinavica, Mathematics and Computing Series No. 116. Espoo: the Finnish Academies of Technology.

Kuutti, K. 2000. Käyttöliittymä- ja käytettävyystutkimuksen haasteet. Teoksessa: Keinonen, T. (toim.). Miten käytettävyys muotoillaan? Helsinki: Taideteollinen korkeakoulu. S. 79–91. (Taik:n julkaisu B61.)

Lee, G., Howard, J. & Anderson P. 2002. Safety-Critical Requirements Specification and Analysis Using SpecTRM. Teoksessa: Proceedings of the 2nd Meeting of the US Software System Safety Working Group, February 2002. Saatavissa on-line: <http://www.safeware-eng.com/pubs/sswg2002.pdf>

Leppänen, A. & Norros, L. 2002. Teollisten prosessien inhimilliset käyttövarmuustekijät. Työ ja ihminen 16(2002)2, s. 105–118.

Leveson, N., de Villepin, M., Daouk, M., Bellingham, J., Srinivasan, J., Neogi, N., Bachelor, E., Pilon, N. & Flynn, G. 2001. A Safety and Human-Centered Approach to Developing New Air Traffic Management Tools. Teoksessa: Proceedings of the ATM 2001, Dec 2001. Saatavissa on-line: <http://sunnyday.mit.edu/papers/atm2001.pdf>

Lignell, L., Kytösalmi, M., Haho, P. & Smeds, R. 2003. Interactive process development through process simulations to identify the requirements of information system. The 7th International Workshop of the IFIP WG 5.7

Special Interest Group on Experimental Interactive Learning in Industrial Management, Aalborg, Denmark, 22–24 May, 2003. S. 89–99.

Saatavissa on-line: <http://www.simlab.hut.fi/publications/lignell.pdf>

Lind, M. 1999. Making Sense of the Abstraction Hierarchy [esitelmä]. CSAPC'99 (Cognitive Science Approaches to Process Control), Villeneuve d'Ascq, Ranska. 21.–24.9.1999.

Saatavissa on-line: <http://www.iau.dtu.dk/~ml/csapc99.pdf>

Merenkulkulaitos. [viitattu 5.11.2002]. Liikenteenohjaus (VTS). Saatavissa on-line: <http://www.fma.fi/toiminnot/liikenteenohjaus/>

Miller, C. 2000. The Human Factor in Complexity. Teoksessa: Samad, T. & Weyrauch, J. (toim.). Automation, Control and Complexity. West Sussex, Englanti: John Wiley & Sons Ltd. S. 35–57.

Nielsen, J. 1993. Usability Engineering. San Diego: Academic Press, Inc.

Norman, D. 1988. The Psychology of Everyday Things. New York: Basic Books.

OMG 2002 [viitattu 4.3.2003] UML Profile for Schedulability, Performance, and Time Specification.

Saatavissa on-line: <http://www.omg.org/docs/ptc/02-03-02.pdf>

Oxford Advanced Learner's Dictionary 1995. Oxford: Oxford University Press.

Parnas, D. L. 2000. Requirements Documentation: Why a Formal Basis is Essential. Fourth IEEE International Conference on Requirements Engineering (ICRE 2000) Keynote presentation II.

Saatavissa on-line: <http://www.cse.msu.edu/ICRE2000/parnas.pdf>

Paunonen, H. 1997. Roles of Informing Process Control Systems. Tampere: Tampereen teknillinen korkeakoulu. 166 s.

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T. 1994. Human-Computer Interaction. Harlow, England: Addison Wesley.

Savioja, P. 2003. Käyttäjäkeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa. Diplomityö. Automaatio- ja systeemitekniikan osasto. Teknillinen korkeakoulu.

Sawyer, P. & Kotonya, G. 2001. Software Requirements. Teoksessa: SWEBOK – Guide to Software Engineering Body of Knowledge, Trial Version. Saatavissa on-line: <http://www.swebok.org/stoneman/version09.html>

Schach, S. 2002. Object-Oriented and Classical Software Engineering. New York: McGraw-Hill.

Smith, J. A. 1995. Semi-Structured Interviewing and qualitative analysis. Teoksessa: Smith, J. A., Harre, R. & Langenhove, L. V. (toim). Rethinking Methods in Psychology. London: Sage Publications.

Sutcliffe, A. 2002. User-Centred Requirements Engineering. London: Springer Verlag.

Vicente, K. J. 1999. Cognitive Work Analysis. Toward Safe, Productive, and Healthy Computer-Based Work. Mahwah, New Jersey: Lawrence Erlbaum.

Vredenburg, K., Mao, J.-Y., Smith, P. W. & Carey, T. 2002. A survey of user-centered design practice. Proceedings of CHI 2002. New York: ACM. S. 471–478.

VTT 2002. Suomenlahden alusliikenteen ohjaus- ja informaatiojärjestelmän esisimulointi. Tutkimusselostus Nro TUO34-021653.

Wiersma, E. & Heijer, T. 1996. Human Factors Models and Methodologies for Vessel Traffic Service Operator Performance. Delft University of Technology. 16 s.

Woods, D. 1988. Coping with complexity: the psychology of human behaviour in complex systems. Teoksessa: Goodstein, L. P., Andersen, H. B. & Olsen, S. E. (toim.). Tasks, Errors and Mental Models. London: Taylor & Francis.

Zuboff, S. 1988. In the Age of the Smart Machine. The Future of Work and Power. New York: Basic Book Inc.

Liite A: Standardin ISO 13407 periaatteet

Standardi ISO 13407 (1999) toteaa, että käyttäjäkeskeiselle suunnittelulle luonteenomaisia ovat seuraavat prosessia koskevat yleiset kuvaukset:

- **Käyttäjien aktiivinen osallistuminen suunnitteluun, sekä suunnittelijoiden selkeä ymmärrys käyttäjä- ja tehtävävaatimuksista:** Käyttäjien osallistuttamisella saadaan tieto heidän ympäristöstään, tehtävistään ja työtavoistaan sisällytettyä järjestelmän suunnitteluun. Suunniteltaessa järjestelmää tietyille, ennalta määritellyille käyttäjille, lisää heidän osallistumisen suunnitteluun uuden järjestelmän käyttöönoton helppoutta sekä vähentää yleistä muutosvastarintaa. Kuluttajatuotteiden suunnittelussa osallistuttamalla sopivia käyttäjiä suunnitteluun voidaan määritellä uuden tuotteen yleiset vaatimukset.
- **Tarkoituksenmukainen tehtävien jaottelu ihmisen ja teknologian välillä:** Nämä suunnitteluratkaisut määrittelevät kuinka suuri osa nykyisistä järjestelmään liittyvistä toiminnoista on uudessa järjestelmässä automatisoitava. Ratkaisujen tulisi perustua ihmisen ja tekniikan erilaisiin esimerkiksi luotettavuutta, nopeutta, tarkkuutta, joustavuutta koskeviin kykyihin ja rajoitteisiin.
- **Suunnitteluratkaisujen iterointi:** Iteratiivisessa lähestymistavassa loppukäyttäjiltä suunnitteluvaiheessa saatu palaute toimii koko suunnitteluprosessin ajan kriittisenä tiedonlähteenä. Iteroinnilla voidaan minimoida se riski, että valmis järjestelmä ei toteuttaisikaan suunnittelun alussa asetettuja käyttäjä- ja organisatorisia vaatimuksia. Myös joidenkin eksplisiittisesti vaikeasti ilmaistavissa olevien vaatimusten todentaminen on iteratiivisen suunnittelun avulla mahdollista. Samalla mahdollistuu myös erilaisten suunnitteluratkaisujen kokeilu ennen varsinaisen toteutettavan ratkaisun valintaa.

Liite B: Standardin ISO 11064 periaatteet

ISO 11064 rakentuu yhdeksän eri pääperiaatteen ympärille. Nämä ovat:

1. Ihmiskeskeisen suunnittelun lähestymistapa
2. Ergonomian integrointi insinööritoimintaan
3. Suunnittelun parantaminen iteroimalla
4. Tilanneanalyysin tekeminen
5. Tehtäväanalyysin tekeminen
6. Virheitä sietävän järjestelmän suunnittelu
7. Käyttäjien osallistumisen varmistaminen
8. Monitieteisen suunnittelutiimin muodostaminen
9. Ergonomiatekijöiden dokumentoiminen

Liite C: ISO 11064 -prosessin vaiheet

A: Selvitys

1 Tavoitteiden ja taustavaatimusten selventäminen

B: Analyysi ja määrittely

2 Järjestelmän suoritusasteen määrittely

Ihmisten ominaisuudet ja vaatimukset

3 Tehtävien allokointi ihmiselle ja koneelle

Järjestelmän toimintojen ja vaatimusten määrittely

4 Tehtävävaatimusten määrittely

5 Töiden ja tehtävien organisoimisen suunnittelu

simulointi

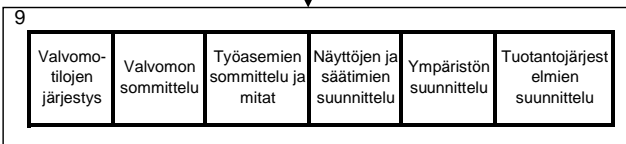
6 Saatujen tulosten todennus ja kelpuus

C: Käsitteellinen suunnittelu

7 Suunniteltavan keskuksen käsitteellinen mallinnus

8 Käsitteellisen suunnittelun hyväksyminen

D: Yksityiskohtainen suunnittelu



simulointi

10 Detaljisuunnittelun todennus ja kelpuus

E: Operatiivinen palaute

11 Käyttökokemusten kerääminen

Toiseen projektiin soveltaminen

Liite D: Alustavat vaatimukset

Yleinen ADS-malli SRS-järjestelmästä.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät	Aluksen ohjausjärjestelmä	Fyysiset osat ohjausjärjestelmässä
Ympäristön ja ihmishenkien suojele, meriliikenneturvallisuus					
Alusten turvallinen navigointi					
Liikennetilannetiedon välittyminen osapuolille					
Kurssit, nopeudet, sää, ympäristö					

Tilannesidonnainen ADS-malli aluksen saapumisesta SRS-alueelle.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät	Aluksen ohjausjärjestelmä	Fyysiset osat ohjausjärjestelmässä
Ympäristön ja ihmishenkien suojele, meriliikenneturvallisuus	Alus ilmoittaa saapuvansa SRS-alueelle Monitorointi				
Alusten turvallinen navigointi		"Reitti selvä"			
Liikennetilannetiedon välittyminen osapuolille	Alus näkyy tutkakuvassa	Alus seurataan	Ennakkoreportti annettu?		
Kurssit, nopeudet, sää, ympäristö		"Onko kaikki kunnossa?"			

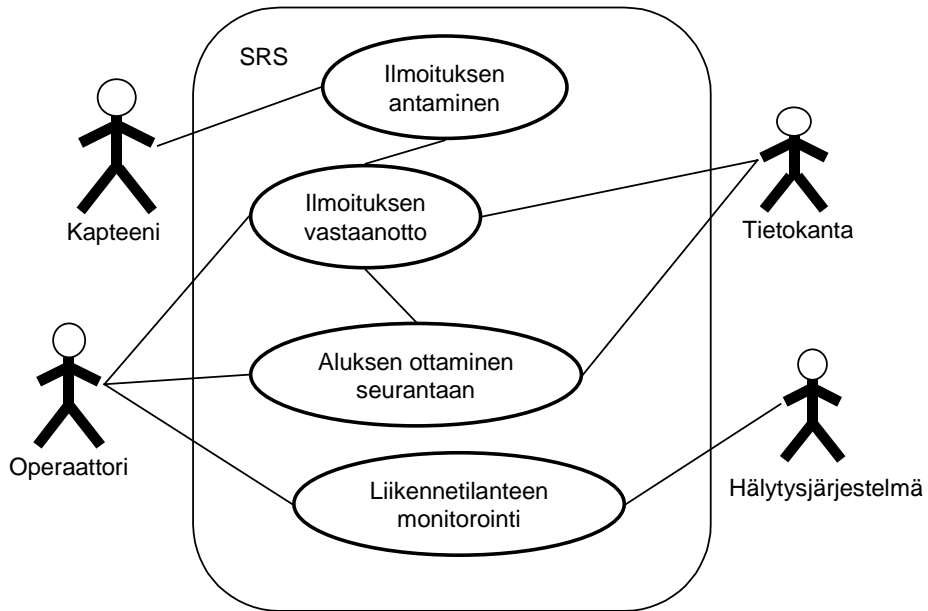
Tilannesidonnainen ADS-malli tunnistamattoman aluksen saapumisesta SRS-alueelle.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät	Aluksen ohjausjärjestelmä	Fyysiset osat ohjausjärjestelmässä
Ympäristön ja ihmisten suojeleminen, meriliikenneturvallisuus		Alueen rajaylittämässä tunnistamaton alus	Nimen tiedustelu VHF-puhelimen avulla, aluksen kutsuminen paikkatiedon perusteella		
Alusten turvallinen navigointi					
Liikennetilanteiden välittäminen osapuolille	Soitto toiseen SRS-keskukseen ja aluksen tietojen tiedustelu	Aluksen kutsunimellä VHF-puhelinta käyttäen	Saapumisilmoituksen anto SRS-keskukselle		
Kurssit, nopeudet, sää, ympäristö					

Tilannesidonnainen ADS-malli kahden aluksen törmäysuhan välttämisestä.

	Usean liikenteessä olevan aluksen sekä ympäristön muodostama kokonaisuus	Yksittäinen alus ja sen ympäristö	Aluksen komentosillan järjestelmät	Aluksen ohjausjärjestelmä	Fyysiset osat ohjausjärjestelmässä
Ympäristön ja ihmisten suojeleminen, meriliikenneturvallisuus	Tutkakuva: kaksi alustaristeä välillä kurssilla		Aluksen kurssia muuttettava		
Alusten turvallinen navigointi		Alusten nopeudet		Kurssin muuttaminen	
Liikennetilantiedon välittyminen osapuolille			Yhteys aluksiin: Tilanne huomioitu?		
Kurssit, sää, ympäristö	Onko liikennekuva oikea?				Kurssin muutos

Käyttötapausmalli aluksen saapumisesta SRS-alueelle.



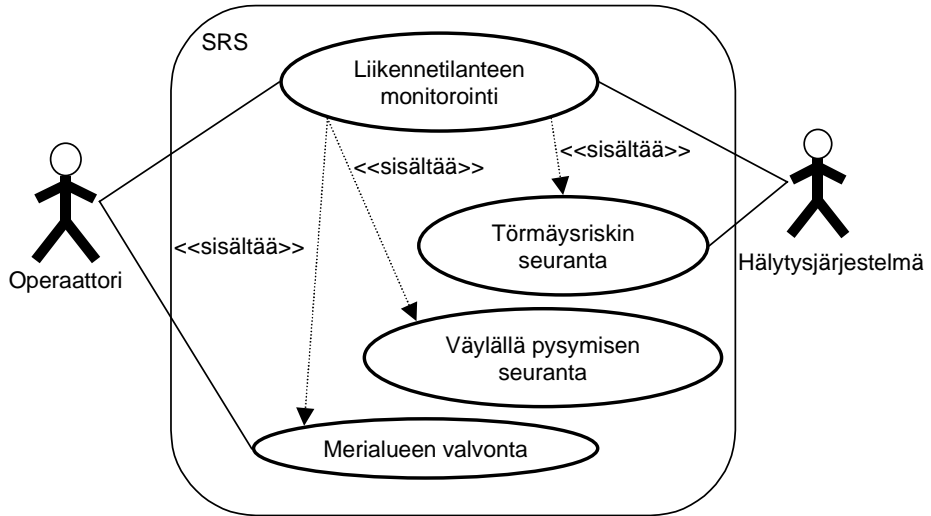
Yllä olevaan malliin liittyvät skenaariot.

Nimi:	Ilmoituksen antaminen
Suorittajat:	Aluksen kapteeni
Alkuehdot:	Alus ylittämässä SRS-alueen rajan, ennakoilmoitus annettu
Kuvaus:	Aluksen kapteeni kutsuu VHF-puhelimen kutsukanavalla SRS-keskusta. Operaattori vastaa. Molemmat siirtyvät VHF:n työkanavalle. Kapteeni antaa lyhyen alueelle saapumisilmoituksen. Operaattori kiittää ja kehottaa palaamaan kutsukanavalle.
Poikkeukset:	Ennakoilmoitusta ei ole annettu. Tällöin kapteeni antaa koko ilmoituksen VHF:n työkanavalla.
Loppuehdot:	Operaattori ottanut vastaan ilmoituksen ja tunnistanut aluksen tutkakuvaltaan.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

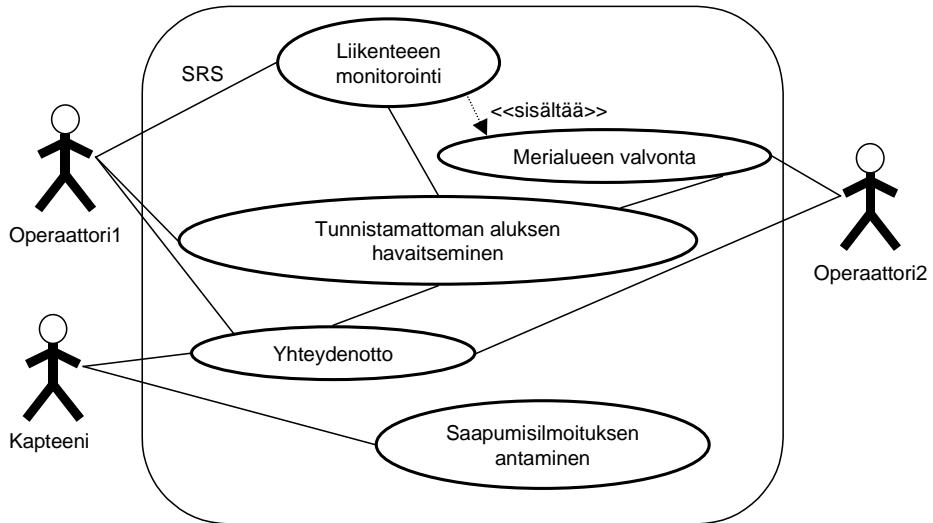
Nimi:	Ilmoituksen vastaanotto
Suorittajat:	SRS-keskuksen operaattori, tietokanta
Alkuehdot:	Alus ylittämässä SRS-alueen rajan, ennakoilmoitus annettu
Kuvaus:	Operaattori vastaa aluksen kapteenin kutsuun VHF-puhelimen kutsukanavalla. Molemmat siirtyvät VHF:n työkanavalle. Operaattori vastaanottaa lyhyen alueelle saapumisilmoituksen ja päivittää tietokantaansa. Operaattori kiittää ja kehottaa alusta palaamaan kutsukanavalle.
Poikkeukset:	Ennakoilmoitusta ei ole annettu. Operaattori vastaanottaa pitkän raportin ja päivittää samalla tietokantaansa kaikki tiedot.
Loppuehdot:	Operaattori ottanut vastaan ilmoituksen ja tunnistanut aluksen tutkakuvaltaan.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Nimi:	Aluksen ottaminen seurantaan
Suorittajat:	SRS-keskuksen operaattori, tietokanta
Alkuehdot:	Alus tunnistettu, ilmoitukset annettu
Kuvaus:	Operaattori merkitsee tutkakuvaansa ilmoituksen antaneen aluksen tiedot. Muiden operaattoreiden tutkakuvat päivittyvät samalla.
Poikkeukset:	–
Loppuehdot:	Aluksen tiedot näkyvät tutkakuvassa.
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Käyttötapausmalli SRS-operaattorin toiminnasta yleisellä tasolla.



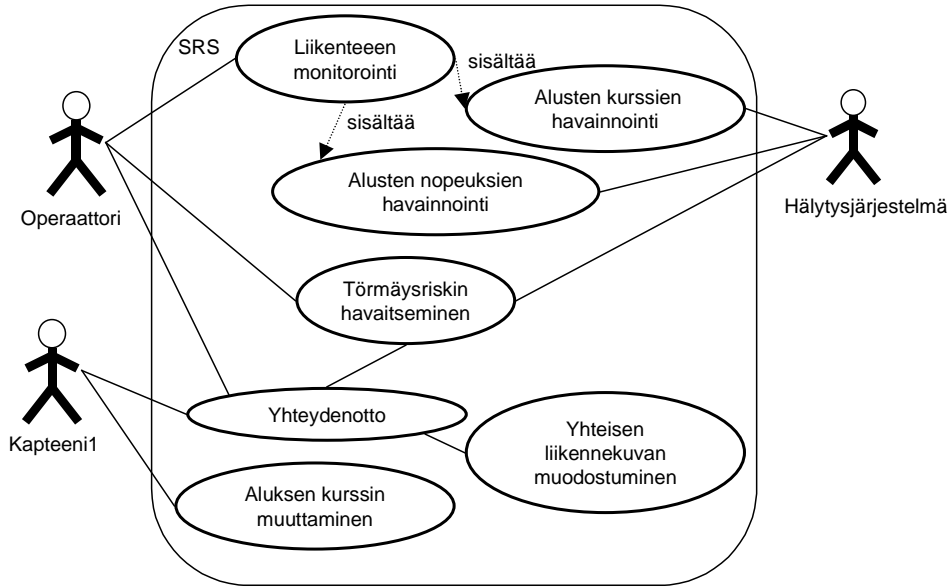
Käyttötapausmalli tunnistamattoman aluksen saapumisesta SRS-alueelle.



Skenaario tunnistamattoman aluksen saapumisesta SRS-alueelle.

Nimi:	Tunnistamattoman aluksen havaitseminen
Suorittajat:	SRS-keskuksen operaattori, aluksen kapteeni, toisen keskuksen operaattori
Alkuehdot:	Alus saapumassa SRS-alueelle toiselta valvotulta alueelta
Kuvaus:	Operaattori havaitsee aluksen olevan saapumassa SRS-alueelle. Kyseisen aluksen tietoja ei näy operaattorin tutkakuvassa, joten se ei ole seurannassa, eikä siten antanut saapumisilmoitusta. Operaattori kutsuu alusta sen paikkakoordinaattien mukaan VHF-puhelimen kutsukanavalla. Alus huomaa kutsun ja vastaa siihen. Molemmat siirtyvät työkanavalle ja alus antaa saapumisilmoituksen.
Poikkeukset:	Alus ei huomaa kutsua kutsukanavalla. Operaattori ottaa yhteyttä toiseen keskukseen ja kysyy aluksen tietoja. Toinen keskus antaa aluksen nimen. Operaattori kutsuu alusta nimellä kutsukanavalla. Alus vastaa kutsuun. Molemmat siirtyvät työkanavalle ja alus antaa saapumisilmoituksen.
Loppuehdot:	Alus antaa saapumisilmoituksen
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Käyttötapauskaavio kahden aluksen törmäysuhan välttämisestä.



Skenaario kahden aluksen törmäysuhan havaitsemisesta ja välttämisestä.

Nimi:	Törmäysriskin havaitseminen
Suorittajat:	SRS-keskuksen operaattori, Aluksen kapteeni
Alkuehdot:	Alukset tunnistettu, ilmoitukset annettu jne.
Kuvaus:	Operaattori havaitsee tutkakuvastaan kahden aluksen olevan lähestymässä toisiaan. Alusten nopeuksien perusteella operaattori havaitsee törmäysvaaran olevan syntymässä. Operaattori kutsuu toista aluksista ja tiedustelee onko tilanne havaittu komentosillalla. Aluksen kapteeni suorittaa tarvittavat toimenpiteet törmäysuhan välttämiseksi.
Poikkeukset:	–
Loppuehdot:	Alukset ohittavat toisensa turvallisesti
Muut vaatimukset:	Järjestelmien toiminnassa ei vikoja

Tekijä(t) Savioja, Paula			
Nimeke Käyttäjakeskeiset menetelmät monimutkaisten järjestelmien vaatimusten kuvaamisessa			
<p>Tiivistelmä</p> <p>Käyttäjakeskeisillä suunnittelumenetelmillä pyritään parantamaan suunniteltavien tuotteiden käytettävyyssominaisuuksia. Hyvä käytettävyys lisää tuotteen hyödyllisyyttä, tehokkuutta ja käyttömukavuutta. Monimutkaisten järjestelmien avulla hallitaan laajoja prosesseja, jotka ovat luonteeltaan dynaamisia, turvallisuuskriittisiä, hajautettuja ja pitkälti automatisoituja. Monimutkaisia järjestelmiä ovat esimerkiksi prosessilaitosten ohjausjärjestelmät sekä erilaiset liikenteenohjausjärjestelmät. Vaatimusten määrittely on suunnitteluprosessin vaihe, jossa tuodaan julki suunniteltun kohteen halutut ja merkitykselliset ominaisuudet. Kokonaisvaltaisen käytettävyyden kannalta käyttäjakeskeisten menetelmien käyttö on erityisen hyödyllistä vaatimusten määrittelyssä.</p> <p>Työssä esitetään melko laajasti erilaisia vaatimusten määrittelyyn liittyviä käyttäjakeskeisiä menetelmiä. Nämä on jaoteltu sen mukaan, koskevatko ne käyttäjätutkimusta vai vaatimusten kuvaamista. Menetelmien osalta on pohdittu, mitkä niistä sopivat parhaiten ns. monimutkaisten järjestelmien suunnitteluun. Sopivien menetelmien on nostettava esiin juuri niitä vaatimuksia, jotka ovat suunniteltavan kohteen monimutkaisuuden kannalta olennaisia. Syväällisimmin perehdytään kognitiiviseen työn analyysiin, jossa suunniteltavaa järjestelmää lähestytään sillä hallittavan kohteen ominaisuuksien kautta. Tällä pyritään formatiiviseen malliin järjestelmästä. Formattiivisen suunnittelun tuloksena on järjestelmä, joka tukee käyttäjien yksilöllisiä ongelmanratkaisutapoja ja adaptoituu käyttötilanteeseen tuottaen informaatiota käyttäjän päätöksenteon kannalta merkityksellisistä kohteen osista.</p> <p>Työn käytännön osuudessa vertaillaan Kognitiivisen työn analyysin ADS-malleilla sekä UML:n käyttötapauksilla esitettyjä vaatimuksia. Mallinnuskohteena on Suomenlahden alusten ilmoittautumisjärjestelmä (SRS). Käytännön osuudessa havaittiin, että mallinnustavat tuottavat täysin erityyppisiä vaatimuksia. ADS-malleilla kuvatut vaatimukset kuvaavat käyttäjän tarvitsemaa kohteesta saatavaa informaatiota, sillä ADS:lla voidaan mallintaa sitä, mikä tieto on käyttäjän kannalta merkityksellistä informaatiota. Tätä voidaan käyttää hyväksi monimutkaisten järjestelmien käyttöliittymäsuunnittelussa.</p> <p>Työssä havaittiin, että tilanteessa, jossa käyttäjakeskeisen suunnittelun kohteena on jokin monimutkainen sosiotekninen järjestelmä, vaaditaan sekä käyttäjätutkimus- että vaatimusten kuvaus -menetelmiltä erityisominaisuuksia. Molemmissa vaiheissa on pystyttävä ilmentämään mallinnuksen kohteen niitä piirteitä, jotka tekevät siitä monimutkaisten ja sosioteknisen. Näitä ovat esimerkiksi järjestelmän dynaaminen käyttäytyminen, sen toimintaan liittyvät suuret riskitekijät sekä suoritettavien tehtävien yhteistoiminnallinen luonne. Käyttäjätutkimusmenetelmistä näitä piirteitä tukevat esimerkiksi Contextual Inquiry ja erilaiset simulaatiot. Vaatimusten kuvausmenetelmistä erityisen sopivia ovat UML:n käyttötapaukset sekä kognitiivisen työn analyysin ADS-mallit.</p> <p>Tutkimuksessa havaittiin myös, että monimutkaisten järjestelmien käytettävyys on monimutkainen käsite, jonka teoreettinen määrittely vaatii syvällistä analyysia siitä, mitkä ovat järjestelmien hyvyden kriteerit. Monimutkaisten järjestelmien käytettävyys ilmenee ihmisen toiminnan tavoitteiden täyttymisenä tiettyjen reunaehtojen vallitessa.</p>			
Avainsanat Cognitive Work Analysis (CWA), user-centred design, user-centered design, requirements specification, complex systems, usability			
Toimintayksikkö VTT Tuotteet ja tuotanto, Tekniikantie 12, PL 1301, 02044 VTT			
ISBN 951-38-6186-4 (nid.) 951-38-6187-2 (URL: http://www.vtt.fi/inf/pdf/)		Projektinnumero TUO9523	
Julkaisu-aika Lokakuu 2003	Kieli Suomi, engl. tiiv.	Sivu- 132 s. + liitt. 10 s.	Hinta C
Projektin nimi		Toimeksiantaja(t)	
Avainnimeke ja ISSN VTT Tiedotteita – Research Notes 1235-0605 (nid.) 1455-0865 (URL: http://www.vtt.fi/inf/pdf/)		Myynti: VTT Tietopalvelu PL 2000, 02044 VTT Puh. (09) 456 4404 Faksi (09) 456 4374	

Author(s) Savioja, Paula			
Title User-centred methods in presenting the requirements of complex systems			
<p>Abstract</p> <p>User-centred design methods aim at improving the usability of the object being designed. High degree of usability leads to an increase in the product's effectiveness, efficiency and user satisfaction. Complex systems are used to control extensive processes, which can be characterised as dynamic, safety critical, distributed and highly automated. Examples of such systems are the various control systems in process industry and different traffic control systems. Requirement specification is a phase in a design process in which the required and significant features of the object being designed are more explicit. In achieving comprehensive usability the User-centred design methods are specifically significant in the requirements elicitation and specification phase of the design process.</p> <p>This study introduces various user-centred methods that are relevant in the requirements phase of a design process. The methods are divided into two categories according to their relevance in either gathering the requirements or presenting them. The methods are presented in order to investigate which ones of them would best suit the design of so-called complex systems. The suitable methods shall be able to present the requirements of the system being designed that are relevant relating to its complexity. In this study the most thorough analysis is done on the Cognitive Work Analysis (CWA), in which the system being designed is approached through the qualities of the domain of system. This aims at creating a formative model of the system. The result of a formative design process is a system that supports the users' unique problem solving strategies and adapts to the context of use producing relevant domain information for the users' decision-making needs.</p> <p>Requirements represented as the abstraction-decomposition space (ADS) of CWA and as UML's use cases are compared in the empirical part of this study. The domain of the modelling is the Ship reporting System (SRS) of the Gulf of Finland. The requirements produced by the two modelling techniques are of a comprehensively different type. The ADS models represent requirements for the domain information that the user needs because ADS depicts what domain information is relevant for the users. This can be used in the user interface design of complex systems.</p> <p>In this study it was discovered that in situations in which the object of user-centred design is a complex sociotechnical system there are some special requirements upon the user study and requirements representation of the design process. In both those phases the methods shall be able to represent the aspects of the domain that specifically make it a complex sociotechnical system. These are for example the dynamic behaviour of the system, the high degree of potential hazards and the social aspects of the work relating to it. The user study methods which support the eliciting of these aspects are for example Contextual Inquiry and different types of simulations. In representing the requirements suitable methods are Use Cases and Abstraction-decomposition spaces.</p> <p>In this study it was also found out the usability of complex systems is a complex concept and the definition of which requires a profound theoretical analysis on the criteria of appropriateness of a system. The usability of a complex sociotechnical system appears in the action of the users of the system under certain conditions.</p>			
<p>Keywords</p> <p>Cognitive Work Analysis (CWA), user-centred design, user-centered design, requirements specification, complex systems, usability</p>			
<p>Activity unit</p> <p>VTT Industrial Systems, Tekniikantie 12, P.O. Box 1301, FIN-02044 VTT, Finland</p>			
<p>ISBN</p> <p>951-38-6186-4 (soft back ed.) 951-38-6187-2 (URL: http://www.vtt.fi/inf/pdf/)</p>		<p>Project number</p> <p>TUO9523</p>	
<p>Date</p> <p>October 2003</p>	<p>Language</p> <p>Finnish, Engl. abstr.</p>	<p>Pages</p> <p>132 p. + app. 10 p.</p>	<p>Price</p> <p>C</p>
<p>Name of project</p>		<p>Commissioned by</p>	
<p>Series title and ISSN</p> <p>VTT Tiedotteita – Research Notes 1235-0605 (soft back ed.) 1455-0865 (URL: http://www.vtt.fi/inf/pdf/)</p>		<p>Sold by</p> <p>VTT Information Service P.O. Box 2000, FIN-02044 VTT, Finland Phone internat. +358 9 456 4404 Fax +358 9 456 4374</p>	

VTT TIEDOTTEITA – RESEARCH NOTES

VTT TUOTTEET JA TUOTANTO – VTT INDUSTRIELLA SYSTEM – VTT INDUSTRIAL SYSTEMS

- 2117 Malm, Timo, Hämäläinen, Vesa & Kivipuro, Maarit. Paperiteollisuuden rullankäsittelyn turvallisuus ja luotettavuus. 2001. 68 s. + liitt. 12 s.
- 2140 Reiman, Teemu & Oedewald, Pia. The assessment of organisational culture. A methodological study. 2002. 42 p.
- 2148 Aaltonen, Pertti, Bojinov, Martin, Helin, Mika, Kinnunen, Petri, Laitinen, Timo, Muttilainen, Erkki, Mäkelä, Kari, Reinwall, Anneli, Saario, Timo & Toivonen, Aki. Facts and views on the role of anionic impurities, crack tip chemistry and oxide films in environmentally assisted cracking. 2002. 68 p. + app. 21 p.
- 2149 Hemilä, Jukka. Information technologies for value network integration. 2002. 97 p. + app. 1 p.
- 2150 Pöyhönen, Ilpo, Kylmälä, Kaarle, Harju, Hannu, Kempainen-Kajola, Pia, Kuhakoski, Kalle, Spankie, Greig & Ventä, Olli. Vaatimukset ohjelmistoa sisältäville lääkintälaitteille. Hallinta ja menetelmät vaatimustenmukaisuuden osoittamiseksi. 2002. 135 s. + liitt. 40 s.
- 2151 Harju, Hannu. Kustannustehokas ohjelmiston luotettavuuden suunnittelu ja arviointi. Osa 1. 2002. 114 s. + liitt. 15 s.
- 2156 Rääkkönen, Timo. Riskienhallinnan kehityskaari ja vaikuttavuusarviointi. Turvallisuus- ja ympäristöriskit. 2002. 47 s. + liitt. 14 s.
- 2160 Hentinen, Markku, Hynnä, Pertti, Lahti, Tapio, Nevala, Kalervo, Vähänikkilä, Aki & Järvi-
luoma, Markku. Värähtelyn ja melun vaimennuskeinot kulkuvälineissä ja liikkuvissa
työkoneissa. Laskenta-periaatteita ja käyttöesimerkkejä. 2002. 118 s. + liitt. 164 s.
- 2171 Tonteri, Hannele, Vatanen, Saija, Lahtinen, Reima & Kuuva, Markku. Elinkaariajattelu
työkoneiden ympäristömyötäisessä suunnittelussa. 2002. 33 s.
- 2172 Tonteri, Hannele, Vatanen, Saija, Lahtinen, Reima & Kuuva, Markku. Life cycle thinking in the
design for environment aware work machines. 2003. 32 p.
- 2173 Häkkinen, Kai. Valmistuksen ja suunnittelun yhteistyö toistuvan erätuotannon alihankintapro-
sessissa; havaintoja suomalaisessa pk-konepajateollisuudessa vuonna 2002. 2002. 52 s.
- 2178 Andersson, Peter, Tamminen, Jaana & Sandström, Carl-Erik. Piston ring tribology. A literature
survey. 2002. 105 p.
- 2180 Kaunisto, Tuija. Talousvesijärjestelmien materiaalien ja tuotteiden hyväksymismenettelyt. EAS-
prosessi Suomessa. 2002. 25 s. + liitt. 4 s.
- 2184 Kondelin, Kalle & Karhela, Tommi. Gallery Markup and Query Language Specification. 2003.
111 p.
- 2190 Häkkinen, Kai, Pötry, Jyri & Joutsen, Peik. Konepajateollisuuden alihankintaprosessien kehit-
tämisedellytykset ja -tavat pk-sektorilla. Koneali-projektin loppuraportti. 2003. 129 s.
- 2193 Harju, Hannu & Koskela, Mika. Kustannustehokas ohjelmiston luotettavuuden suunnittelu ja
arviointi. Osa 2. 2003. 107 s.
- 2208 Rääkkönen, Timo & Rouhiainen, Veikko. Riskienhallinnan muutosvoimat. Kirjallisuuskatsaus.
2003. 77 p.
- 2216 Savioja, Paula. Käyttäjakeskeiset menetelmät monimutkaisten järjestelmien vaatimusten ku-
vaamisessa. 2003. 132 s. + liitt. 10 s.

Monipuolistuva teollinen ympäristö aiheuttaa monenlaisia haasteita järjestelmäsuunnittelijoille. Niin järjestelmien toiminnallisuuden kuin käyttöliittymänkin suunnittelussa on omaksuttava luovia ja innovatiivisia ratkaisumalleja. Tämä luo kasvavia tarpeita käyttäjäkeskeisen suunnittelun menetelmiä työjärjestelmien suunnittelussa. Käyttäjakeskeisillä suunnittelumenetelmillä pyritään parantamaan järjestelmien käytettävyyttä eli käytön hyödyllisyyttä, tehokkuutta ja käyttömukavuutta. Menetelmiä käytetään sekä toiminnallisuuden että käyttöliittymäominaisuuksien suunnittelussa. Julkaisussa tarkastellaan monimutkaisten järjestelmien suunnittelussa erityisesti vaatimusten määrittelyyn sopivia käyttäjakeskeisiä menetelmiä.

Vaatimusten määrittely on tutkimuksessa jaettu vaatimusten hankinta- ja kuvausprosesseihin. Näiden lisäksi vaatimukset on myös voitava validoida käyttäjien kanssa. Tutkimuksen tapaustutkimusosuudessa verrattiin kahta eri vaatimusten kuvaustapaa, use case ja ADS-malleja. Tapaustutkimuksen kohteena on Suomenlahden kansainvälinen alusten ilmoittautumisjärjestelmä (SRS).

Tutkimuksessa havaittiin, että teollisten järjestelmien käytettävyys on mutkikas ja moniulotteinen käsite, jonka teoreettinen määrittely vaatii syvällistä analyysia siitä, mitkä ovat järjestelmien hyvyyden kriteerit. Vain hyvän järjestelmän kriteerejä hyväksi käyttämällä voidaan määrittellä järjestelmän keskeiset vaatimukset.

Tätä julkaisua myy
VTT TIETOPALVELU
PL 2000
02044 VTT
Puh. (09) 456 4404
Faksi (09) 456 4374

Denna publikation säljs av
VTT INFORMATIONSTJÄNST
PB 2000
02044 VTT
Tel. (09) 456 4404
Fax (09) 456 4374

This publication is available from
VTT INFORMATION SERVICE
P.O.Box 2000
FIN-02044 VTT, Finland
Phone internat. + 358 9 456 4404
Fax + 358 9 456 4374