

Juha Lamminen

Intentiolähtöinen
lähestymistapa käytettävyyttä
koskevien vaatimusten
määrittelyyn

Kuluttajamarkkinoille suunnatun
innovatiivisen ohjelmistotuotteen
vuorovaikutussuunnittelu



Juha Lamminen

Intentiolähtöinen lähestymistapa
käytettävyyttä koskevien
vaatimusten määrittelyyn

Kuluttajamarkkinoille suunnatun innovatiivisen
ohjelmistotuotteen vuorovaikutussuunnittelu

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen Delta-salissa
joulukuun 19. päivänä 2012 kello 12.



JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ 2012

Intentional lähestymistapa
käytettävyyttä koskevien
vaatimusten määrittelyyn

Kuluttajamarkkinoille suunnatun innovatiivisen
ohjelmistotuotteen vuorovaikutussuunnittelu

JYVÄSKYLÄ STUDIES IN COMPUTING 165

Juha Lamminen

Intentiolähtöinen lähestymistapa
käytettävyyttä koskevien
vaatimusten määrittelyyn

Kuluttajamarkkinoille suunnatun innovatiivisen
ohjelmistotuotteen vuorovaikutussuunnittelu



JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ 2012

Editors

Seppo Puuronen

Department of Computer Science and Information Systems, University of Jyväskylä

Pekka Olsbo

Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-5021-7
ISBN 978-951-39-5021-7 (PDF)

ISBN 978-951-39-5020-0 (nid.)
ISSN 1456-5390

Copyright © 2012, by University of Jyväskylä

Jyväskylä University Printing House, Jyväskylä 2012

ABSTRACT

Lamminen, Juha

An Intention-Driven Approach for Usability-Related Requirements Engineering:
Interaction Design of Innovative Software products for Consumer Market

Jyväskylä: University of Jyväskylä, 2012, 120 p.

(Jyväskylä Studies in Computing,

ISSN 1456-5390; 165)

ISBN 978-951-39-5020-0 (nid.)

ISBN 978-951-39-5021-7 (PDF)

This dissertation examines the usability and user experience requirements from the viewpoint of human-technology interaction design, with the special focus on designing innovative consumer information systems. Based on the earlier research, design of innovative technology is seen to be very challenging from the requirements engineering perspective. The research presented in this dissertation aims to provide means to these challenges by analyzing human interaction behavior in human-technology interaction. The main research problem is how user needs analysis and requirements engineering should be executed during the design of innovative software products. In order to give answers to this research problem, a research strategy with theoretical, quantitative, and qualitative research methods is applied together with intention-driven requirements engineering thinking. As a result the following outcomes were developed: a) a framework for intention-driven requirements engineering, b) a light-weight human-technology interaction ontology, c) the Interaction Deconstruction method for usability-related requirements engineering, and d) the D-TEO method for obtaining deeper understanding of the limits of critical values concerning quantitative usability requirements. In addition to these practical artifacts, this dissertation offers an extensive literature review of the challenges and solutions in interaction design and requirements engineering.

Keywords: human interaction behavior, human-technology interaction, intention, usability, user experience, innovative consumer information systems, requirements engineering, user psychology

Author's address	Juha Lamminen Dept. of Computer Science and Information Systems University of Jyväskylä P.O. Box 35 40014 University of Jyväskylä, Finland juha.e.lamminen@gmail.com
Supervisors	Professor, Ph.D., Pertti Saariluoma Dept. of Computer Science and Information Systems University of Jyväskylä, Finland Lecturer, Ph.D., Mauri Leppänen Dept. of Computer Science and Information Systems University of Jyväskylä, Finland
Reviewers	Professor, Art.D., Turukka Keinonen School of Art and Design Aalto University, Finland Researcher, Ph.D., Eleni Berki School of Information Sciences University of Tampere, Finland
Opponents	Professor, D.Sc. (Tech.) Marko Nieminen School of Science Aalto University, Finland Research Scientist, Ph.D., Markku Häkkinen Center for Foundational and Validity Research Princeton, New Jersey, USA

ESIPUHE

Väitöskirjan tekeminen on pitkä prosessi. Sen aikana kirjoittaja ehtii keskustella monien eri ihmisten kanssa. Haluan kiittää kaikkia, jotka ovat osallistuneet näihin keskusteluihin. Kaikki entiset työtoverini Jyväskylän yliopiston tietojenkäsittelytieteiden- ja tietotekniikan laitokselta sekä Agora Centeristä ansaitsevat lämpimät kiitokset. Kiitokset kuuluvat myös nykyiselle työantajalleni Fortum Oyj:lle kaikesta tuesta ja ymmärryksestä liittyen väitöskirjan viimeistelyvaiheeseen.

Erikseen haluan kiittää väitöskirjani ohjaajia Pertti Saariluomaa ja Mauri Leppästä. Pertin ansioista kiinnostuin kognitiotieteistä ja opin ymmärtämään, kuinka suuri merkitys käyttäjäpsykologialla on onnistuneen ihmisen-tekniologia vuorovaikutuksen suunnittelussa. Pertti kannusti ja tuki minua läpi väitöskirjaprosessin luottamaan itseeni ja tuomaan rohkeasti omia ajatuksiani ja ideoitani osaksi väitöskirjaa. Maurin ansioista opin ymmärtämään, että tieteen tekemiseen tarvitaan hyvien ajatusten ja ideoiden lisäksi päämäärätietoisuutta, kurinalaisuutta ja kriittistä otetta omaa työtä kohtaan. Mauri on erinomainen ohjaaja. Huomasin monta kertaa prosessin aikana, että hän todellakin perehtyi työhöni, antoi laadukasta palautetta ja opetti minua ajattelemaan tieteellisesti. Kahden hyvin erityyppisen ohjaajan kanssa työskentely oli minulle jatko-opiskelijana aika ajoin haastavaa, mutta lopulta se ilmeni rikkautena monitieteellisessä tutkimuksessa. Yhteinen piirre molemmille ohjaajilleni on se, että he tuntevat poikkeuksellisen laajasti ja perusteellisesti omat tutkimusalueensa ja osaavat soveltaa osaamistaan sekä teoriassa että käytännössä. Kolmas henkilö, jolle haluan välittää erityiskiitokset väitöskirjaprosessin läpiviennistä, on Pekka Neittaanmäki. Pekka palkkasi minut tutkimusharjoittelijaksi Jyväskylän yliopistoon jo Pro Gradu -vaiheessa ja on luottanut tekemiseeni siitä lähtien. Pekka on erinomainen motivoija ja lisäksi hän on osoittanut aitoa kiinnostusta tutkimus-alueettani kohtaan.

Väitöskirjan tekeminen on muutakin kuin tutkimista ja kirjoittamista. Lämmin kiitos tukemisesta kuuluu vanhemmilleni Erkille ja Siskolle, Heidille perheineen, Sinille sekä sukulaisille ja kavereille.

Jyväskylä 8.12.2012

KUVIOT

KUVIO 1	Tutkimusalueen poikkitieteellisyys ja moniulotteisuus	13
KUVIO 2	Vuorovaikutussuunnittelun vaiheet	23
KUVIO 3	Vaatusmäärittelyn keskeiset vaiheet	26
KUVIO 4	Käyttäjakeskeisen suunnittelun prosessi.....	31
KUVIO 5	Intentionalähtöisen lähestymistavan mukainen vaatimusten määrittelyn prosessi.....	60

TAULUKOT

TAULUKKO 1	Käyttäjakeskeisen suunnittelun menetelmiä.....	33
TAULUKKO 2	Menetelmien soveltuvuus tuotekehityksen päävaiheisiin.....	34
TAULUKKO 3	Tutkimusmenetelmät tutkimuskysymysten ja artikkeleiden yhteydessä	72
TAULUKKO 4	Tutkimukseen sisällytetyt artikkelit	84

SISÄLLYS

ABSTRACT

ESIPUHE

KUVIOT JA TAULUKOT

SISÄLLYS

ALKUPERÄISET ARTIKKELIT

1	JOHDANTO.....	11
2	TEOREETTINEN TAUSTA	15
	2.1 Kuluttajamarkkinoille suunnatut innovatiiviset ohjelmistotuotteet: suunnittelun haasteet	15
	2.2 Vuorovaikutussuunnittelu	19
	2.3 Käyttäjätarpeiden tunnistaminen ja vaatimusmäärittely	23
	2.4 Käyttäjakeskeinen suunnittelu	30
	2.5 Käytettävyys- ja käyttäjäkokemusvaatimusten määrittely	34
	2.6 Yhteenveto ja vastaamattomat kysymykset.....	44
3	INTENTIO-LÄHTÖINEN VUOROVAIKUTUSSUUNNITTELU	46
	3.1 Intentio	46
	3.2 Ihmisen toiminta ja vuorovaikutustutkimus	48
	3.3 Kognitiivinen tehtäväanalyysi.....	53
	3.4 Käyttäjäpsykologia	56
	3.5 Kohti intentiolähtöistä vaatimusten määrittelyä	57
	3.6 Yhteenveto intentiolähtöisestä vuorovaikutussuunnittelusta	61
4	TUTKIMUSONGELMA JA -METODOLOGIA	63
	4.1 Tutkimusongelma ja -kysymykset	63
	4.2 Tutkimusmetodologia.....	70
	4.3 Tutkimuksen reliabiliteetti ja valideetti	75
5	TUTKIMUKSEEN SISÄLLYTETYT ARTIKKELIT.....	78
	5.1 Tutkimus I: Viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi.....	78
	5.2 Tutkimus II: Käyttäjätarpeiden tunnistaminen ihminen-tekniologia- vuorovaikutusontologian avulla kehitettäessä innovatiivista ohjelmistotuotetta	79
	5.3 Tutkimus III: Interaction Deconstruction -menetelmä käytettävyys- ja käyttäjäkokemusvaatimusten määrittelyyn	81
	5.4 Tutkimus IV: D-TEO -menetelmä innovatiivisen ohjelmistotuotteen käytettävyysvaatimusten määrittelyn tukena....	83
	5.5 Tutkimusartikkelien suhteet toisiinsa	84

6	YHTEENVETO	86
6.1	Kontribuutio	86
6.2	Tutkimuksen vahvuudet ja rajoitteet.....	92
6.3	Jatkotutkimuskohteet	94
	SUMMARY	96
	YHTEENVETO.....	98
	LÄHTEET	100

ALKUPERÄISET ARTIKKELIT

Leppänen, M., Lamminen, J. & Saariluoma, P. 2011. A framework for intention-driven requirements engineering of innovative software products. Teoksessa J. Pokorny, V. Repa, K. Richta, W. Wojtkowski, H. Linger, C. Barry & M. Lang (toim.) Information Systems Development. New York: Springer, 417-428.

Saariluoma, P., Parkkola, H., Honkaranta, A., Leppänen, M., Lamminen, J. 2009. User psychology in interaction design – the role of design ontologies. Teoksessa P. Saariluoma & H. Isomäki (toim.) Future Interaction Design II, Springer, Berlin, 69-86.

Saariluoma, P., Lamminen, J., & Leppänen, M. 2009. Ontologies for human-technology interaction design. Teoksessa Proceedings of the 16th International Product Development Management Conference, Twente, The Netherlands: European institute for advanced studies in management, EIASM.

Lamminen, J. & Saariluoma, P. 2010. Towards action-oriented user interface design. Teoksessa Proceedings of the 2010 International Conference on Management Science and Engineering, Hong Kong: ETP Engineering Technology Press, 58-61.

Lamminen, J., Rousi, R. & Saariluoma, P. 2011. Interaction deconstruction method for usability requirements engineering. Teoksessa K. Blashki (toim.) Proceedings of the IADIS International Conference Interfaces and Human Computer Interaction 2011, Portugali: IADIS Press, 161-168.

Lamminen, J., Leppänen, M., Heikkinen, R., Kämäräinen, A. & Jokisuu, E. 2009. A quantitative method for localizing user interface problems: D-TEO Method. Human Technology Journal 5 (2), 121-145.

1 JOHDANTO

“Requirements are a ubiquitous part of our lives, so it may seem strange that they have been singled out for study in computer science. Requirements and communication are inextricably intertwined. We start making our requirements clear soon after we are born by crying, usually for food. Parents soon become expert requirements engineers in inferring what their children want; but even in the cradle the dilemma of requirements is exposed. A baby’s cry is ambiguous. Does he or she want food, warmth or a cuddle? How do we translate our interpretation into the right food, degree of warmth, or appropriate rocking motion?” (Sutcliffe, 2002 s.1)

Oheinen Alistair Sutcliffen (2002) sitaatti tuo taidokkaasti esille vaatimusmäärittelyn perimmäisen luonteen monine haasteineen ja ongelmineen. Sitaatissa havainnollistetaan, kuinka monimutkaista ja vaikeaa vaatimusten taustalla olevien tarpeiden selvittäminen on. Esimerkissä tulee esille myös se, että vaatimukset ovat usein pohjimmiltaan ihmislähtöisiä. Sutcliffen (2002) mukaan vaatimusten määrittely on kuitenkin jo pitkään kehittynyt suuntaan, jossa tekninen suunnitteluprosessi korostuu enemmän kuin käyttäjien tarpeiden ymmärtäminen.

Tässä tutkielmassa keskitytään käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittämiseen tilanteessa, jossa kehitetään kuluttajamarkkinoille suunnattuja innovatiivisia ohjelmistotuotteita. Aihe on ajankohtainen, sillä kuluttajamarkkinoille suunnatut ohjelmistotuotteet ovat tulleet pysyväksi osaksi jokapäiväistä elämäämme. Esimerkiksi kuntosalivuoron varaaminen, bussiaikataulujen tarkistaminen tai junalipun ostaminen onnistuu älypuhelinsovelluksen avulla. Samat toimenpiteet kuluttaja voi toki suorittaa myös kotitietokoneella tai vaikkapa nykyaikaisella televisiolla. Esimerkkejä tällä hetkellä nousussa olevista kuluttajamarkkinoille suunnatuista innovatiivista ohjelmistotuotteista ovat päivittäistavaroiden ostamisen helpottamiseen tarkoitettut mobiilipalvelut. Tällaisten sovellusten avulla kuluttaja voi kaupassa ollessaan tarkastaa tuotetiedot lukemalla viivakoodin mobiililaitteen kameralla, kertoa elintarvikkeista ja resepteistä mielipiteitä ja suositella niitä ystäville. Sovellukset oppivat myös tunnistamaan kuluttajien makutottumukset ja osaavat ehdottaa niiden mukaisia ostosvaihtoehtoja.

Edellä esitetyt esimerkit tuovat esille sen, että kaikki kuluttajamarkkinoille suunnatut ohjelmistotuotteet eivät ole välttämättömiä, vaikka ne helpottavatkin elämää. Useissa tapauksissa ne ovat kuitenkin ainoa mahdollisuus selvitä arjesta, sillä perinteisempiä vaihtoehtoja ei enää tarjota. Esimerkiksi oma-toiminen laskujen maksaminen verkkopankilla on jo osa jokaisen kuluttajan arkipäivää. Teknologian soluttautuessa yhä syvemmälle kuluttajien arkeen ja elämään, palveluiden käytettävyys ja käyttäjäkokemus tulevat olemaan konkreettinen osa ihmisten elämää. Tällöin huonolle käytettävyydelle ja käyttökokemukselle ei ole enää sijaa ja kuluttajat tulevat asettamaan käytettävyydelle ja käyttäjäkokemukselle selvästi nykyistä kovempia vaatimuksia. Tutkimuksissa on jo nyt tuotu esille esimerkiksi se, että perinteinen käyttäjänäkökulma on liian suppea suunniteltaessa tämän tyyppisiä ohjelmistotuotteita (Tuunanen ym. 2008). Lisäksi on osoitettu, että kuluttajamarkkinoille suunnattujen ohjelmistotuotteiden suunnittelussa tulisi huomioida nykyistä perusteellisemmin käyttäjäkokemuksen näkökulma (Han ym. 2001; Kim & Han, 2008; Kwahk & Han, 2002; Rossi & Tuunanen, 2010).

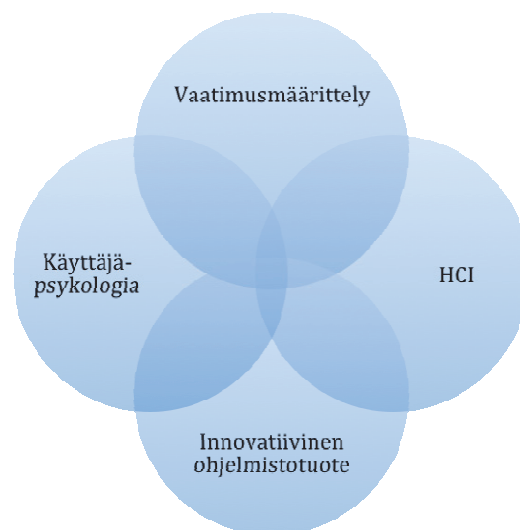
Olemassa olevat vuorovaikutussuunnittelun menetelmät, tekniikat ja lähestymistavat eivät tuota kuluttajille käytettävyyden ja käyttäjäkokemuksen näkökulmasta tarpeeksi laadukkaita ratkaisuja kehitettäessä kuluttajamarkkinoille suunnattuja innovatiivisia ohjelmistotuotteita. Käyttäjakeskeinen suunnittelu (ISO 13407, 1999) on jo pitkään ollut tästä näkökulmasta yksi lupaavimmista ohjelmistokehityksen lähestymistavoista. Käyttäjakeskeistä suunnittelua on kuitenkin kritisoitu sen käyttäjätarpeiden tunnistamisen ja vaatimusmäärittelyn puutteista. Esimerkiksi Viukarin (2010) mukaan jo käyttäjakeskeisen suunnittelun lähtökohtana on usein visio toteutettavasta tuotteesta ja käytettävästä teknologiasta. Tällöin vision tai teknologian taustalla oleviin käyttäjätarpeisiin ei keskitytä tarpeeksi ja huonoimmassa tilanteessa teknologia lähtee ohjaamaan suunnitteluprosessia. Käyttäjakeskeisen suunnittelun standardi ISO 13407 korvattiin vuonna 2010 uudistetulla ISO 9241-210 standardilla (ISO 9241-210, 2010), mutta sekään ei määrittele käyttäjätarpeiden tunnistamisen prosessia eikä ota kantaa vaatimusmäärittelyn menetelmiin ja tekniikoihin.

Kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta tulisi käyttäjätarpeiden tunnistamisessa ja vaatimusmäärittelyssä lähteä liikkeelle tutkimalla ihmisen toimintaa mahdollisimman teknologiariippumattomasta näkökulmasta. Tällaisessa tilanteessa intentio on lupaava lähtökohta käyttäjätarpeiden tunnistamiselle ja vaatimusmäärittelylle. Intentio on psykologian näkökulmasta keskeisin ihmisen toimintaa ohjaava ja kontrolloiva tekijä (Ajzen, 1985; Fishbein & Ajzen, 1975; Bratman, 1987; Gollwitzer & Bargh, 1996; Hampshire, 1965; Irwin, 1971; James, 1890; Lewin, 1951; Meiland, 1970; Miller ym. 1960; Shultz, 1980; Tolman, 1932).

Tässä väitöskirjassa intentiota lähestytään kahdesta eri näkökulmasta. Ensimmäisen näkökulman muodostaa *intentio* ihmisen toimintaa motivoivana ja kontrolloivana tekijänä (Brentano, 1874/1973; Bratman, 1987; Shultz, 1980). Tämä intention näkökulma on motivoinut kehittämään väitöskirjassa esitettävän intentiolähtöisen vaatimusmäärittelyn viitekehyksen ja ihminen-teknologia-

vuorovaikutusontologian. Toisen näkökulman intentioon väitöskirjassa muodostaa Millerin ym. (1960) määritelmä. Tämän määritelmän mukaan intentio viittaa sellaisiin toistaiseksi toteutumattomiin osiin ihmisen suunnitelmia, joiden toteuttaminen on kokonaisuudessaan jo alkanut. Määritelmä yhdistää intention ihmisen toimintaan yksittäisten operaatioiden tarkkuudella ja soveltuu hyvin käytettävyyss- ja käyttäjäkokemusvaatimusten määrittämisen taustateoriaksi. Väitöskirjassa esitetyt kaksi käytännönläheistä vaatimusmäärittelyä tukevaa menetelmää on kehitetty tästä näkökulmasta. Yhteenvedon voidaan sanoa, että väitöskirjan tulokset tarjoavat intentiolähtöisen viitekehyksen, ihmisen-tekniologia-vuorovaikutusontologian ja kaksi käytännönläheistä menetelmää kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vaatimusmäärittelyn tueksi.

Tutkimus on aidosti poikkitieteellinen. Siinä sovelletaan kognitiotieteen ja tietojärjestelmätieteen menetelmiä, lähestymistapoja, käsitteitä, teorioita ja tekniikoita ja luodaan uudentyypinen intentiolähtöinen vaatimusmäärittelyn lähestymistapa käytettävyyss- ja käyttäjäkokemusvaatimusten määrittelyn tueksi. Tutkimus sijoittuu vaatimusmäärittelyyn, ihmisen-tekniologia vuorovaikutustutkimukseen (HCI), innovatiivisen ohjelmistotuotteen ja käyttäjäpsykologian leikkauspintaan (kuvio 1).



KUVIO 1 Tutkimusalueen poikkitieteellisyys ja moniulotteisuus

Edellä mainittujen tieteenalojen sisällä tutkimus hyödyntää erityistietämystä seuraavilta osa-alueilta: hierarkkinen tehtäväanalyysi (HTA), kognitiivinen tehtäväanalyysi (CTA), kuluttajamarkkinoille suunnatut innovatiiviset ohjelmistotuotteet, käytettävyyssuunnittelu, käytettävyyteen liittyvien vaatimusten määrittely, käyttäjäkeskeinen suunnittelu, käyttäjäkokemuksen suunnittelu, käyttäjatarpeiden tunnistaminen, tavoitelähtöinen lähestymistapa ja toiminnan teoria.

Tämän tutkimuksen näkökulmasta keskeisimmät osa-alueet ovat kuitenkin intentio ihmisen toimintaa ohjaavana tekijänä, käyttäjäpsykologia ja vaatimusmäärittely sekä kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen kehittäminen.

Väitöskirjan johdanto-osan rakenne on seuraava. Luvussa 2 luodaan katsaus innovatiivisen ohjelmistotuotteen vuorovaikutussuunnitteluun vaatimusmäärittelyn näkökulmasta käymällä läpi seuraavat osa-alueet: kuluttajamarkkinoille suunnatut innovatiiviset ohjelmistotuotteet, vuorovaikutussuunnittelu, käyttäjätarpeiden tunnistaminen ja vaatimusmäärittely, käyttäjäkeskeinen suunnittelu ja käytettävyyteen liittyvien vaatimusten määrittely mukaan lukien käytettävyyss- ja käyttäjäkokemusvaatimukset. Luvun 2 toinen päämäärä on tuoda esille vaatimusmäärittelyn keskeiset haasteet kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta ja selvittää, kuinka nämä haasteet tulevat esille määriteltäessä erityisesti käytettävyyteen liittyviä vaatimuksia. Luvussa 3 pohjustetaan intentiolähtöistä lähestymistapaa vaatimusten määrittelyyn käymällä läpi: intention käsite, ihmisen toiminta vuorovaikutustutkimuksen näkökulmasta, kognitiivinen tehtäväanalyysi ja käyttäjäpsykologia. Luvussa 4 esitetään tutkimusongelma, tutkimuskysymykset ja metodologia. Luvussa 5 kuvataan tutkimuksen tulokset esittelemällä väitöskirjaan sisältyvien artikkelien yhteenvedot. Luvussa 6 keskustellaan tutkielman tärkeimmistä löydöksistä ja luodaan suuntaa tulevaisuuden tutkimukselle.

2 TEOREETTINEN TAUSTA

Tässä luvussa asemoidaan ja jäsennetään tutkimusalue pohjautuen väitöskirjan tutkimusongelmaan: Kuinka käyttäjätarpeiden tunnistaminen ja vaatimusten määrittely tulisi suorittaa osana vuorovaikutussuunnittelua kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta? Alaluvussa 2.1 selvitetään, mitä kuluttajamarkkinoille suunnatulla innovatiivisella ohjelmistotuotteella tarkoitetaan ja mitä haasteita sen kehittämiseen liittyy. Alaluvussa 2.2 lähestytään haasteita vuorovaikutussuunnittelun näkökulmasta ja alaluvussa 2.3 rajaudutaan tarkemmin käyttäjätarpeiden tunnistamiseen ja vaatimusten määrittelyyn. Alaluvussa 2.4 käydään läpi käyttäjakeskeisen suunnittelun keskeiset periaatteet, menetelmät ja prosessit. Alaluvussa 2.5 käsitellään käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyä. Alaluvussa 2.6 esitetään yhteenvedona ne keskeiset haasteet ja vastaamattomat kysymykset, joihin tässä tutkielmassa etsitään vastausta.

2.1 Kuluttajamarkkinoille suunnatut innovatiiviset ohjelmistotuotteet: suunnittelun haasteet

Innovaatiota voidaan tarkastella käsitteenä useista eri näkökulmista, kuten innovaation omaksuminen (Heiskala, 2007), innovaation suhteellinen uutuus (Damanpour & Wischnevsky, 2006), innovaation diffuusio (Greenhalgh ym. 2004), innovaation luominen (Gopalakrishnan & Damanpour 1997) tai innovaatioprosessi (Baregheh ym. 2009). Olennaista *innovaation* määrittelyssä on tehdä ero käsitteiden keksintö ja innovaatio välille. Keksintö tai idea muuttuu innovaatioksi siinä vaiheessa, kun se onnistutaan kaupallistamaan tai se otetaan jollakin muulla tavalla laaja-alaiseen käyttöön. (Greenhalgh ym. 2004; Pol & Ville, 2009) Myös innovatiivisen ohjelmistotuotteen kehittämistä on lähestytty useista eri näkökulmista informaatioteknologian alalla käyttäen hieman eri merkityksissä termejä "Information Systems Innovations" (Swanson, 1994), "Information Technology Innovation" (Moore & Benbasat, 1991) "Software-based Innovati-

on” (Quinn ym. 1996) ja ”Software Innovation” (Von Krogh ym. 2003). Tässä väitöskirjassa innovaatiota tarkastellaan innovaatioprosessin näkökulmasta keskittyen kuluttajamarkkinoille suunnatun ohjelmistotuotteen käyttäjätarpeiden tunnistamiseen ja vaatimusten määrittelyyn.

Innovaatioprosessi viittaa monivaiheiseen prosessiin, jossa organisaatio muuntaa uudet ideansa uusiksi tuotteiksi, palveluiksi tai prosesseiksi tai kehittää olemassa olevia tuotteita, palveluita tai prosesseja vahvistaakseen omaa etuaan ja kilpailukykyään sekä erottuakseen edukseen oman toimialansa markkinoilla (Baregheh ym. 2009). On huomattava, että kaikki kuluttajamarkkinoille suunnatut ohjelmistotuotteet eivät kuitenkaan ole innovatiivisia, eikä kaikkia innovatiivisia tuotteita ole suunnattu kuluttajamarkkinoille. Tässä väitöskirjassa keskitytään sellaisiin kuluttajamarkkinoille suunnattuihin ohjelmistotuotteisiin, jotka ovat innovatiivisia. *Kuluttajamarkkinoille suunnatulla ohjelmistotuotteella* (engl. Consumer Information System) viitataan järjestelmään, joka hyödyntää informaatioteknologiaa mahdollistaen kuluttajille Internetiä hyödyntäviä palveluita, joiden avulla kuluttajat pääsevät heille merkitykselliseen dataan käsiksi ja voivat muokata sekä ohjailta sitä (Bastek ym. 2008). Esimerkkejä kuluttajamarkkinoille suunnatuista ohjelmistotuotteista ovat massakäyttäjille suunnatut Internet-palvelut, vuorovaikutteiset tv-palvelut ja älypuhelinsovellukset.

Veryzer (2003) on määritellyt innovaation kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämisen näkökulmasta. Tällöin innovaatiolla viitataan muutokseen ja innovaation diffuusiolla kuluttajien innokkuuteen hyväksytyä muutos. Määritelmän mukaan innovaatiolle on välttämätöntä loppukäyttäjien hyväksynnän saavuttaminen. Käyttäjien on koettava, että uusi keksintö tai idea parantaa olemassa olevia käytänteitä, tuotteita tai palveluita. (Veryzer, 2003) Esimerkkejä kuluttajamarkkinoille suunnatuista innovatiivisista ohjelmistotuotteista ovat verkkokirjakauppa Amazon.com, verkkohuutokauppa eBay ja virtuaalimaailma Habbo Hotel.

Tässä väitöskirjassa kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen käyttäjätarpeiden tunnistaminen ja vaatimusmäärittely pyritään viemään läpi mahdollisimman teknologiariippumattomasti. *Teknologiariippumattomuudella* tarkoitetaan käyttäjätarpeiden pitämistä käytettävyyss- ja käyttäjäkokemusvaatimusten perustana läpi ohjelmistotuotteen suunnitteluprosessin riippumatta teknologisista ratkaisuksista. Tämä määritelmä on johdettu teknologialähtöisyyden määritelmästä. *Teknologialähtöisyydellä* tarkoitetaan tilannetta, jossa teknologia lähtee kehittäjien ja käyttäjien huomaamatta ohjaamaan suunnittelua kohti sellaisia suunnitteluratkaisuja, jotka eivät kohtaa alkuperäisiä käyttäjätarpeita (Bergvall-Kåreborn & Ståhlbröst, 2010; Kankainen & Oulasvirta, 2003; Patnaik & Becker, 1999; Vredenburg ym. 2002).

Motivaatiota innovatiivisten ohjelmistotuotteiden kehittämiseksi ja tutkimukselle voidaan tarkastella ainakin kolmesta eri näkökulmasta: globaali näkökulma, kilpailunäkökulma ja kehittäjän näkökulma (Cooper, 2000; Quinn ym. 1996; Rose, 2011). Globaali näkökulma (Rose, 2011) viittaa maailman teknologiseen globalisoitumiseen, jonka seurauksena ohjelmistot suunnitellaan palvelemaan koko ajan laajempia käyttäjäryhmiä ja organisaatioita eri maissa ja kult-

tuureissa. Siinä missä 1960-luvulla ohjelmistot suunniteltiin tyypillisesti tarkoin rajatuille ammattikäyttäjille, nykyään teknologia on kaikkialla, kuten autoissa, kauppakeskuksissa, kodeissa ja toimistoissa. Tämä on johtanut siihen, että yhä useammin uusi teknologia tulisi pystyä suunnittelemaan siten, että se pystyy vastaamaan ns. massakäyttäjien vaatimuksiin. Suurille käyttäjäryhmille ei kuitenkaan voida kehittää innovatiivisia ohjelmistosovelluksia rutiininomaisesti nykyisillä menetelmillä ja tekniikoilla, koska myös käyttäjät vaativat enemmän arvoa käyttämiltään ohjelmistosovelluksilta. (DiRomualdo & Gurbaxani, 1998; Heeks, 2002; Heeks & Kenny, 2002; Rose, 2011). Kilpailunäkökulma (Rose, 2011) viittaa siihen, että yritysten tulee pystyä pitämään mahdollisesti saavuttamansa kilpailukykyinen asema koko ajan kiristyvillä ohjelmistotuotannon markkinoilla. Menestyvän ohjelmistoyrityksen on pystyttävä olemaan koko ajan kehityksen aallonharjalla niin käytettävien teknologioiden, menetelmien ja tekniikoiden kuin loppukäyttäjän ymmärtämisenkin suhteen (Atuahene-Gima, 1996; Rose, 2011; Quinn ym. 1996). Kehittäjän näkökulma (Rose, 2011) viittaa ohjelmistojen suunnittelijoiden, kehittäjien, konsulttien ja muiden ohjelmistotuotannon ammattilaisten ymmärrykseen siitä, että innovatiivisia ohjelmistotuotteita ei pystytä kehittämään tehokkaasti turvautumalla perinteisiin ohjelmistotekniikan menetelmiin ja tekniikoihin. Tarvitaan uusia menetelmiä, tekniikoita ja ajattelutapoja uudentyyppisten haasteiden kohtaamiseen (Fisher ym. 2005; Rose, 2011; Tang, 1998; Von Krogh ym. 2003).

Ensimmäinen kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämisen haaste on perinteisen käyttäjänäkökulman rajallisuus. Tuunasen ym. (2008) mukaan tämän tyyppisiä sovelluksia ja palveluita kehitettäessä käyttäjää tulisi tarkastella nimenomaan kuluttajana. Myös muiden tutkimusten mukaan perinteiset näkemykset käyttäjästä ovat liian rajoittuneita. Esimerkiksi Lamb ja Kling (2003) rohkaisevat siirtymään sosiaalisen toimijan käsitteeseen perinteisestä käyttäjän käsitteestä. Tällöin käyttäjää tarkastellaan organisaatiossa huomioiden käyttäjän yhteydet (engl. affiliation), ympäristö, vuorovaikutus ja identiteetti. Käyttäjää voidaan tarkastella myös psykologian näkökulmasta. Käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004) on lähestymistapa, joka tarkastelee ihmistä laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja (Oulasvirta & Saariluoma 2004, 2006). Käyttäjäkeskeinen suunnittelu (ISO 13407, 1999) on puolestaan lähestymistapa, joka pyrkii huomioimaan käyttäjän perusteellisesti osana käytännön ohjelmistokehitystä.

Toinen kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämisen haaste on tehokkuus-tuottavuus-keskeinen suunnittelu. Kehitettäessä tämän tyyppisiä sovelluksia tulisi käytön tehokkuuden ja tuottavuuden sijasta keskittyä enemmän käytön tuottamiin hedonisiin piirteisiin, kuten nautinnollisuuteen ja mielihyvään (Kim & Han, 2008; Kwahk & Han, 2002; Rossi & Tuunanen, 2010; Tuunanen ym. 2008) sekä käyttäjien kokemuksiin ja vaikutelmiin (Han ym. 2001). Ihminen-teknologia vuorovaikutustutkimuksen alalla on jo pitkään käytetty käyttäjäkokemuksen näkökulmaa tukemaan käytettävyyden näkökulmaa, koska etenkin perinteinen käytettävyyden painottuu suorituskeskeiseen tehok-

kuuteen ja tuottavuuteen. Tällöin jätetään huomioimatta käyttäjän aiempi käyttökokemus, arvot, hedoninen ja kulttuurillinen näkökulma, estetiikka, odotukset ja tunteet. (Arhippainen, 2009; Blythe ym. 2003; Hassenzahl, 2002; Hassenzahl, 2003; Hassenzahl & Roto, 2007; Law ym. 2009; Tractinsky, 2000; Väänänen-Vainio-Mattila ym. 2008; Wright & Blythe, 2007). Perinteinen suorituskeskeinen näkemys käyttäjään ja käytettävyyteen on lähtöisin pyrkimyksistä kasvattaa työprosessien tehokkuutta ja tuottavuutta juontaen juurensa alunperin työanalyysin (engl. work analysis) eri muodoista, erityisesti deskriptiivisestä työanalyysistä (esim. Hoffman & Militello, 2009). Vuorovaikutussuunnittelu (Preece, 2002) ja käyttäjäkeskeisen suunnittelun vuonna 2010 uusittu standardi (ISO 9241-210, 2010) ovat käytettävyyttä ja käyttäjäkokemusta yhdistäviä lähestymistapoja ohjelmistokehitykseen.

Kahta edellä esitettyä haastetta vahvistaa myös sähköisten kuluttajatuotteiden (engl. consumer electronic products) käytettävyyteen rajautunut tutkimus (Han ym. 2001; Kim & Han, 2008; Kwahk & Han, 2002). Käsite sähköinen kuluttajatuote viittaa ohjelmiston lisäksi laitteistoon, kuten esimerkiksi fyysiseen DVD-soittimeen tai videonauhuriin. Tässä tutkielmassa keskitytään kuluttajamarkkinoille suunnattuihin innovatiivisiin ohjelmistotuotteisiin eikä laitteiston näkökulmaa huomioida.

Kolmas haaste on kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vaatimusmäärittely. Tämä haaste sisältää neljä osaongelmaa. 1) Tulevien käyttäjien on vaikea ilmaista ja visioida täysin uudentyyppiseen ohjelmistotuotteeseen liittyviä tarpeita ja vaatimuksia (Cassab ym. 2010). 2) Kehitettäessä täysin uudentyyppistä ja innovatiivista ohjelmistotuotetta ei välttämättä edes tiedetä keitä loppukäyttäjät ovat ja mihin tarkoitukseen he tulevat sovellusta käyttämään (Mayhew, 1999). 3) Vastaavaa tuotetta, jota voitaisiin käyttää vaatimusten määrittelyn lähtökohtana ei tyypillisesti ole olemassa (Dearden & Howard, 1998). 4) Tällaisessa suunnittelutilanteessa ei osata ennakoita, kuinka uusi tuote vaikuttaa loppukäyttäjien käyttäytymiseen, kuinka kilpailukykyinen tuote todellisuudessa on ja kenelle se itse asiassa tuo hyötyä. Vaatimusten määrittelyn näkökulmasta edellä mainittuja osaongelmia on pyritty ratkaisemaan keskittymällä vaatimusmäärittelyn alkuvaiheisiin (Yu 1997, 1999), soveltamalla tavoitepohjaisia lähestymistapoja (Chung ym. 2000; Dardenne ym. 1993; Lapouchnian, 2005) ja kehittämällä luovuutta edistäviä menetelmiä ja tekniikoita vaatimusmäärittelyn tueksi (Grube & Schmid, 2008; McFadzean, 1998).

Kuluttajamarkkinoille suunnatuista ohjelmistotuotteista löytyy useita esimerkkejä Internetiä hyödyntävistä palveluista, jotka voidaan yhdistää ainakin yhteen edellä mainituista neljästä osaongelmasta. Tällaisia ovat esimerkiksi verkkokirjakauppa Amazon.com ja verkkohuutokauppa eBay. Suomalainen Habbo Hotel virtuaalimaailma on esimerkki vastaavasta innovatiivisesta ohjelmistotuotteesta, joka voidaan yhdistää jokaiseen edellä mainittuun neljään haasteeseen. Alunperin Habbo Hotel oli hyvin rajatulle käyttäjäryhmälle kehitetty sovellus, jossa käyttäjät pystyivät liikkumaan esimerkiksi virtuaalisessa baarissa ja keskustelemaan siellä keskenään (Johnson ym. 2005). Nykyään pal-

velu nähdään sosiaalisena virtuaalimaailmana, jossa on ainakin 10 miljoonaa aktiivista käyttäjää 32 eri maasta. Palveluun rekisteröityminen on ilmaista, mutta käyttäjät voivat somistaa pienillä rahasiirroilla eli mikrotransaktioilla omia huoneitaan, personoida virtuaalihahmojaan tai liittyä Habbo Clubiin. (Mäntymäki & Salo, 2012) Esimerkiksi vuonna 2006 Habbo Hotel tuotti omistajalleen Sulake Oy:lle n. 55 miljoonaa USD pääosin mikrotransaktioiden kautta (Lehdonvirta ym. 2009). Habbo Hotel -virtuaalimaailman kasvu on jatkunut, sillä vuonna 2011 palveluun oli rekisteröitynyt maailmanlaajuisesti yhteensä jo 200 miljoonaa virtuaalihahmoa ja arviolta 100 000 uutta virtuaalihahmoa luotiin päivittäin (Ahmad, 2011). Tutkijoiden näkökulmasta Habbo Hotel on herättänyt paljon mielenkiintoa otollisena ympäristönä tutkia muun muassa virtuaalista kuluttajakäyttäytymistä, eri maiden kuluttajakulttuureja ja virtuaalirahaan liittyviä erityiskysymyksiä (Lehdonvirta ym. 2009; Mäntymäki & Salo, 2012). Sovelluksen menestyksen taustalla vaikuttaa olevan tutkimusten mukaan (Lehdonvirta ym. 2009) kuluttajatarpeiden syvälinen ymmärrys liittyen siihen, miksi sovelluksen käyttäjät ovat motivoituneet kuluttamaan rahaa sosiaalisessa virtuaalimaailmassa. Vastaukset tähän kysymykseen juontavat juurensa Lehdonvirran ja kumppanien (2009) mukaan lopulta vastaavanlaisten ihmisen tarpeiden äärelle (esim. nautinnollisuus, sosiaalinen identiteetti), joita Kraft (2012) on käsitellyt perustavanlaatuisina ihmisen tarpeina käyttäjätarpeiden tunnistamisen yhteydessä (esim. sosiaalisuuden tarve, vapaa-ajan tarve, nautinnollisuuden tarve). Habbo Hotel esimerkki tuo esille konkreettisesti sen, miksi on keskeistä selvittää kuinka nykyiset vaatimusmäärittelyn menetelmät ja tekniikat pystyvät vastaamaan edellä esitettyihin käyttäjätarpeiden tunnistamisen ja vaatimusmäärittelyn erityiskysymyksiin ja haasteisiin.

Tässä väitöskirjassa edellä esitettyjä haasteita lähestytään ihminen-teknologia vuorovaikutussuunnittelun näkökulmasta rajaten tutkimus käytettävyys- ja käyttäjäkokeemusvaatimuksiin. Tarkastelu aloitetaan vuorovaikutussuunnittelun käsitteestä. Tämän jälkeen keskitytään käyttäjätarpeiden tunnistamiseen ja vaatimusmäärittelyyn sekä käydään läpi käyttäjäkeskeisen suunnittelun perusteet. Lopuksi tuodaan esille käytettävyys- ja käyttäjäkokeemusvaatimusten määrittelyn haasteet sekä nostetaan esille aihealueen vastaamattomat kysymykset. Tarkastelun ulkopuolelle jätetään selvitys siitä, kuinka laajasti olemassa olevia vaatimusmäärittelyn lähestymistapoja, menetelmiä ja tekniikoita käytetään käytettävyys- ja käyttäjäkokeemusvaatimusten määrittelyssä.

2.2 Vuorovaikutussuunnittelu

Ihmisen ja tietokoneen välinen vuorovaikutustutkimus (engl. Human-Computer Interaction) on itsenäinen tieteenala, joka kattaa teoreettisen tutkimuksen ja käytännön soveltamisen näkökulmat ja edelleen niihin liittyvät metodologiat, teorialat ja yksityiskohtaiset käytännöt. HCI:n ytimen muodostaa tietokonepohjaisten järjestelmien kehittäminen aina suunnittelusta arviointiin mukaan lukien järjestelmien käytön opettelu sekä kaikki järjestelmien kehittämiseen ja käyt-

töön liittyvät dokumentaatiot. (Hartson, 1998). Kuten edellä esitetystä määritelmästä käy ilmi, HCI on varsin laaja tutkimusalue, jonka päämääränä on mahdollistaa ihmisille helppokäyttöisiä vuorovaikuttavia tuotteita ja järjestelmiä. Käsitteenä HCI:lle on vaikea antaa yksiselitteistä määritelmää. Esimerkiksi kokoomateoksessa ”Human-Computer Interaction in the New Millennium” (Carroll, 2002) HCI on määritelty noin 30 eri tavalla riippuen kirjoittajasta ja esitettävästä asiasta. Tieteenalana HCI on monitieteinen (Carroll, 2003; Coiera, 2003; Harrison ym. 2007) kattaen työtieteet, ergonomian, kognitiivisen psykologian, käyttäytymistieteet, tietojärjestelmätieteen ja tietotekniikan (Hartson, 1998).

Vuorovaikutussuunnittelu (engl. interaction design) on prosessi, jonka päämääränä on suunnitella vuorovaikuttavia tuotteita, jotka tukevat ihmisiä heidän jokapäiväisessä elämässään (Preece, 2002). Osana vuorovaikutussuunnittelua on pystyttävä luomaan, hahmottelemaan ja päättämään olemassa olevien resurssien puitteissa kaikki käyttäjään liittyvät kehitettävän artefaktin piirteet, kuten esimerkiksi tuotteen rakenteelliset, toiminnalliset, eettiset ja esteettiset ominaisuudet (Löwgren & Stolterman, 2005). Osaksi vuorovaikutussuunnittelua tulisi pystyä integroimaan myös tieteelliseen psykologiaan perustuva tietämys ihmisestä käyttäjänä (Saariluoma, 2004). Vuorovaikutussuunnittelu on kuitenkin vielä suhteellisen uusi ja vakiintumaton käsite. Käsitteet *vuorovaikutus* ja *suunnittelu* ovat jo itsessään laajoja ja vaikeasti määriteltäviä (Liikkanen, 2007). Vuorovaikutussuunnittelun luonne on lisäksi hyvin erilainen verrattuna esimerkiksi sellaisiin perustellusti määriteltyihin ja ymmärrettyihin suunnittelun käsitteisiin, kuten graafinen suunnittelu, käyttöliittymäsuunnittelu tai teollinen suunnittelu, jotka tuottavat tuloksenaan selkeämmin ymmärrettävän ja konkreettisen lopputuloksen (vrt. graafinen suunnittelu - grafiikka, käyttöliittymäsuunnittelu - käyttöliittymä, vuorovaikutussuunnittelu - vuorovaikutus).

Ihmisen ja tietokoneen välistä vuorovaikutusta voidaan tarkastella suunnittelun näkökulmasta jakamalla se kolmeen historialliseen kehitysvaiheeseen, eli ”kehitysaaltoon” (Bødker, 2006, Kampuri, 2011; Kaptelinin ym. 2003): 1) Käytettävyyssuunnittelu (engl. Usability Engineering), 2) Käyttäjakeskeinen suunnittelu (engl. User-Centred Design) ja 3) Käyttäjäkokeamussuunnittelu (engl. Designing for User Experience). Tässä tutkielmassa esitettävä intentiolähtöinen lähestymistapa käytettävyyden ja käyttäjäkokeamussuunnittelun määrittelyyn hyödyntää piirteitä kaikista edellä mainituista kolmesta kehitysaallosta. Seuraavassa käyn läpi nämä kolme vaihetta ja tuon esille niiden yhteydet intentiolähtöiseen lähestymistapaan. Myöhemmin väitöskirjan yhteenvedossa käsitelen tarkemmin tiettyjä lähestymistapoja, menetelmiä ja tekniikoita (katso esim. alaluvut 2.3, 2.4, 2.5, 3.2, 3.3, 3.4).

Käytettävyyssuunnittelu syntyi 1980-luvun alussa ja sille tyypillisiä piirteitä ovat keskittyminen käyttäjiin, tuottavuuteen ja työstä sekä työtehtävistä suoriutumiseen (Kampuri, 2011). Tuottavuuden näkökulman korostuminen on luonnollista, sillä samaan aikaan keskeiset käytettävyyden määritelmät (esim. Shackel, 1984) korostivat pääosin käyttäjän tehokasta ja tuottavaa tehtävistä suoriutumista. Myös hieman myöhäisemmät eksplisiittiset käytettävyyden

määritelmät (Nielsen, 1993; Shackel, 1991) ja käytettävyyttä koskevat standardit (ISO 9241-1, 1997; ISO 9126-1, 2001) olivat vaikuttamassa niin käytettävyyssuunnittelun kuin sitä tukeneiden menetelmien ja tekniikoiden kehittymiseen (Folmer & Bosch, 2004). Tällaisia menetelmiä ja tekniikoita ovat esimerkiksi käytettävyydestaus (engl. usability evaluation) ja käytettävyyden heuristinen arviointi (engl. heuristic evaluation) (Kamppuri, 2011). Lisäksi varhaiset käytettävyyssuunnittelun määrittelymenetelmät ja -tekniikat (esim. Gould & Lewis, 1985; Nielsen, 1993; Wixon & Wilson, 1997) kuuluvat selvästi tähän ”ensimmäiseen aaltoon”, koska ne perustuvat edellä mainittuihin varhaisiin käytettävyyden määrittelymenetelmiin (Folmer & Bosch, 2004). Käytettävyyssuunnittelulle tyypillisiä piirteitä olivat myös yksilöön painottuva näkökulma käyttäjään (ei huomioitu esim. ryhmä- ja organisaationäkökulmia), laboratorioissa suoritettut perusteelliset käyttäjäkokeet sekä pyrkimys määrittellä tarkkoja käytettävyyssuunnittelun kehitettävälle tuotteelle ja mitata niiden toteutumista osana vaatimusten määrittelyprosessia (Kamppuri, 2011). Psykologian näkökulmasta käytettävyyssuunnittelun ja sitä tukevien menetelmien ja tekniikoiden taustateorioiksi voidaan nähdä ergonomia, kognitiivinen psykologia ja kognitiotiede (Grudin, 1990; Kamppuri, 2011). Monista tutkimuksista (esim. Carroll, 1997; Olson & Olson, 2003) kuitenkin nähdään, että psykologisen tietämyksen sisällyttäminen käytettävyyssuunnitteluun ja siitä polveutuneisiin menetelmiin ja tekniikoihin on ollut vähäistä ja painottunut käytettävyyden testaamiseen ja arkitietämykseen.

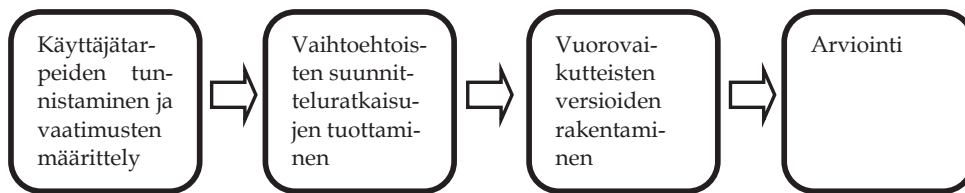
Käyttäjäkeskeinen suunnittelu alkoi muodostua siirryttäessä kohti 1990-lukua, kun muun muassa Internet, mobiililaitteet, sähköposti ja erilaiset ryhmätyöteknologiat toivat uuden teknologian työkontekstista yhtäkkiä osaksi lähes kaikkien ihmisten arkipäivää (Kamppuri, 2011). Motivaationa käyttäjäkeskeisen suunnittelun syntyyn voidaan nähdä siis esimerkiksi tarve huomioida käyttäjä yksilönäkökulman lisäksi myös osana ryhmää, organisaatiota ja tunnistaa muun muassa yksilöön ja ryhmiin liittyvät erilaiset laitteiden käyttöön liittyvät taustatekijät (Grudin, 1990). Tämän lisäksi motivaatiotekijänä nähtiin tarve ymmärtää teknologia ikään kuin välittävänä tekijänä ihminen-ihminen-vuorovaikutuksessa (Grudin, 1990). Käyttäjäkeskeinen suunnittelu korosti kontekstin merkitystä (Kamppuri, 2011) ja johti sellaisten menetelmien ja tekniikoiden kehittymiseen, kuten osallistuva suunnittelu (engl. Participatory Design) (Greenbaum & Kyng, 1991) ja kontekstuaalinen suunnittelu (engl. Contextual Design) (Beyer & Holtzblatt, 1998). Myöhemmin käyttäjäkeskeisen suunnittelun periaatteet voidaan nähdä olevan ikään kuin sinetöity ISO 13407-standardiin (ISO 13407, 1999) (katso esim. Leikas, 2009; Iivari & Iivari, 2006). Psykologian näkökulmasta käyttäjäkeskeinen suunnittelu otti vaikutteita organisaatiopsykologiasta ja sosiaalipsykologiasta huomioiden myös antropologian näkökulman (Grudin, 1990). Edelleen, psykologian näkökulmasta keskeisiä teoreettisia lähestymistapoja olivat tilannesidonainen kognitio (eng. situated action) (Suchman, 1987), fenomenologia (eng. phenomenology) (Dourish, 2001; Winograd & Flores, 1987), toiminnan teoria (eng. activity-theory) (Bødker, 1989, 1991; Kaptelinin ym. 1999; Kaptelinin & Nardi, 2006) ja hajautettu kognitio (eng. distributed cognition) (Hutchins, 1996; Salomon, 1993). Käyttäjäkeskeisen suunnittelun ISO 13407-

standardi (ISO 13407, 1999) päivitetään vuonna 2010 standardiksi ISO 9241-210 (ISO 9241-210, 2010). Uudistettu standardi pyrkii sisällyttämään käyttäjäkokemuksen entistä paremmin osaksi käyttäjäkeskeistä suunnittelua (Bevan, 2009b).

Käyttäjäkokeilu syntyi siirryttäessä kohti 2000-lukua vastaamaan teknologisen globalisaation haasteisiin, jossa teknologia on levinnyt kaikkien käytettäväksi riippumatta kulttuureista, arvoista, historiallisista taustoista tai muista taustatekijöistä. Käyttäjäkokeilun myötä on myös alettu kiinnittää enemmän huomioita siihen, kuinka teknologia saavuttaa erityisryhmät kuten vammaiset, kodittomat, muistihäiriöiset tai lukihäiriöiset. (Kamppuri, 2011). Käyttäjäkokeilun myötä teknologia on levittäytynyt selvästi laajemmalle verrattuna esimerkiksi 1990-lukuun muodostaen sellaisia tutkimussuuntauksia, kuten puettava tietotekniikka (eng. wearable computing), jokapaikan tietotekniikka (eng. pervasive and ubiquitous technology) ja käsin kosketeltavat käyttöliittymät (eng. tangible technology). (Kamppuri, 2011). Käyttäjäkokeilu korostaa uudemman sukupolven HCI-tutkimuksen merkitystä viitaten käsitteisiin arvot (eng. values) (Cockton, 2004), luottamus (eng. trust) (Fogg & Tseng, 1999), estetiikka (engl. aesthetics) (Tractinsky, 1997) ja tunteet (eng. emotional aspects of usability) (Norman, 2004).

Tässä väitöskirjassa kehitettävä intentiolähtöinen lähestymistapa käytettävyyden ja käyttäjäkokemuksivaatimusten määrittelyyn hyödyntää jokaista edellä mainittua ihmisen ja tietokoneen välisen vuorovaikutuksen kehitysaaltoa. Tähän suuntaan työtä on motivoinut yleisesti tunnettu (katso esim. Kamppuri, 2011) toteama siitä, että tällä hetkellä vuorovaikutussuunnittelun parhaat tulokset saavutetaan yhdistämällä kaikkien "kehitysaaltojen" vahvuudet toisiinsa. Intentiolähtöinen lähestymistapa hyödyntää käytettävyyssuunnittelua (ensimmäinen aalto) pyrkimällä tarkkoihin käytettävyyttä koskeviin vaatimuksiin, joiden toteutuminen voidaan testata käytettävyydestä tuttuun menetelmien avulla. Intentiolähtöinen lähestymistapa pyrkii määrittämään käytettävyyksivaatimusten lisäksi myös käyttäjäkokemusta koskevia vaatimuksia. Tällä tavoin perinteisiä käytettävyyden määritelmiä täydennetään käyttäjäkokemuksen (kolmas aalto) määritelmillä. Yhteistä intentiolähtöiselle lähestymistavalle ja käytettävyyssuunnittelulle on myös käytännönläheinen ote vuorovaikutustietämisen integroimiseksi osaksi vaatimustenmäärittelymenetelmiä. Käyttäjäkeskeisen suunnittelun näkökulmasta (toinen aalto) hyödynnetään tietämyksen osin tietämystä toiminnan teoriasta, hajautetusta kognitiosta ja fenomenologisesta lähestymistavasta sekä käyttäjäkeskeisen suunnittelun tarjoamia työkaluja vaatimusten määrittelyyn. Käytännön vuorovaikutussuunnittelumenetelmät (esim. Preece, 2002) pitävät käyttäjäkeskeistä suunnittelua käytännön vuorovaikutussuunnittelun lähtökohtana. Myös intentiolähtöinen lähestymistapa vaatimusten määrittelyyn noudattaa tiettyjä ISO 13407 -viitekehystä tuttuja periaatteita mukaan lukien käyttäjää koskevien vaatimusten määrittely ja niiden arviointi. Intentiolähtöisen lähestymistavan näkökulmasta suurimman eron kolmeen edellä esitettyyn "kehitysaaltoon" tuovat kuitenkin intentiolähtöisyys ja käyttäjäpsykologia, joita käsitellään väitöskirjan johdanto-osan luvussa 3.

Vuorovaikutussuunnittelu nähdään tyypillisesti iteratiivisena ja käyttäjakeskeisenä prosessina, joka koostuu neljästä vaiheesta (Kuvio 2) (Preece, 2002): a) Käyttäjätarpeiden tunnistaminen ja vaatimusten määrittely, b) Vaihtoehtoisten suunnitteluratkaisujen tuottaminen, c) Vuorovaikutteisten versioiden rakentaminen ja d) Arviointi. Vuorovaikutussuunnittelulle luonteenomaista on käyttäjakeskeisyyden ja iteratiivisuuden lisäksi keskittyminen käytettävyyden ja käyttäjäkokeisuusvaatimuksiin, jotka tulee tunnistaa ja dokumentoida huolellisesti jo projektin alkuvaiheessa. (Preece, 2002)



KUVIO 2 Vuorovaikutussuunnittelun vaiheet

Tässä väitöskirjassa keskitytään vuorovaikutussuunnittelun ensimmäiseen vaiheeseen (käyttäjätarpeiden tunnistaminen ja vaatimusten määrittely). Tämän lähestymistavan mukaan käyttäjätarpeiden tunnistaminen nähdään osana käytettävyyden ja käyttäjäkokeisuusvaatimusten määrittelyä. Seuraavassa alaluvussa tarkastellaan, mitä käyttäjätarpeiden tunnistamisella ja vaatimusten määrittelyllä tarkoitetaan sekä tuodaan esille, millaisia lähestymistapoja, menetelmiä ja tekniikoita tämän vaiheen tueksi on olemassa. Alaluvussa arvioidaan myös, mitkä vaatimusmäärittelyn menetelmät ja tekniikat soveltuvat innovatiivisen kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämiseen.

2.3 Käyttäjätarpeiden tunnistaminen ja vaatimusmäärittely

Käyttäjätarpeiden tunnistaminen (engl. user needs analysis) viittaa prosessiin, jonka aikana käyttäjien tarpeet (engl. user needs) kehitettävää ohjelmistosovellusta kohtaan tunnistetaan tyypillisesti kiinteänä osana vaatimusten määrittelyä (Anastassova ym. 2007; Coble ym. 1997; Kujala ym. 2001; Kotonya & Sommerville, 2002; Lindgaard ym. 2006; Mayhew, 1999; Preece, 2002; Robertson, 2001; Sutcliffe, 2002). Tällöin käyttäjätarpeiden tunnistaminen nähdään osana vaatimusten keräämisvaihetta (Anastassova ym. 2007; Preece, 2002; Sutcliffe, 2002). Myös tässä tutkielmassa esitettävä intentialähtöinen lähestymistapa määrittelee käyttäjätarpeiden tunnistamisen ja vaatimusmäärittelyn suhteen tällä tavoin. Joidenkin lähestymistapojen mukaan (esim. Bergvall-Kåreborn & Ståhlbröst, 2010; Lindgaard ym. 2006) käyttäjätarpeiden tunnistaminen voidaan nähdä myös omana ja itsenäisenä vaatimusten määrittelystä erillisenä vaiheena. Osa

lähestymistavoista tekee lisäksi selkeän eron tarpeen ja vaatimuksen käsitteiden välille siten, että tarpeet liittyvät ihmiseen ja vaatimukset teknisiin ratkaisuihin (Bergvall-Kåreborn & Ståhlbröst, 2010). Tässä väitöskirjassa keskitytään käytettävyyttä ja käyttäjäkokemusta koskeviin vaatimuksiin ja käyttäjätarpeiden tunnistaminen nähdään osana vaatimusten määrittelyä, jolloin näin selkeää kah-tiajakoa ei ole syytä tehdä.

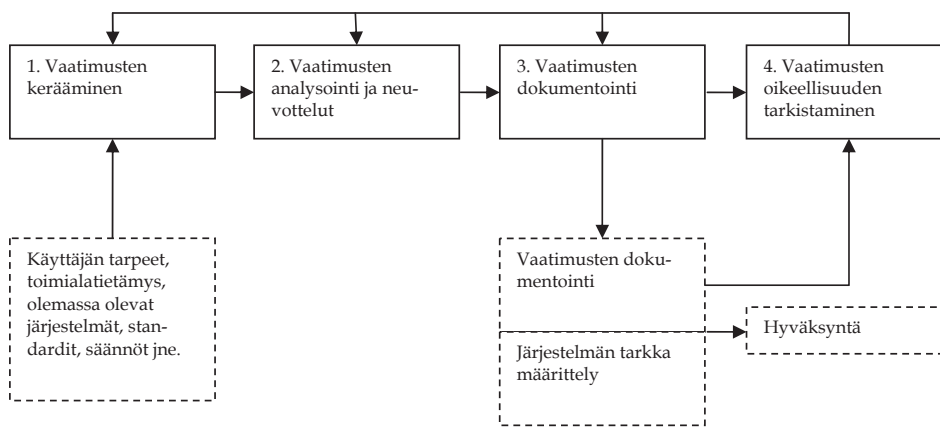
Käsitettä *käyttäjätarve* (engl. user need) voidaan lähestyä ainakin kahdesta eri näkökulmasta (Anastassova ym. 2007). Ensimmäisen näkökulman mukaan käyttäjätarpeella viitataan kyvykkyyteen (engl. capability), jonka toimija tarvitsee tai haluaa itselleen tehdä johonkin teknologian välityksellä (Robertson, 2001). Tämän määritelmän mukaan käyttäjätarpeen ja tavoitteen (engl. goal) käsitteet tarkoittavat samaa asiaa (Anastassova ym. 2007). Tavoitelähtöisen lähestymistavan (engl. goal-oriented) mukaan *tavoite* (engl. goal) viittaa syyhyn, miksi jotakin tehdään, valmistetaan tai käytetään (Lamsweerde, 2009). Tavoitteet voidaan muotoilla ja luokitella eri tarkkuustasojen mukaan vaihdellen korkeasta abstraktitasosta (esim. motivaatio ja tarve) matalaan abstraktitasoon (esim. tekninen yksityiskohta) (Lamsweerde, 2009). Toisen näkökulman mukaan käyttäjätarve syntyy sen keskustelun perusteella, jonka suunnittelijat, käyttäjät ja käytettävyyssiantuntijat käyvät läpi määritellössään tarpeita uudelle ohjelmistosovellukselle (Brangier, 2007). Tämän määritelmän mukaan käyttäjätarve nähdään jatkuvasti muuttuvana sosiaalisena konstruktiona (Anastassova ym. 2007), jolla on kosketuspintaa vaatimusten keräämisen lisäksi myös vaatimusten analysointivaiheeseen (engl. requirements analysis and negotiation). Käyttäjätarpeiden tunnistamisella on merkittävä rooli osana ohjelmiston kehitysprosessia. Käyttäjätarpeiden tunnistaminen on erityisen tärkeää kehitettäessä innovatiivista ohjelmistotuotetta (Lindgaard ym. 2006; Macintosh ym. 1978). Sen tavoitteena on mahdollistaa, että a) kehitettävä tuote vastaa loppukäyttäjien tarpeita, b) loppukäyttäjät hyväksyvät kehitettävän tuotteen ja sen käyttötarkoituksen ja c) lopputuotteen käyttö olisi mahdollisimman hyödyllistä (engl. utility) loppukäyttäjien näkökulmasta (Anastassova ym. 2007; McKeown ym. 1994). Käyttäjätarpeiden varhaisen tunnistamisen ja lopputuotteen hyödyllisyyden välillä onkin todettu selvä yhteys (Anastassova ym. 2007). Onnistuneella käyttäjätarpeiden tunnistamisella on myös yhteys innovaatioon erityisesti innovaation diffuusion kautta. Tällä tarkoitetaan innovatiivisen ohjelmistotuotteen kehittämisessä kuluttajien innokkuutta hyväksyä innovaation määrittämä muutos. Muutos hyväksytään paremmin käyttäjätarpeiden ja kehitettävän tuotteen kohdatessa aidosti toisensa. (Veryzer, 2003).

Käyttäjätarpeiden tunnistaminen ei ole kuitenkaan ongelmatonta. Käyttäjätarpeiden ja vaatimusten määrittelyn välinen suhde on epäselvä aiheuttaen muun muassa merkittäviä välillisiä taloudellisia menetyksiä osana käytännön ohjelmistoprojekteja (esim. Fenton & Pfleeger, 1997; Lindgaard ym. 2006). Käyttäjätarpeiden tunnistamiseen liittyvät menetelmät ja tekniikat eivät esimerkiksi anna selkeitä vastauksia seuraaviin kysymyksiin (Lindgaard ym. 2006): 1) Millaisia vaiheita käyttäjätarpeiden tunnistamiseen tulisi sisältyä? 2) Mitä käyttäjätarpeiden tunnistamisen lopputuloksena tulisi syntyä? 3) Kuinka käyttäjätar-

peiden tunnistamisen tulokset tulisi esittää ja dokumentoida? ja 4) Kuinka käyttäjätarpeiden tunnistuksen tuloksista tulisi keskustella ja kenen kanssa? (Lindgaard ym. 2006). Lisäksi käyttäjätarpeiden tunnistamista (esim. Lindgaard ym. 2006) ja vaatimusten määrittelyn alkuvaiheita (esim. Kotonya & Sommerville, 2002; Mayhew, 1999; Preece, 2002; Sutcliffe, 2002) käsittelevä keskeinen kirjallisuus vaikuttaa ohittavan lähes kokonaan tarpeiden käsittelyn psykologian näkökulmasta. Esimerkiksi toiminnan psykologian (Gollwitzer & Bargh, 1996) mukaan intentio, motivaatio ja arvot ovat keskeisessä asemassa, kun tutkitaan toimijan tarpeita. Tätä tietämystä on lisäksi tutkittu psykologiassa jo hyvin kauan ja useista eri näkökulmista (vrt. luku 3). Väitöskirjassa oletetaan, että lähestymällä vaatimusmäärittelyä intentialähtöisesti voidaan ymmärtää syvällisemmin, miksi joku ihminen haluaa tehdä jotakin tai ei (esim. teknologian avulla tai ilman sitä). Intentialähtöisyys avaa uusia näkökulmia vaatimusmäärittelyn lisäksi myös käyttäjätarpeiden tunnistamiseen. Yksi esimerkki tästä voisi olla intentialähtöinen käyttäjäryhmien määrittely. Käyttäjäryhmien määrittely on keskeinen osa sekä käyttäjätarpeiden tunnistamista että vaatimusten määrittelyä (esim. Anastassova ym. 2007). Seuraavaksi selvitetään vaatimusten määrittelyprosessi ja siihen kuuluvat keskeiset vaiheet.

Vaatimusten määrittely (engl. Requirements Engineering) on prosessi, jossa tunnistetaan järjestelmän sidosryhmät (engl. stakeholders) ja heidän tarpeensa kehitettävää järjestelmää kohtaan sekä dokumentoidaan tulokset tavalla, joka mahdollistaa vaatimuksia koskevan analyysin, neuvottelut ja tarkoituksenmukaisen järjestelmän kehittämisen (Nusebeih & Easterbrook, 2000). Vaatimusten määrittelyn tarkoitus on etsiä vastaukset kysymyksiin, ketkä ihmiset ovat tekemisissä järjestelmän kanssa, mitä he haluavat kehitettävältä järjestelmältä ja kuinka tietämys heidän tarpeistaan saadaan huomioitua järjestelmän suunnittelun tasolla (Sutcliffe, 2002). Vaatimusmäärittely liittyy läheisesti ohjelmistotekniikkaan (engl. software engineering), mutta näiden kahden käsitteen välinen ero on ilmeinen. Vaatimusten painopiste on oikeiden asioiden suunnittelussa ("designing the right thing") siinä missä ohjelmistotekniikka keskittyy asioiden suunnitteluun oikealla tavalla ("designing the thing right") (Boehm, 1981). Vaatimusten määrittely on käytännössä iteratiivinen prosessi, vaikka sen aktiviteetit kuvataan kirjallisuudessa tyypillisesti vaihe vaiheelta etenevinä yksittäisinä aktiviteetteina (Nusebeih & Easterbrook, 2000). Vaatimusten määrittelyprosessi koostuu neljästä aktiviteetista (Kuvio 3): vaatimusten kerääminen (engl. Requirements Elicitation), vaatimusten analysointi ja neuvottelut (engl. Requirements Analysis and Negotiation), vaatimusten dokumentointi (engl. Requirements Documentation) ja vaatimusten oikeellisuuden tarkistaminen (engl. Requirements Validation) (Kotonya & Sommerville, 2002). Ensimmäinen vaihe, *vaatimusten kerääminen*, voidaan jakaa a) sovelluksen toimialan ymmärtämiseen, b) keskeisen ongelman ymmärtämiseen, c) liiketoiminnan ymmärtämiseen ja d) järjestelmän sidosryhmien tarpeiden ja rajoitusten ymmärtämiseen. Vaatimusten keräämisen tueksi on olemassa useita menetelmiä ja tekniikoita (Nusebeih & Easterbrook, 2002). Esimerkiksi Sutcliffe (2002) on jakanut vaatimusten keräämisen tueksi kehitetyt tekniikat kuuteen eri luokkaan: haastattelutekniikat,

fokusryhmäteknikat, havainnointitekniikat, osallistumistekniikat, kysely- ja tutkimustekniikat ja dokumentaatioon perustuvat tekniikat. Toinen vaihe, *vaatimusten analysointi ja neuvottelut*, viittaa vaatimusten yksityiskohtaiseen analysointiin ja tarkasteluun sidosryhmien toimesta. Tässä vaiheessa käydään myös sidosryhmien välisiä keskusteluja liittyen vaatimusten viralliseen dokumentointiin ja erityisesti siihen, mitkä vaatimukset hyväksytään ja mitä ei eri rajoitteiden takia voida hyväksyä. (Kotonya & Sommerville, 2002). Vaatimusten analysoinnin ja neuvottelujen tueksi on kehitetty useita erilaisia menettelytapoja ja strategioita (esim. Nunamaker ym. 1991; Elboushi & Sherif, 1997).



KUVIO 3 Vaatimusmäärittelyn keskeiset vaiheet (Kotonya & Sommerville, 2002)

Kolmannessa vaiheessa, *vaatimusten dokumentoinnissa*, vaatimukset dokumentoidaan sopivalla tarkkuustasolla. Erilaisia vaihtoehtoja vaatimusten dokumentoinnin tarkkuustasoiksi edustavat formaalit, puoliformaalit ja informaalit menetelmät ja tekniikat (Nuseibeh & Easterbrook, 2002). *Vaatimusten oikeellisuuden tarkistaminen* tarkoittaa vaatimusmäärittelydokumentin huolellista läpikäyntiä, ennen kuin vaatimuksia käytetään kehitettävän sovelluksen kehittämiseen (Kotonya & Sommerville, 2002). Vaatimusten oikeellisuuden tarkistaminen on keskeinen osa vaatimusten määrittelyprosessia ollen vastuussa siitä, että loppukäyttäjät todellakin ymmärtävät millaista ohjelmistotuotetta heille ollaan kehitämässä (Sutcliffe, 2002). Skenaariot ovat yksi tyypillisimmistä vaatimusten oikeellisuuden tarkistamiseen sovelletuista tekniikoista (esim. Heymans & Dubois, 1998; Sutcliffe ym. 1998).

Tässä tutkielmassa keskitytään vaatimusten määrittelyprosessin kahteen ensimmäiseen vaiheeseen (vaatimusten kerääminen, vaatimusten analysointi ja neuvottelut) painottuen vaatimusten keräämiseen ja siihen sisältyviin alivaiheisiin nimenomaan käytettävyy- ja käyttäjäkokemusvaatimusten näkökulmasta. Vaatimusten keräämistä voidaan lähestyä eri tavalla viitaten kahteen englantilaiseen termiin, jotka ovat "requirements elicitation" ja "requirements cap-

turing”. Näiden termien välinen ero määritellään siten, että ”elicitation” korostaa, että vaatimuksia ei voida vain ”kerätä” kysymällä kysymyksiä vaan kyse on monimutkaisemmasta ja laajemmasta ongelman asettelusta (esim. Goguen & Jirotko, 1994; Kotonya & Sommerville, 2002; Nuseibeh & Easterbrook, 2002). Tässä tutkielmassa vaatimusten keräämistä lähestytään termin ”elicitation” kautta.

Vaatimukset voidaan perinteisesti jakaa toiminnallisiin ja ei-toiminnallisiin vaatimuksiin (Chung ym. 2000; Sutcliffe, 2002). Tässä tutkimuksessa *toiminnallisilla vaatimuksilla* viitataan niihin toimintoihin ja palveluihin, joita kehitettävän järjestelmän odotetaan tarjoavan (Mitä järjestelmän täytyy tehdä?) ja *ei-toiminnallisilla vaatimuksilla* viitataan rajoituksiin ja reunaehtoihin, joita asetetaan järjestelmälle ja sen kehittämiseksi (Kuinka järjestelmän täytyy se tehdä?). (Sommerville, 2004). Jako toiminnallisiin ja ei-toiminnallisiin vaatimuksiin on hyvin karkea ja tästä syystä kummankin kategorian vaatimuksia voidaan jakaa tarpeen mukaan edelleen lukuisiin eri alakategorioihin (esim. Chung ym. 2000; Preece ym. 2002). Vuorovaikutussuunnittelun näkökulmasta vaatimusten määrittelyprosessi kattaa sekä toiminnalliset että ei-toiminnalliset vaatimukset, jotka määritellään käyttäjakeskeisen suunnittelun periaatteiden mukaisesti (Preece, 2002). Tässä tutkielmassa keskitytään ei-toiminnallisten käytettävyyttä ja käyttäjäkokeumusvaatimusten määrittämiseen. Ennen perehtymistä käytettävyyttä koskeviin vaatimuksiin on syytä tarkastella, millaisia lähestymistapoja vaatimusten määrittelyyn on kehittynyt eri vuosikymmenien aikana.

Vaatimusten määrittelyä voidaan lähestyä useista eri näkökulmista. Alun perin tarve määritellä yksityiskohtaisia vaatimuksia kehitettävälle teknologialle lähti liikkeelle monimutkaisten sotilas- ja avaruusjärjestelmien kehittämisen kautta 1940-luvulla (Sommerville, 2005). Ensimmäisenä ohjelmistokehityksen lähestymistapana, jolla on ollut merkittävä vaikutus vaatimusten määrittelyyn kehittämiseen, voidaan pitää Winston Roycen 1970-luvulla (Royce, 1970) esittämää vesiputousmallia (Larman & Basil, 2003). Vesiputousmalli on vaiheittain etenevä prosessi, jossa vaatimusten määrittely on kuvattu prosessin ensimmäiseksi vaiheeksi. Vesiputousmallin toimintamalli on omaksuttu vuosien saatossa niin voimakkaasti, että se aiheuttaa edelleen tiettyjä ”virheellisiä” oletuksia vaatimusten määrittelyprosessia kohtaan. Tällaisia ovat esimerkiksi (Sommerville, 2005) seuraavat oletukset: a) vaatimusten määrittely on täysin itsenäinen vaihe, joka tehdään ennen varsinaista ohjelmistokehitystä, b) vaatimukset eivät muutu enää merkittävästi siinä vaiheessa, kun varsinainen ohjelmistokehitys alkaa ja c) vaatimusten määrittelyvaihe tulee erottaa ja irrottaa ohjelmiston suunnitteluvaiheesta. Vastoin yleistä käsitystä, kirjallisuudessa on esitetty ja todistettu (Larman & Basil, 2003), että Winston Royce pyrki itse asiassa kehittämään joustavia, iteratiivisia, inkrementaalaisia ja evolutionäärisiä lähestymistapoja ohjelmistokehitykseen, joista vesiputousmalli oli merkittävästi yksinkertaistettu versio 1970-luvun käytännön ohjelmistokehityksen tarpeisiin. 1970-luku oli tietynlaista vaatimusten määrittelymenetelmien ja -tekniikoiden pioneeriaikaa, jonka juuret juonsivat aina 1940-luvulle saakka. 1970-luvulla ei vielä pyritty selvästi yhdistämään tietämystä käyttäjästä, käytettävyydestä tai ihminen-teknologia

vuorovaikutuksesta osaksi vaatimusmäärittelyä. Toisaalta tarkasteltaessa erityisesti käyttöliittymäsuunnittelua (engl. user-interface design) voidaan jo 1970-luvulla nähdä viitteitä loppukäyttäjän huomioimisesta osana ohjelmistokehitystä (Grudin, 1990) aivan ensimmäisten kotitietokoneiden tulon myötä (Gentner & Grudin, 1990).

1980-luvulla vaatimusmäärittelyä lähestyttiin sellaisten ohjelmistokehityksen lähestymistapojen mukaisesti, jotka veivät samaan aikaan eteenpäin koko tietojenkäsittelytieteiden tutkimusta (Larman & Basil, 2003; Sommerville, 2005). Yksi esimerkki tällaisesta merkittävästä tutkimuksesta on oliokeskeinen lähestymistapa (engl. object-oriented) (Jacobson ym. 1992; Rumbaugh ym. 1991), jolla oli keskeinen rooli myöhemmin UML:n (engl. Unified Modeling Language) (Booch ym. 1998; Jacobson ym. 1992) syntymiseen ja sen soveltamiseen osana vaatimusten määrittelyä (Sommerville, 2005). Toinen esimerkki 1980-luvulla tehdystä tutkimuksesta, jolla on ollut suuri vaikutus vaatimusten määrittelyn kehittämiseen, on ohjelmistokehityksen spiraalimalli (Boehm, 1986). Vaatimusten määrittelyn näkökulmasta 1980-luvulla korostuivat iteratiivisuus, inkrementaaliset mallit ja evolutionäärinen prototyyppi (katso esim. Larman & Basil, 2003). Vaatimusten määrittelyä lähestyttiin siis näkökulmasta, jonka mukaan kaikkia vaatimuksia ei tarvinnut määrittellä kerralla täydellisesti. 1980-luku oli merkittävä myös käytettävyyteen liittyvien vaatimusten määrittelyn näkökulmasta, koska silloin kehittyi myös jo aiemmin tämän väitöskirjan yhteenvedossa esitetty käytettävyyssuunnittelu (engl. usability engineering), joka aloitti ns. ihminen-teknologia vuorovaikutussuunnittelun ensimmäisen aallon (Kamppuri, 2011). Itse asiassa myös käyttäjäkeskeinen suunnittelu (engl. user-centered design) alkoi jo saada merkittävää jalansijaa siirryttäessä kohti 1980-luvun loppupuolta (katso esim. Gould & Lewis, 1985; Gulliksen ym. 2003; Karat, 1997; Norman, 1986).

1990-luvulle tyypillisiä vaatimusten määrittelyn lähestymistapoja olivat näkökulmapohjainen (engl. viewpoint-oriented), skenaariopohjainen (engl. scenario-based) ja tavoitelähtöinen (engl. goal-oriented) vaatimusten määrittely (Cheng & Atlee, 2007; Lamsweerde, 2000; Sommerville, 2005; Sutcliffe, 1998). 1990-luvulla kehitettiin myös IEEE-standardi (IEEE, 1998) vaatimusten dokumentoinnille, pyrittiin vahvistamaan monista näkökulmista vaatimusten määrittelyn prosessimalleja (Kotonya & Sommerville, 2002) ja korostettiin matemaattisia formaaleja menetelmiä osana vaatimusten määrittelyä (Sommerville, 2005). Myös juurensa jo 1970 ja 1980-luvulla luoneet iteratiiviset ja inkrementaaliset ohjelmistokehityksen menetelmät ja tekniikat saivat edelleen paljon huomioita 1990-luvun vaatimusmäärittelytutkimuksessa (Larman & Basil, 2003). 1990-luku oli merkittävä myös käytettävyyteen liittyvien vaatimusten tutkimuksen näkökulmasta, sillä tuohon aikaan käyttäjäkeskeinen suunnittelu löi itsensä kunnolla läpi niin tutkijoiden kuin kehittäjienkin keskuudessa samaan aikaan kun käyttäjäkokemussuunnittelun perusteita kehitettiin (Kamppuri, 2011). Kuitenkin, edelleen 1990-luvulla ja myös 2000-luvun alussa vaatimusmäärittelyä lähestyttiin pääosin prosessina, joka pyrki määrittämään mahdollisimman tarkat vaatimukset ennen kuin varsinainen ohjelmiston tekninen kehi-

tys alkoi (Sommerville, 2005). Poikkeuksena tälle ajattelumallille 2000-luvun vaihteessa olivat ja ovat edelleen esimerkiksi ketterät menetelmät (Abrahams-son ym. 2003; Beck, 1999; The Agile Manifesto, 2001), joiden mukaan vaatimukset kehittyvät ja tarkentuvat jatkuvasti ohjelmiston kehitysprosessin aikana.

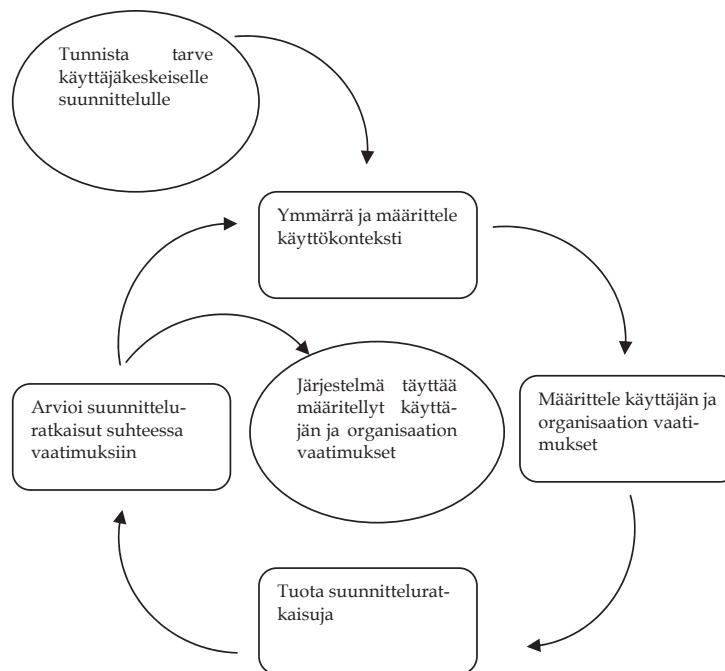
2010-luvun haasteet vaatimusten määrittelyä kohtaan voidaan nähdä kulminoituvan innovatiivisten ohjelmistotuotteiden nopeaan kehittämiseen siten, että ne vastaavat loppukäyttäjien tarpeita. Esimerkiksi Sommerville (2005) on selvittänyt 2010-luvun haasteiksi vaatimusten määrittelylle neljä pääkohtaa. 1) Tarve löytää sopiva tasapaino olemassa olevan teknologian, uuden teknologian ja käyttäjien tarpeiden välillä kehitettäessä innovatiivista ohjelmistotuotetta. Uusi teknologia perustuu usein pieniin muutoksiin, jota tehdään olemassa oleviin järjestelmiin ja sovelluksiin, mutta vaatimusten uudelleenkäyttö on tästä näkökulmasta puutteellista. 2) Tarve nopealle ohjelmistotuotteiden kehittämiseen. Ohjelmistotuotteiden kehittämisen näkökulmasta bisnesmaailma toimii nykyään uskomattoman nopeasti ja perinteiset tavat vaatimusten määrittelyyn ovat auttamatta liian hitaita. Uudet tuotteet myös katoavat markkinoilta nopeasti ja tilalle tulee saada välittömästi uusia kilpailukykyisiä tuotteita. Lisäksi kilpailijayritykset muuttavat jatkuvasti strategioitaan ja näihin muutoksiin tulee pystyä reagoimaan välittömästi. 3) Vaatimusten jatkuva muuttuminen. Tämä on seurausta pyrkimyksestä edellä mainittuun nopeaan ohjelmistokehitykseen. Riskinä on tällöin vaatimusten määrittelyn toteuttaminen hosuen ja huolimattomasti, johtaen jopa dramaattisiin lopputuloksiin. Vaikka vaatimusten määrittely toteutettaisiin nopeasti, sen tulee olla perusteellista ja yksityiskohdat käsiteltävää. 4) Sijoitetun pääoman voittoprosentin (ROI) maksimointi. Ohjelmistoyritykset haluavat tehdä ymmärretysti mahdollisimman hyvää tulosta ja tuottaa voittoa, jolloin olemassa olevien järjestelmien hyödyntäminen halutaan maksimoida eikä suinkaan pyritä aina kehittämään ”jotakin uutta” puhtaalta pöydältä. Tämä johtaa tilanteeseen, jossa ”innovatiivinen” ohjelmistotuote ei itse asiassa eroa paljon edellisestä versiostaan. Vaatimusten määrittelyn tulisikin pystyä saavuttamaan taso, jossa voidaan vertailla, kuinka paljon uusia piirteitä uuteen tuotteeseen itse asiassa tarvitaan ja kuinka tuotteen edellinen ja uusi versio nivoutuvat toisiinsa. Sommerville (2005) ei tuo juurikaan esille yhtä tärkeimmistä vaatimusmäärittelyn haasteista tulevaisuudelle. Psykologian ja käyttäytymistieteiden laajalla tutkimusalueella vuosikymmenien aikana saavutettu ymmärrys ihmisestä teknologian käyttäjänä tulisi pystyä integroimaan nykyistä paremmin osaksi käyttäjätarpeiden tunnistamista ja vaatimusmäärittelyä. Ellei meillä ole syvällistä tieteeseen perustuvaa ymmärrystä esimerkiksi ihmisten toiminnan tarpeista ja toiminnan motivaatioista, emme pysty kehittämään tarpeeksi laadukkaita innovatiivisia ohjelmistotuotteita. (Bergvall-Kåreborn & Ståhlbröst, 2010; Kankainen & Oulasvirta, 2003; Patnaik & Becker, 1999). Tässä tutkielmassa esiteltävä intentialähtöinen lähestymistapa eroaa aiemmin esitellyistä vaatimusten määrittelyn lähestymistavoista eniten juuri tästä näkökulmasta.

2.4 Käyttäjakeskeinen suunnittelu

Käyttäjakeskeinen suunnittelu tarkoittaa käyttäjätietämyksen huomioimista ja käyttäjän mukaan ottamista kaikissa ohjelmistokehityksen vaiheissa. Käyttäjä huomioidaan tavalla, joka käsittää käyttäjän kyvyt ja tarpeet aiottua järjestelmää kohtaan, tehtävät joita käyttäjä suorittaa aiotulla järjestelmällä ja ympäristön jossa aiottua järjestelmää käytetään. (Stone, 2005). Vuorovaikutussuunnittelu perustuu käyttäjakeskeisen suunnittelun periaatteisiin, menetelmiin ja prosesseihin, jotka käydään läpi tässä alaluvussa ISO 13407-standardin näkökulmasta. ISO 13407-standardin mukaiset periaatteet, prosessit ja menetelmät ovat yleisesti tunnettuja ja laajasti käytettyjä sekä tutkittuja (Maquire & Bevan, 2002; Väänänen-Vainio-Mattila, 2011; Jokela ym. 2003; Jokela ym. 2006; Kantola & Jokela, 2007). ISO 13407-standardi (ISO 13407, 1999) päivitettiin vuonna 2010 standardiksi ISO 9241-210 (ISO 9241-210, 2010). Tähän väitöskirjaan sisältyvät artikkelit keskittyvät ISO 13407-standardiin. Tästä syystä myös väitöskirjan yhteenvedo painottuu käyttäjakeskeiseen suunnitteluun ISO 13407-standardin näkökulmasta tuoden kuitenkin esille suurimmat muutokset standardien välillä.

Käyttäjakeskeisen suunnittelun neljä periaatetta ovat (ISO 13407, 1999): 1) Käyttäjien aktiivinen osallistuminen sekä käyttäjä- ja tehtävävaatimusten selkeä ymmärtäminen, 2) Toimintojen kohdentaminen käyttäjien ja teknologian välillä, 3) Suunnitteluratkaisujen iterointi ja 4) Monialainen suunnittelu. Ensimmäisellä periaatteella viitataan siihen, että tuotteen loppukäyttäjien tulee osallistua suunnitteluun siten, että he ovat vuorovaikutuksessa kehittäjien kanssa (Väänänen-Vainio-Mattila, 2011). Tämän periaatteen edellytyksenä on suunnitteluun osallistuvien käyttäjien huolellinen valinta (Gulliksen ym. 2003). Käyttäjien aktiivinen osallistuminen suunnitteluprosessiin tukee sekä vaatimusten määrittelyä että yksityiskohtaisempaa suunnittelua (Kujala, 2003). Toisella periaatteella tarkoitetaan tasapainon löytämistä ihmiselle ja teknologialle soveltuvien toimintojen kesken. Teknologian hoidettavaksi kannattaa esimerkiksi antaa rutiinitehtävät, joissa käsitellään nopeasti suuria tietomääriä ja monimutkaisia hakemistorakenteita. Ihminen ei kuitenkaan halua "ulkoistaa" kaikkia toimintojaan teknologialle. Tutkimusten mukaan tällaisia toimintoja ovat esimerkiksi kotiintulorutiineihin liittyvät asiat, kuten valojen päälle laittaminen ja verhojen sulkeminen (Koskela & Vainio-Väänänen-Mattila, 2004). Kolmannella periaatteella viitataan suunnitteluratkaisuihin kohdistuvaa toistuvaa palautetta, joka kerätään käyttäjiltä. Tyypillisesti palaute kohdistuu suunnitteluratkaisujen kautta käyttäjää ja organisaatiota koskeviin vaatimuksiin (Väänänen-Vainio-Mattila, 2011). Suunnitteluratkaisujen iteroinnin tulee sisältää empiirisiä mittauksia, joissa käyttäjät suorittavat mahdollisimman todenmukaisia tehtäviä esimerkiksi käyttöliittymäprototyyppien avulla (Gulliksen ym. 2003). Iteroinnin painopiste tulee olla suunnittelun alkuvaiheissa (Spool ym. 1997). Neljännellä periaatteella viitataan suunnitteluryhmään, joka koostuu eri ammattialojen asiantuntijoista (esim. käytettävyys, markkinointi, muotoilu, teknologia) (Väänänen-Vainio-Mattila, 2011).

ISO 13407-standardi määrittelee edelleen keskeiset vaiheet vuorovaikutteisten järjestelmien käyttäjakeskeiselle suunnitteluprosessille. Prosessi (Kuvio 4) alkaa siitä, että organisaatiossa tunnistetaan tarve käyttäjakeskeiselle suunnittelulle. Tämän jälkeen prosessin vaiheita toistetaan iteratiivisesti, kunnes kehitettävä järjestelmä täyttää sille asetetut käyttäjä- ja organisaatiovaatimukset. Käyttäjakeskeisen suunnitteluprosessin vaiheet ovat: 1) Ymmärrä ja määrittele käyttökonteksti, 2) Määrittele käyttäjän ja organisaation vaatimukset, 3) Tuota suunnitteluratkaisuja ja 4) Arvioi suunnitteluratkaisut suhteessa vaatimuksiin.



KUVIO 4 Käyttäjakeskeisen suunnittelun prosessi (ISO 13407, 1999)

Uudistetun standardin (ISO 9241-210, 2010) mukaan käyttäjakeskeisen suunnittelun kuusi periaatetta ovat: 1) Suunnittelun yksikäsitteinen perustuminen ymmärrykseen käyttäjistä, tehtävistä ja ympäristöstä, 2) Käyttäjien osallistuminen kaikkiin suunnittelu- ja kehitysvaiheisiin, 3) Suunnittelun käytettävyyssarviointikeskeisyys, 4) Suunnittelun iteratiivisuus, 5) Käyttäjäkokemuksen kokonaisvaltainen huomiointi ja 6) Suunnitteluryhmän monialaisuus. Ensimmäisellä periaatteella viitataan käyttökontekstin ymmärtämiseen painottuen käyttäjäryhmien määrittelyyn. Tällöin keskeistä on saavuttaa tietämys esimerkiksi siitä, miksi ja mihin tarkoitukseen tietty käyttäjäryhmä haluaa käyttää aiottua järjestelmää tietyssä käyttökontekstissa (Travis, 2011). Toisella periaatteella tarkoitetaan sitä, että käyttäjien tulee osallistua kaikkiin suunnitteluprosessin

vaiheisiin. Esimerkiksi se, että käyttäjät osallistuisivat vain suunnittelun alkuvaiheeseen tai käytettävyydestäukseen ei riitä (Travis, 2011). Kolmannella periaatteella viitataan siihen, että käyttäjäkeskeinen käytettävyyden ja käyttäjäkokemuksen arviointi ohjaa suunnittelua. Käytännössä tämä tarkoittaa sitä, että käytettävyyden- ja käyttäjäkokemusraviointeja tehdään läpi suunnitteluprosessin eikä arvioida käytettävyyttä pelkästään järjestelmän valmistuttua. (Travis, 2011). Neljännellä periaatteella tarkoitetaan sitä, että aiotun tuotteen suunnittelu perustuu iteratiivisen suunnittelun hyödyntämiseen. Tämän periaatteen oletuksena on, että vuorovaikutteista järjestelmää ei saada sovitettua optimaalisesti käyttäjien tarpeisiin ilman toistuvia suunnittelukierroksia, jotka perustuvat käyttäjiltä saatuun palautteeseen (Travis, 2011). Viidennellä periaatteella viitataan sekä käytettävyyden että käyttäjäkokemuksen sisällyttämiseen osaksi vuorovaikutteisen järjestelmän suunnittelua. Tämä periaate voidaan nähdä merkittävimpänä muutoksena verrattaen aiempaan standardia (ISO 13407, 1999) ja uudistettua standardia (ISO 9241-210) toisiinsa (Hertzum & Clemmensen, 2012). Kuudennella periaatteella tarkoitetaan suunnitteluryhmää, joka koostuu useiden eri tieteenolojen asiantuntijoista. Tämä periaate on tarkoitukseltaan vastaavasti esitettyä myös aiemmassa standardissa (vrt. ISO 13407, 1999; ISO 9241-210, 2010).

On tärkeää ymmärtää, että sekä ISO 13407 että ISO 9241-210-standardit kuvaavat vuorovaikutteisten järjestelmien käyttäjäkeskeisen suunnitteluprosessin vain yleisellä tasolla. Jokainen prosessin vaihe tarvitsee käytännössä rinnalleen yksityiskohtaisemmat menetelmät ja tekniikat (Jokela ym. 2003; Väänänen-Vainio-Mattila, 2011). ISO/TR 16982-standardissa (ISO/TR 16982, 2002) esitellään joukko tällaisia menetelmiä ja tekniikoita (Taulukko 1). ISO/TR 16982-standardiin perustuen esimerkiksi vaiheen 1 (Ymmärrä ja määrittele käyttök konteksti) tukena voidaan käyttää käyttök kontekstiin kohdistuvaa haastattelua tai mallipohjaista analyysiä. Vastaavasti vaihe 4 (Arvioi suunnitteluratkaisut suhteessa vaatimuksiin) voidaan toteuttaa käyttäen menetelminä käyttäjien tarkkailua ja suorituksiin liittyviä mittauksia.

ISO/TR 16982-standardissa (ISO/TR 16982, 2002) arvioidaan myös edellä esitettyjen menetelmien ja tekniikoiden soveltuvuutta tuotekehityksen eri vaiheisiin (Taulukko 2). Vaatimusmäärittelyn tueksi menetelmistä sopivat hyvin käyttäjien tarkkailu, kysely, haastattelut ja ääneen ajattelu. Toteutusvaiheen tueksi sopivat hyvin suorituksiin liittyvät mittaukset, ääneen ajattelu, luovat menetelmät ja dokumenttien analyysi. Testauksen tueksi sopivat hyvin suorituksiin liittyvät mittaukset, kysely ja haastattelut.

Tässä tutkielmassa ollaan kiinnostuneita niistä vaatimusmäärittelyä tukevista menetelmistä ja tekniikoista, joita voidaan hyödyntää määriteltäessä käytettävyyteen liittyviä vaatimuksia kuluttajamarkkinoille tarkoitetuille innovatiivisille ohjelmistotuotteille. Tällaisessa suunnittelutilanteessa sopivien menetelmien ja tekniikoiden valinta on yksi keskeinen kysymys. Maquire ja Bevan (2002) ovat jäsentäneet käyttäjäkeskeiseen suunnitteluun soveltuvia menetelmiä ja tekniikoita käyttäjä- ja käytettävyyden vaatimusten näkökulmasta. Heidän esittämänsä tutkimus auttaa sopivien menetelmien ja tekniikoiden valinnassa ver-

tailemalla 25 menetelmän sopivuutta käyttäjakeskeisen suunnittelun prosessin (ISO 13407, 1999) eri vaiheisiin ja tuomalla esille niiden vahvuudet ja heikkoudet.

TAULUKKO 1 Käyttäjakeskeisen suunnittelun menetelmiä (ISO/TR 16982, 2002)

Menetelmä	Kuvaus
Käyttäjien tarkkailu	Käyttäjien toiminnan tarkka ja systemaattinen havainnointi tuotteen käyttöympäristössä.
Suorituksiin liittyvät mittaukset	Määrällisiä mittauksia, joilla pyritään ymmärtämään käytettävyyso Ongelmien vaikutukset.
Kriittisten tapahtumien analyysi	Hyvien ja huonojen käyttötilanteiden systemaattinen arviointi.
Kysely	Käyttäjiltä kerätään mielipiteitä tuotteesta ja sen käyttöliittymästä.
Haastattelut	Haastattelija kerää käyttäjiltä syvällistä tietoa heidän tarpeistaan ja tuotteen käytöstä.
Ääneen ajattelu	Käyttäjä ajattelee käytettävyydestin aikana ääneen.
Yhteistyössä tapahtuva suunnittelu ja arviointi	Tuotteen suunnitteluun osallistuvat henkilöt kokoontuvat yhdessä suunnittelemaan tai arvioimaan tuotetta.
Luovat menetelmät	Työpajoissa tai aivoriihissä käytettäviä menetelmiä, joiden avulla pyritään kehittämään uusia tuotteita tai tuotepiirteitä.
Dokumenttien analyysi	Tuotetta koskevan dokumentaation tarkastelu, joka suoritetaan käytettävyyden asiantuntijoiden toimesta.
Mallipohjaiset analyysit	Käytetään tuotteen käyttöä kuvaavia malleja ennustamaan käyttäjien toimintaa.
Asiantuntija-arvioinnit	Käytettävyyden asiantuntijan tekemä käytettävyy sarviointi, joka perustuu arvioijan kokemukseen, ammatitaitoon ja mahdollisesti heuristiikkasääntöihin.
Automaattiset arvioinnit	Algoritmeihin pohjautuva automaattinen menetelmä, jolla arvioidaan tuotteen käytettävyyttä ja toimintaa verrattuna ennalta määriteltyihin malleihin ja sääntöihin.

Menetelmien ja tekniikoiden valinta ei kuitenkaan ole ristiriidatonta. Esimerkiksi tehtäväänalyysi soveltuu hyvin epäselvien ja monimutkaisten käyttäjätarpeiden selvittämiseen, mutta on yleensä liian formaalinen ja työläs menetelmä kattamaan yksityiskohtaisesti koko kehitettävän järjestelmän yksittäiset tehtävät (Maquire & Bevan, 2002). Verrattuna ISO/TR 16982 -standardiin, Maquire ja Bevan (2002) tuovat esille menetelmät ja tekniikat tarkemmalla tasolla (esim. listat yksittäisistä menetelmistä lähdetietoineen). Esimerkiksi Maquire ja Bevan (2002) selvittävät, että aivoriihet (engl. Brainstorming) sopivat hyvin käytettävyyden arviointiin tilanteessa, jossa halutaan yhdistää innovatiivinen ajattelu ja nopea käytettävyyden arviointi ryhmätyönä. ISO/TR 16982 -standardin mukaan luovat menetelmät, joihin myös aivoriihet luokitellaan, eivät sovellu käytettävyyden arviointiin. Edelleen tarkasteltaessa erityisesti innovatiivisen oh-

jelmistotuotteen vaatimusten määrittelyä voidaan huomata, että myös sitä on lähestytty usein luovien menetelmien, kuten aivoriilien (engl. brainstorming) ja luovien ideapajojen (engl. creativity workshops) näkökulmasta (esim. Grube & Schmid, 2008; Maiden & Robertson, 2005a, 2005b; Maiden ym. 2004a; 2004b; McFadzean, 1998). Tämä on luonnollista, sillä tarkasteltaessa vaatimusten määrittelyä innovatiivisen ohjelmistotuotteen näkökulmasta määritellään se kirjallisuuden perusteella luovana prosessina, jossa sidosryhmät ja kehittäjät työskentelevät yhdessä luodakseen uudelle tuotteelle ideoita, jotka voidaan lopulta kuvata vaatimuksina (Maiden ym. 2004a, Maiden ym. 2004b). ISO/TR 16982 -standardin (ISO/TR 16982, 2002) mukaan aivoriihat ja luovat pajat eivät kuitenkaan sovellu parhaalla mahdollisella tavalla vaatimusten määrittelyn menetelmiksi (vrt. taulukko 2).

TAULUKKO 2 Menetelmien soveltuvuus tuotekehityksen päävaiheisiin (ISO/TR 16982, 2002)

Menetelmä	Vaatimusmäärittely	Toteutus	Testaus
Käyttäjien tarkkailu	++	+	+
Suorituksiin liittyvät mittaukset	+	++	++
Kriittisten tapahtumien analyysi	+		+
Kysely	++	+	++
Haastattelut	++	+	++
Ääneen ajattelu	++	++	+
Yhteistyössä tapahtuva suunnittelu ja arviointi	+	+	+
Luovat menetelmät	+	++	
Dokumenttien analyysi	+	++	+
Mallipohjaiset analyysit	+	+	+
Asiantuntija-arvioinnit	+	+	+
Automaattiset arvioinnit		+	+

Sopii vaiheeseen (+), Sopii hyvin vaiheeseen (++)

Sopivien menetelmien ja tekniikoiden valintaan tilanne-/projektikohtaisesti voidaan käyttää edellä esitettyjen jäsennysten lisäksi esimerkiksi kehitysprojektin piirteistä (esim. projektin koko, monimutkaisuus, resurssit) johdettuja suosituksia (Jian ym. 2005). Tällöin voidaan esimerkiksi arvioida, kannattaako vaatimusten keräämiseen käyttää fokusryhmää, haastatteluja vai etnografista tutkimusta.

2.5 Käytettävyys- ja käyttäjäkokemusvaatimusten määrittely

Vaatimusten määrittelyn näkökulmasta vuorovaikutussuunnittelussa tulee kehittäväälle tuotteelle määritellä sekä käytettävyysvaatimukset että käyttäjäko-

kemusvaatimukset (Preece, 2005). Käytettävyyksivaatimuksilla tarkoitetaan kvalitatiivisia ja/tai kvantitatiivisia käytettävyyden tavoitteita (engl. usability goal) kehitettävälle tuotteelle. Kvalitatiiviset käytettävyyksivaatimukset (esim. käyttäjän tyytyväisyys) ovat tyypillisesti subjektiivisia ja vaikeita mitata. Kvantitatiiviset käytettävyyksivaatimukset (esim. aika, joka kuluu tehtävän suorittamiseen) ovat objektiivisia ja mitattavia vaatimuksia. (Stone, 2005). Kvalitatiiviset käytettävyyksivaatimukset nähdään tyypillisesti yleisinä käytettävyyssuunnittelun päämäärinä ja kvantitatiiviset käytettävyyksivaatimukset tarjoavat tarkat kriteerit käytettävyyden arvioinnille (Mayhew, 1999). Käyttäjäkokeskusvaatimuksilla viitataan tyypillisesti subjektiivisiin vaatimuksiin, jotka sisältävät käyttäjäkokemuksen näkökulman (Preece, 2005; Stone, 2005). Käyttäjäkokesmuksen näkökulma on laaja kattaen muun muassa aiemman käyttökokesmuksen, arvot, hedonisen näkökulman, kulttuurillisen näkökulman, estetiikan, odotukset ja tunteet (Arhippainen, 2009; Blythe ym. 2003; Hassenzahl, 2002; Hassenzahl, 2003; Hassenzahl & Roto, 2007; Law ym. 2009; Tractinsky, 2000; Väänänen-Vainio-Mattila ym. 2008; Wright & Blythe, 2007).

Vaatimusten määrittelyn näkökulmasta käsitteiden käytettävyyks (engl. usability) ja käyttäjäkokesmuks (engl. user experience) erottaminen on kuitenkin vaikeaa. Esimerkiksi Bevan (2009a) on todennut, että käytettävyyden ja käyttäjäkokesmuksen suhdetta voidaan lähestyä kolmesta eri näkökulmasta: 1) Käyttäjäkokesmuks käytettävyyden määrittelyn laajentajana siten, että käyttäjäkokesmuks tarkoittaa nimenomaan tyytyväisyys (engl. satisfaction) attribuuttia (Bevan, 2009a). 2) Käyttäjäkokesmuks täysin käytettävyydestä erillisenä käsitteenä sillä perusteella, että käytettävyyden juuret ovat suoriutumiskeskeiset (engl. user performance) painottuen mahdollisimman nopeaan ja virheettömään tehtävästä suoriutumiseen (Roto & Obrist, 2009). 3) Käyttäjäkokesmuks ”sateenvarjokäsitteenä”, joka kattaa sekä objektiivisen että subjektiivisen käyttäjään liittyvän informaation (ISO 9241-210, 2010). Tässä väitöskirjassa integroidaan käytettävyyks ja käyttäjäkokesmuksvaatimusten määrittely yhtenäiseksi prosessiksi soveltamalla hierarkkista tehtäväanalyysiä (HTA), kognitiivista tehtäväanalyysiä (CTA) ja käyttäjäpsykologiaa. Tällöin käytettävyyttä ja käyttäjäkokesmuksa käsitellään vaatimusten määrittelyprosessin alkuvaiheessa erillisinä käsitteinä, mutta prosessin loppuvaiheessa ne sulautetaan toisiinsa.

Käytettävyyksvaatimusten määrittelyn tueksi on kehitetty erilaisia menetelmiä ja prosessimalleja (Gould & Lewis, 1985; Mayhew, 1999; Nielsen, 1993; Wixon & Wilson, 1997). Tyypillisesti käytettävyyksvaatimusten määrittelyprosessi etenee siten, että ensin määritellään kvalitatiiviset käytettävyyksvaatimukset, joista keskeisimmät tarkennetaan prosessin edetessä kvantitatiiviseen muotoon. Esimerkiksi Wixonin ja Wilsonin (1997) esittämä 6-vaiheinen käytettävyyksvaatimusten määrittelyprosessi alkaa käyttäjien ja käyttäjäryhmien kartoittamisesta ja päättyy siihen, että käytettävyyden attribuuteille asetetaan tavoitearvot (esim. maksimiaika, joka saa kulua tietyn tehtävän suorittamiseen). Vastaavasti Mayhew (1999) esittää 9-vaiheisen prosessin, jonka aikana kvalitatiiviset käytettävyyksvaatimukset ensin määritellään ja priorisoidaan ja tämän jälkeen tärkeimmiksi priorisoidut vaatimukset pyritään tarkentamaan kvantitatiiv-

visiksi käytettävyyksivaatimuksiksi. Nielsenin (1993) mukaan kvantitatiiviset arvot voidaan asettaa esimerkiksi perustuen tuotteen edelliseen versioon tai kilpailevaan tuotteeseen. Kehitettävän tuotteen ollessa täysin uudentyypinen kvantitatiivisten kriteerien asettaminen on haastavampaa, mutta Nielsenin mukaan ne voidaan selvittää tällöin esimerkiksi perustuen käytettävyyssiantuntijoiden arvioihin ja aikaisempiin kokemuksiin. Edellä mainitut käytettävyyksivaatimusten määrittelyn tueksi kehitetyt menetelmät ja prosessimallit (Gould & Lewis, 1985; Mayhew, 1999; Nielsen, 1993; Wixon & Wilson, 1997) eivät sisällä käyttäjäkokemusvaatimusten määrittelyä vaan keskittyvät työprosessien tehokkuuden ja tuottavuuden kasvattamiseen (Stone, 2005). Seuraavaksi tässä luvussa käydään läpi seuraavat käytettävyyks- ja käyttäjäkokemusvaatimusten määrittelyyn liittyvät haasteet: a) kvantitatiivisten käytettävyyksivaatimusten määrittely, b) käytettävyyks- ja käyttäjäkokemusvaatimusten sisällön analyysi, ja c) käytettävyyden ja käyttäjäkokemuksen arviointi. Tämän jälkeen alaluvussa 2.6 esitetään yhteenveto ja vastaamattomat kysymykset perustuen lukuun 2 kokonaisuudessaan.

Haasteena kvantitatiivisten käytettävyyksivaatimusten määrittely

Käytettävyyksivaatimusten määrittely on osoittautunut käytännössä ongelmalliseksi ja usein laiminlyödyksi osaksi vuorovaikutussuunnittelua (Heiskari ym. 2009; Jokela ym. 2005; Kantola & Jokela, 2007). Yksi keskeinen haaste tällä tutkimusalueella on kvantitatiivisten käytettävyyksivaatimusten määrittäminen (Jokela ym. 2005). Kirjallisuudesta löytyy useita tutkimuksia, joissa on määritelty kvalitatiivisia käytettävyyksivaatimuksia (esim. Nivala ym. 2008; Tsalgatiidou ym. 2003), mutta empiirisiä tutkimuksia liittyen kvantitatiivisiin käytettävyyksivaatimuksiin on hyvin vähän (Kantola & Jokela, 2007). Tyypillisesti kvalitatiiviset käytettävyyksivaatimukset on kuvattu yleisellä tasolla (esimerkiksi "Käyttöliittymän pitäisi olla hyvin yksinkertainen ja käyttäjäystävällinen...", "virheilmoitusten tulisi olla selkeitä, informatiivisia ja selvästi erottuvia..."). Tällöin ne toimivat ikään kuin yleisen tason ohjeina käyttöliittymäsuunnitteluun, mutta ne eivät missään nimessä tarjoa tarvittavia kriteerejä käytettävyyksiarvioinnin tueksi.

Tarkasteltaessa edellä kuvattua haastetta tavoitelähtöisestä (engl. goal-oriented) näkökulmasta vaikuttaa siltä, että käytettävyyksivaatimukset kuvataan usein korkean tason tavoitteina, mutta niitä ei jalosteta edelleen matalan tason tavoitteiksi ja/tai vaatimuksiksi. Tavoitelähtöisessä vaatimusten määrittelyssä (Chung ym. 2000; Lamsweerde, 2009) tavoitteet voidaan jakaa korkean tason tavoitteisiin ja matalan tason tavoitteisiin, joista tarkennetaan edelleen varsinaisia vaatimuksia. Kirjallisuuden mukaan korkean tason tavoitteista voidaan jalostaa matalan tason tavoitteita tarkentamalla niitä jatkuvilla "kuinka?" kysymyksillä (Lamsweerde, 1995; Lamsweerde, 2000). Esimerkiksi yleisen tason käytettävyyksivaatitetta "virheilmoitusten tulisi olla selkeitä, informatiivisia ja selvästi erottuvia..."), voitaisiin tarkentaa "kuinka?" kysymyksillä muotoon "virheilmoitusten tulisi olla selkeitä, informatiivisia ja selvästi erottuvia

käyttämällä niissä punaista fonttia ja asettamalla ne virhekohtien välittömään läheisyyteen...". Tämän tarkennuksen ansiosta käytettävyyksivaatimus tuo lisäarvoa myös käytettävyyden arvioinnille, sillä se on selvästi todennettavissa. Vastaavasti kysymällä "miksi?" kysymyksiä saadaan yksityiskohtaiset vaatimukset ja matalan tason tavoitteet purettua korkean tason tavoitteiksi tavoitelähtöisen lähestymistavan mukaan (Lapouchnian, 2005).

Purettaessa matalan tason tavoitteita korkean tason tavoitteiksi saatetaan löytää aikaisempaa korkeamman abstraktitason tavoitteita, joista voidaan mahdollisesti jalostaa uusia matalan tason tavoitteita ja vaatimuksia (Lapouchnian, 2005). Itse asiassa käytettävyyksivaatimusten ja matalan tason käytettävyyksivaoitteiden purkaminen mahdollisimman korkean abstraktitason tarkkuudelle saattaisi avata mielenkiintoisia näkökulmia esimerkiksi käyttöliittymäsuunnitteluratkaisujen analysointiin. Esimerkiksi, miksi käyttöliittymäsuunnittelija on tiettyä käyttöliittymän vuorovaikutusongelmaa ratkaistessaan päätenyt käyttämään juuri tietynlaista valikkoratkaisua, tai miksi hän on päätenyt suunnittelemaan osaksi käyttöliittymää juuri sellaisen animaation, josta käyttäjät ärsyyntyvät. Suunnitteluratkaisujen analysointi ei kuulu tämän tutkimuksen keskeisten tutkimuskysymysten joukkoon, mutta sivuaa näitä useista näkökulmista. Hypoteesina voidaan esittää, että käyttöliittymäsuunnittelijan suunnitteluratkaisut ovat usein peräisin arkipsykologiasta, arkitietämyksestä tai introspektiosta (itsensä tarkasteluun perustuva ihmiskäsitys). Voidaan myös olettaa, että tieteellisen psykologian käsitteiden, teorioiden, säännönmukaisuuksien ja metodien soveltaminen osana käyttöliittymän suunnittelua on edelleen harvinaista. Edellä esitetty esimerkki ilmentää vain yhtä ongelmaa vuorovaikutussuunnittelun ja vaatimusmäärittelyn rajapinnassa. Selvää on kuitenkin, että käytettävyyksivaatimusten määrittelyn tueksi tarvitaan uusia käyttäjäpsykologiaan (Moran, 1981; Saariluoma, 2004) perustuvia menetelmiä ja tekniikoita.

Tavoitelähtöisen lähestymistavan (Chung ym. 2000; Dardenne ym. 1993; Lapouchnian, 2005) mukaan tavoitteita voidaan kategorisoida monella tavalla esimerkiksi toiminnallisiin ja ei-toiminnallisiin tavoitteisiin, ohjelmiston kehitysprosessia kuvaaviin tavoitteisiin, ohjelmiston laatua kuvaaviin tavoitteisiin ja myös muihin kategorioihin. Käytettävyyttä koskevat tavoitteet nähdään kirjallisuuden pohjalta periytyvän ihmisen-teknologia-vuorovaikutustavoitteista, käyttöliittymätavoitteista ja edelleen ei-toiminnallisista laatutavoitteista (Lamsweerde, 2009). Muilta osin käytettävyyttä, käyttöliittymää ja vuorovaikutusta koskevat tavoitteet, vaatimukset ja niiden käsittely ohitetaan tavoitepohjaisuutta koskevassa keskeisessä vaatimusmäärittelykirjallisuudessa (esim. Chung ym. 2000) lähes täysin. Käytettävyyteen, käyttöliittymään ja vuorovaikutukseen liittyvät vaatimukset ovat sisällöltään hyvin erityyppisiä verrattuna muihin ei-toiminnallisiin vaatimuksiin (esimerkiksi tietoturva-vaatimukset ja suorituskykyvaatimukset). Juuri käytettävyyksivaatimusten sisältö muodostaa toisen keskeisen haasteen niiden määrittämiselle.

Haasteena käytettävyys- ja käyttäjäkokemusvaatimusten sisällön analyysi

Toinen käytettävyysvaatimusten määrittelyn keskeinen haaste liittyy niiden sisällön analyysiin. Käytettävyysvaatimukset sisältävät useita psykologian näkökulmasta keskeisiä käsitteitä liittyen esimerkiksi opittavuuteen, muistettavuuteen, havaitsemiseen, tarkkaavaisuuteen ja päätöksentekoon. Psykologisen tietämyksen soveltaminen osana käytännön ohjelmistokehitystä on koettu kuitenkin vaikeaksi erityisesti ohjelmiston kehitysprosessin alkuvaiheissa (Carroll, 1997; Olson & Olson, 2003). Esimerkiksi käytettävyysvaatimusten määrittämistä (Gould & Lewis, 1985; Mayhew, 1999; Nielsen, 1993; Wixon & Wilson, 1997) ja käytettävyysattribuuttien sisältöä (Abran ym. 2003; Bevan, 1995; Seffah ym. 2006) käsittelevä keskeinen kirjallisuus ohittaa kokonaan käytettävyyden keskeisten käsitteiden analyysin tieteellisen psykologian näkökulmasta. Yksi esimerkki tästä on työmuisti (engl. working memory). Työmuistista on tehty paljon tieteellistä tutkimusta (Baddeley, 2007; Baddeley, 2002; Caplan ym. 2008; Kane ym. 2007; Owen ym. 2005; Tulving & Craik, 2000), joka tulisi huomioida käsiteltäessä käytettävyysvaatimuksia tai käytettävyysattributteja. Sama ongelma koskee käyttäjäkokemusta koskevien vaatimusten määrittämistä. Esimerkiksi Preece (2005) ei huomio osana vuorovaikutussuunnittelun prosessia käyttäjäkokemusattribuuttien (esim. hauskuus, motivaatio, nautinnollisuus, viihdyttävyyys) sisältöä modernin psykologian tutkimustiedon valossa.

Hauskuus (engl. fun) on hyvä esimerkki käyttäjäkokemuksen attribuutista. Read ym. (2001, 2002) ovat selvittäneet, että 5-10 vuotiaiden lasten tapauksessa käytettävyys ja hauskuus ovat yhteydessä toisiinsa kolmesta näkökulmasta: odotukset, sitoutuminen (engl. engagement) ja kestävyys (engl. endurability). Odotuksilla viitataan ennako-odotuksiin ja toteutuneisiin odotuksiin. Esimerkiksi, jos käyttäjällä on korkeat odotukset tiettyä tapahtumaa kohtaan, mutta tapahtuma osoittautuukin tylsäksi, hän tuntee pettymystä. Odotusten ollessa matalat saattaa sen sijaan tyydyttäväkin tapahtuma tuntua hyvältä. Sitoutuminen viittaa positiivisiin ja negatiivisiin hauskuuden ilmenemismuotoihin. 5-10 vuotiailla lapsilla positiivisia ilmenemismuotoja ovat hymyileminen, nauraminen, keskittyminen (esim. sormi suussa) tai positiivinen ilmaiseminen. Negatiivisia ilmenemismuotoja ovat otsan rypistäminen, tylsyyden eleet (esim. sormien hypistely), olankohautus tai negatiivinen ilmaiseminen. Kestokyvyllä (engl. endurability) viitataan kahteen näkökulmaan: muistelu ja tavoitettavuus. Muistelu viittaa Pollyannan periaatteeseen, eli todennäköisyyteen muistaa asiat, joista olemme aiemmin nauttineet. Tavoitettavuus tarkoittaa halua suorittaa hauska toiminta uudelleen. Tämä on yksi esimerkki siitä, millä tasolla yksittäinen käyttäjäkokemuksen attribuutti tulee vähintään ymmärtää määriteltäessä käyttäjäkokemusta koskevia vaatimuksia. Vastaava psykologinen sisältö on tietenkin analysoitava uudelleen kehitettävän sovelluksen kohderyhmän vaihtuessa. Sama koskee käytettävyyden attribuutteja ja käytettävyysvaatimuksia. Käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten sisältö tulisi aina pystyä

analysoimaan tieteellisen psykologian näkökulmasta välttämättä arkitietämystä ja arkipsykologiaa (Saariluoma & Oulasvirta, 2010). Arkipsykologialla viitataan esimerkiksi siihen, että käyttäjään liittyviä suunnitteluongelmia ratkaistaan arki-intuitioiden ja introspektioon eli itsensä tarkasteluun perustuvan ihmiskäsityksen pohjalta (Saariluoma 2011). Tällainen lähestymistapa psykologiaan on yleistä ohjelmistokehityksen alalla, jossa asiantuntijoilla ei tyypillisesti ole ihmistieteellistä koulutusta. Arkitietämykseen ja arkipsykologiaan perustuva tieto ei ole välttämättä virheellistä, mutta ongelman muodostaa se, että emme myöskään tiedä onko se totta (Saariluoma, 2004). Arkipsykologian perustuessa tieteellisen psykologian teorioiden sijasta muun muassa yksittäisten ihmisten omiin käsityksiin sen ulkopuolelle jäävät esimerkiksi alitajunta, aivojen toimintaprosessit ja kulttuurisidonnaisuus (Saariluoma, 2011). Toisaalta on huomioitava, että ei tieteessäkään kaikki ole täysin oikeaa ja varmaa vaan pikemminkin totuuden kaltaista (Niiniluoto, 1987). Esimerkiksi käyttäjäpsykologian soveltaminen osana vaatimusmäärittelyä tai käyttöliittymäsuunnittelua lisää varmuutta ihmisen käyttäytymisen ymmärtämisestä. Varmuuden lisäämisen tarkoitus ei ole kuitenkaan eliminoida täysin arkitietämykseen tai arki-intuitioihin perustuvia oivalluksia esimerkiksi osana käyttöliittymäsuunnittelua.

Käytettävyyden ja käyttäjäkokemuksen attribuutteja, kuten muistettavuus, opittavuus, hauskuus, nautinnollisuus ja mielihyvä on analysoitu lukuisissa tutkimuksissa psykologian näkökulmasta (esim. Baddeley, 2002; Blythe ym. 2003; Tulving & Craik, 2000). Osaa näistä käsitteistä on tarkasteltu myös nimenomaan vaatimusmäärittelyn näkökulmasta. Esimerkiksi Draper (1999) on analysoinut hauskuuden käsitettä vaatimusmäärittelyn näkökulmasta, Callele ym. (2006) ovat analysoineet emotionaalaisia vaatimuksia videopelien näkökulmasta, Stelmaszewska ym. (2004) ovat yhdistäneet hedoniset (nautintoa tuottavat) piirteet ja vaatimusten määrittelyn, ja Bentley ym. (2002) ovat tutkineet yleisemmällä tasolla vaatimusten määrittelyn ja affektiivisten vaatimusten yhteyttä. Seuraava esimerkki havainnollistaa kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen käytettävyyden- ja käyttäjäkokemusvaatimusten määrittelyä psykologian näkökulmasta Angry Birds -pelin avulla. Esimerkki hyödyntää kyseisestä pelisovelluksesta laadittua kognitiiviseen psykologiaan painottuvaa käytettävyysoanalyysia (Mauro, 2011).

Käytettävyyden ja käyttäjäkokemuksen näkökulmasta Angry Birds -pelin suunnittelussa on huomioitu poikkeuksellisen onnistuneesti ihmisen kognitiivisten prosessien (esim. havaitseminen, tarkkaavaisuus, muistaminen) rajallinen kapasiteetti. Ensimmäiseksi voidaan todeta, että pelisovelluksen käyttöliittymä on yksinkertainen, koukuttava ja helposti opittava. Helppokäyttöisen käyttöliittymän suunnittelu ei ole välttämättä vaikeaa, mutta haasteeksi nousee käyttäjien sitouttaminen pidemmällä aikavälillä yksinkertaisiin suunnitteluratkaisuihin. Tällöin käytettävyyden ei yksin riitä täyttämään vaatimusmäärittelylle asetettuja tavoitteita, koska vasta käyttäjäkokemus sitouttaa käyttäjät pysyvästi sovelluksen pariin. Angry Birds -pelin käyttöliittymä tukee myös pelaajan pelistrategioihin liittyvää päätöksentekoa hallitsemalla nokkelasti vasteaikaa. Tällä

tarkoitetaan sitä, että kaiken ei tarvitse tapahtua käyttöliittymällä mahdollisimman nopeasti. Pelissä lintujen lentäminen on tehty tarkoituksella hitaaksi ja lentokaaria havainnollistetaan graafisen lentoradan avulla. Pelissä on vaiheita, jossa käyttäjä joutuu odottamaan jopa 3-5 sekuntia sitä, kaatuuko jokin horjuva seinä, lattia tai katto vai ei. Keskeinen ajatus normaalia pidemmille vasteajoille on se, että vasteaika mahdollistaa pelaajien virheenkorjausprosessin antamalla aikaa miettiä uutta pelistrategiaa (tarvitaanko lentokaareen enemmän nopeutta, vähemmän kulmaa jne.). Vasteaikojen määrittely vaatimusmäärittelyn tasolla on siis tarpeellista myös kvantitatiivisten käytettävyyksivaatimusten tarkkuudella. On tiedettävä, montako sekuntia on sopiva vasteaika, joka antaa ihmiselle tarpeeksi aikaa uuden strategian luontiin mutta ei aiheuta pitkästymistä. Kvantitatiivisesti määritelty käytettävyyksivaatimus helpottaa myös vaatimuksen todentamista myöhemmässä vaiheessa prosessia kun käytettävyyttä testataan. Vasteajan määrittämisen onnistumista helpottaa ihmisen päätöksentekoon vaikuttavien kognitiotieteellisten peruslainsuhteiden tuntemus ja hallinta. Edellä mainittujen ominaisuuksien (yksinkertainen ja koukuttava käyttöliittymä, vasteajan hallinta) lisäksi pelin käyttöliittymä huomioi ihmisen työmuistin rajallisen kapasiteetin soveltaen sitä sekä pelin helpottamiseen että vaikeuttamiseen. Työmuistin kapasiteetin ollessa rajoitettu on käyttöliittymäsuunnittelussa luonnollista pyrkiä suunnitteluratkaisuihin, jotka eivät ylikuormita työmuistia. Anry Birds -peli kuitenkin haastaa pelaajan työmuistin edellyttämällä tietyissä määrin kunkin peliradan tai tason ulkoa muistamista. Tämä tehdään rullaamalla jokaisen pelitason alussa pelirata oikealta vasemmalla siten, että keskeinen osa peliradasta jätetään käyttäjän ulkomuistin varaan. Tämän jälkeen pelaajan tarkkaavaisuus kohdistetaan muihin yksityiskohtiin esimerkiksi lintujen liikkeiden ja musiikin avulla. Käyttäjälle annetaan kuitenkin mahdollisuus tarkastella tarvittaessa uudelleen pelirataa ja palauttaa mieleen tärkeitä yksityiskohtia, jotta mielenkiinto pysyy yllä. Pelissä on muitakin vastaavia piirteitä, jotka tekevät pelaamisesta tarpeeksi haastavan ja koukuttavan kokemuksen käyttäjille. Pelaajien on pystyttävä luomaan itselleen sopivia strategioita myös kognitiivisten prosessien näkökulmasta selvittääkseen peliradoista hyvillä pisteillä. Pelikokemus ei siis rajoitu vain pelisovellukseen vaan pyrkii koskettamaan myös ihmistä yksilönä. Nämä esimerkit ovat vain hyvin pieni osa havainnollistamaan seikkoja, jotka yhdistävät ihmisen käyttäytymisen taustalla olevat perusprosessit kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vaatimusmäärittelyyn. Tämän väitöskirjan ulkopuolelle jätetään ihmisen kognitiivisten prosessien, kuten havaitseminen, tarkkaavaisuus ja muistaminen perusteellinen läpikäynti (katso lisää kognitiivisista prosesseista esim. Baddeley, 1997; Goldstein, 1999; Pashler, 1998).

Esimerkissä havainnollistettu käytettävyyden ja käyttäjäkokemuksen analysointi osana vaatimusmäärittelyä on vielä nykyäänkin harvinaista. Puutteita on havaittu sekä vaatimusmäärittelymenetelmissä että attribuuttien psykologiaan perustuvassa analysoinnissa. Vaikka käyttäjäkokemuksen sisällön analyysiin liittyvää tutkimusta on tehty paljon, ei ole olemassa esimerkiksi

selkeää prosessia, jonka avulla käyttäjäkokemus saataisiin yhdistettyä osaksi vaatimusmäärittelyä. (kts. Alves ym. 2007; Stelmaszewska ym. 2004).

Haasteena käytettävyyden ja käyttäjäkokemuksen arviointi

Kolmas käytettävyyksvaatimusten määrittelyn keskeinen haaste liittyy käytettävyyden arviointiin. Ensimmäisen haasteen yhteydessä käsiteltiin käytettävyyksvaatimuksia eri abstraktitasojen näkökulmista. Samassa yhteydessä selvitettiin, että kirjallisuuden perusteella käytännön ohjelmistokehitystyössä käytettävyyksvaatimukset jätetään usein kvalitatiivisten käytettävyyksvaatimusten tasolle. Tästä seuraa ongelmia käytettävyyden arvioinnille, sillä korkean abstraktitason vaatimusten arviointi/mittaaminen/testaaminen on vaikeaa. Miten esimerkiksi voidaan yksiselitteisesti varmistaa, että vaatimukset "Ohjelmistossa tulee olla nykyaikainen ja käyttäjäystävällinen käyttöliittymä" ja "Järjestelmän käytön oppiminen tulee olla helppoa" toteutuvat? Entä kuinka voidaan paikantaa selvästi käytettävyyksongelmat, jotka estävät juuri näiden vaatimusten toteutumisen? Tämän ongelman voidaan nähdä johtaneen tilanteeseen, jossa uudentyypisiä käytettävyyden arviointimenetelmiä (esim. Ivory & Hearst, 2001) on jo pitkään kehitetty suuntaan, jossa käytettävyyksvaatimukset eivät ole enää käytettävyyden arvioinnin lähtökohtana. Esimerkiksi useat automaattiset käytettävyyden arviointimenetelmät (Ivory & Hearst, 2001) pyrkivät keräämään, analysoimaan ja esittämään käytettävyyden ratkaisuehdotuksia enemmän tai vähemmän automatisoidusti käyttäen datanaan mahdollisimman suurta määrää vuorovaikutusta koskevaa dataa. Näiden menetelmien voidaan nähdä ohittavan käytettävyyksvaatimusten käsitteen lähes kokonaan. Nämä menetelmät eivät myöskään pysty paikantamaan ja käsittelemään käytettävyyksongelmia tarvittavalla tarkkuustasolla (esim. missä käyttöliittymäkomponentissa on käytettävyyksongelma, miksi se on ongelma, mistä ongelma johtuu). Sen sijaan kyseiset menetelmät esittävät tyypillisesti mittaustulokset, joiden tulkinta jätetään käytettävyyksarvioijan ammattitaidon varaan.

Kuten aiemmissa luvuissa selvitettiin, vuorovaikutussuunnittelu, käyttäjäkeskeinen suunnittelu ja vaatimusmäärittely perustuvat karkeasti ottaen ajatukseen määrittellä tarkoituksenmukaisia vaatimuksia kehitettävälle ohjelmistolle ja arvioida/testata vaatimusten toteutumista kehitysprosessin myöhemmässä vaiheessa (Good ym. 1986; Rubin, 1994; Tyldesley, 1988; Wixon & Wilson, 1997). Tämä perusperiaate kuitenkin unohdetaan usein kehitettäessä uusia käytettävyyden arviointimenetelmiä. Esimerkiksi tehtävämalleihin (engl. task model-based) perustuvat automatisoidut käytettävyyden arviointimenetelmät (esim. Paganelli & Paternò, 2002; Paternò & Ballardin, 1999; Paternò & Ballardin, 2000) keskittyvät enemmän yksittäisiin tehtäviin, niiden sisältämiin alitehtäviin ja koehenkilöiden välisiin eroihin (esim. suoritusaikojen ja virheiden suhteen) kuin ennalta määriteltyjen käytettävyyks- ja käyttäjäkokemuksvaatimusten toteutumisen arviointiin. Toisaalta tehtävämallien näkökulmasta tätä voidaan perustella esimerkiksi sillä, että tehtävämallit

voidaan jo itsessään ottaa huolellisesti laaditusta vaatimusmäärittelydokumentista (esim. Propp ym. 2009). Kuitenkin myös yksinkertaisemmat ja suoraviivaisemmat käytettävyyden arviointimenetelmät, kuten esimerkiksi heuristiset arviointimenetelmät (Desurvire ym. 2004; Nielsen & Mack, 1994; Nielsen & Molich, 1990; Pinelle ym. 2008) ohittavat käytettävyyksivaatimuksen käsitteen pyrkien ensisijaisesti paikallistamaan mahdollisimman suuren määrän käytettävyysongelmia. Edelleen tarkasteltaessa erityisesti sähköisille kuluttajatuotteille (engl. consumer electronic products) kehitettyjä käytettävyyden arviointimenetelmiä (Kim & Han, 2008) nähdään, että nekin ohittavat käytettävyyksivaatimukset kokonaan keskittyen kuluttajatuotteille ominaisten käytettävyyksattribuuttien mukaiseen käytettävyyden arviointiin. Kim ja Han (2008) ovat todenneet perinteisten käytettävyyden arviointimenetelmien olevan riittämättömiä sähköisten kuluttajatuotteiden (kuten esim. DVD-soitin) käytettävyyden arviointiin kolmen eri kriteerin perusteella: 1) Perinteiset käytettävyyden arviointimenetelmät keskittyvät liikaa suorituskykykeskeisten (engl. performance) yksityiskohtien arviointiin, eivätkä huomioi käytettävyyttä suurempana kokonaisuutena (esimerkiksi käytöstä syntynyt kokonaisvaikutelma). 2) Perinteiset käytettävyyden arviointimenetelmät vaativat käytettävyyksiantuntijan osallistumista prosessiin, vaikka tämä ei ole aina mahdollista esimerkiksi yritysten asenteista ja prioriteeteista johtuen. 3) Perinteiset käytettävyyden arviointimenetelmät eivät anna selkeää kokonaiskuvaa siitä, mikä on käytettävyyden taso.

Edellä olevia (Kim & Han, 2008) kriteereitä voidaan kritisoida seuraavalla tavalla. Käytettävyyden arvioinnin tulisi keskittyä ensisijaisesti määritelyjen käytettävyyksivaatimusten mukaiseen arviointiin. Esimerkiksi suorituskykykeskeinen käytettävyyden arviointi (kriteeri 1) on tarpeellista silloin, kun sitä kohtaan on määritelty selkeät käytettävyyteen liittyvät vaatimukset. Toki myös kokonaisvaikutelmaan, mieltymyksiin ja vastaaviin seikkoihin kohdistuvat käytettävyyden mittaukset ovat yhtä tärkeitä, mutta tällaisten vaatimusten voidaan nähdä liittyvän enemmän käyttäjäkokemusvaatimukseen. Osana käytettävyyksivaatimusten määrittelyä, käytettävyyksuunnittelua ja käytettävyyden arviointia tulee olla ammattihenkilö, jolla on teknisen koulutuksen lisäksi myös ihmistieteellistä (esim. kognitiotiede, käyttäytymistieteet, psykologia) koulutusta (kriteeri 2). Käytettävyyden ja käyttäjäkokemuksen arviointi vaatii teknologisen tietämyksen lisäksi käyttäjäpsykologista tietämystä. Tällä viittaaan esimerkiksi tietämykseen ihmisistä laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Ilman tällaista ammattitaitoa vaarana on arkipsykologiaan ja arkitietämykseen perustuva vuorovaikutuksen analyysi (Saariluoma, 2011). Kuten edellä mainittiin, arkipsykologiaan ja arkitietämykseen perustuva vuorovaikutuksen analyysi ei välttämättä tuota virheellisiä tuloksia, mutta ongelmaksi muodostuu se, että myöskään tulosten oikeellisuutta ei voida todistaa (Saariluoma, 2004). Toisaalta Kim ja Han (2008) eivät huomioi kehittämässään käytettävyyden arviointimenetelmässä esimerkiksi käsitteiden muistettavuus ja opittavuus analyysiä modernin psykologian näkökulmasta. Käytettävyyksiantuntijan roolista ohjelmiston kehitysprojek-

teissa on tehty myös empiirisiä tutkimuksia (Heiskari ym. 2009), jotka tukevat tätä näkemystä. Kolmatta kriteeriä (Kim & Han, 2008) voidaan kritisoida siten, että perinteiset käytettävyyden arviointimenetelmät antavat hyvän kokonaiskuvan käytettävyydestä, mikäli asianmukaiset käytettävyyttä ja käyttäjäkokemusta koskevat vaatimukset on määritelty ja niiden toteutuminen arvioidaan määriteltyjen tavoitteiden perusteella. Kritiikki, jonka Kim ja Han (2008) ovat esittäneet, ei huomioi käyttäjäkokemuksen käsitettä vaan pyrkii laajentamaan käytettävyyden määritelmää käyttäjäkokemuksen suuntaan. Kim ja Han (2008) perustelevat näkemystään käytettävyyteen sillä, että erityisesti kuluttajatuotteiden tapauksessa käytettävyyden tulisi huomioida perinteisten käytettävyyden attribuuttien lisäksi yksilön kokemuksesta nousevia attribuutteja, kuten esimerkiksi mielikuvat, ennako-odotukset ja vaikutelmat (Han ym. 2001; Kwahk & Han, 2002). Itse asiassa juuri tämän tyyppinen näkökulma on yksi tyyppinen tapa määritellä käyttäjäkokemus (Bevan, 2009b), kuten aiemmin tässä luvussa selvitimme. Tässä tutkielmassa kehittävä intentiolähtöinen lähestymistapa käytettävyyden ja käyttäjäkokemusvaatimusten määrittelyyn huomioi sekä käytettävyyden että käyttäjäkokemuksen näkökulman.

Verrattuna uudemman sukupolven käytettävyyden arviointimenetelmiin, perinteisemmät menetelmät, kuten käytettävyydestaus (Rubin, 1994) tuovat selvästi esille käytettävyyssarvioinnin ensisijaisen tavoitteen selvittää, missä määrin testattava tuote täyttää sille asetetut kvantitatiiviset käytettävyyssvaatimukset. Myös käytettävyyssuunnittelu korostaa kvantitatiivisten käytettävyyssvaatimusten määrittämistä ja niiden mukaista arviointia (Tyldesley, 1988; Wixon & Wilson, 1997). Esimerkiksi Good ym. (1986) ovat todenneet, että ”Ilman mitattavia käytettävyyssvaatimuksia emme voi määrittää perusteellisesti kehitettävän tuotteen käytettävyyttä tai mitata, täyttääkö valmis tuote sille asetetut vaatimukset käytettävyyden suhteen. Jos emme voi mitata käytettävyyttä, emme voi edes puhua käytettävyyssuunnittelusta.”

Kuitenkin monien tutkimusten perusteella voidaan nähdä, että tarkasti määriteltyjen mitattavien vaatimusten määrittäminen on erityisen haasteellista tilanteessa, jossa kehitetään innovatiivista kuluttajamarkkinoille suunnattua ohjelmistotuotetta (Aaen, 2008; Cassab ym. 2010; Dearden & Howard, 1998; Maiden & Robertson, 2005a, 2005b; Mayhew, 1999; Tuunanen ym. 2008). Tarkasteltaessa tätä haastetta käyttäjäkokemuksen näkökulmasta kohdataan edelleen monimutkaisempia haasteita. Käyttäjäkokemus on luonteeltaan subjektiivinen käsite (Hassenzahl ym. 2010; Obrist ym. 2009), käyttäjäkokemusta koskevat vaatimukset ovat subjektiivisia (Stone, 2005) ja käyttäjäkokemusta mitataan tyyppillisesti käyttäen subjektiivisia mittareita (Hassenzahl, 2008; Väänänen-Vainio-Mattila ym. 2008). Tällöin käyttäjäkokemusta ei voida arvioida yhtä suoraviivaisesti kuin määrällisiä käytettävyyssvaatimuksia (esim. klikkausten määrä, virheiden määrä, oppimiseen tarvittut toistot jne.) (Väänänen-Vainio-Mattila ym. 2008). Lopulta käsitteiden käytettävyyden ja käyttäjäkokemuksen ero on kuitenkin hyvin epäselvä ja monella tavalla tulkittavissa niin vaatimusmäärittely- kuin arviointimenetelmienkin suhteen (katso esim. Bevan, 2009a).

Käytettävyyden arviointiin tulisi olla selkeä vaihe vaiheelta etenevä suhteellisen kevyt menetelmä, jonka avulla käytettävyysongelmat voidaan paikantaa täsmällisesti, esimerkiksi yksittäisen käyttöliittymäkomponentin tarkkuudella, vaikka käytettävyyksvaatimuksia ei olisi määritelty matalan abstraktitason tarkkuudella. Tarkka käytettävyysongelmien paikantaminen mahdollistaisi myös ongelmien käyttäjäpsykologisen analyysin. Tällainen menetelmä olisi erityisen hyödyllinen kehitettäessä innovatiivista kuluttajamarkkinoille suunnattua ohjelmistotuotetta etenkin jos sen avulla voitaisiin käyttöliittymän prototyyppien käytettävyyssarviointin yhteydessä määrittellä kriittisiä rajoja käytettävyyksvaatimuksille. Tällöin sen avulla voitaisiin pyrkiä määrittämään mitattavia käytettävyyksvaatimuksia prototyyppien käytettävyyssarviointien yhteydessä.

2.6 Yhteenveto ja vastaamattomat kysymykset

Tässä väitöskirjassa keskitytään käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyyn tilanteessa, jossa kehitetään innovatiivista kuluttajamarkkinoille suunnattua ohjelmistotuotetta. Ohessa esitetään tiivistetysti tämän luvun pohjalta esille nousseet keskeiset haasteet. Tarkat tutkimuskysymykset ja ongelmat esitetään luvussa 4.

Keskeisen kirjallisuuden perusteella kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen kehittäminen asettaa vaatimusten määrittelylle kolme päähaastetta: 1) Tarve laajentaa perinteistä käyttäjänäkökulmaa kuluttajakeskeiseen suuntaan (Tuunanen ym. 2008). 2) Tarve keskittyä perinteiseksi koetun käytettävyyden tehokkuus-tuottavuus-näkökulman sijasta enemmän innovatiivisen ohjelmistotuotteen tuomiin hedonisiin piirteisiin, kuten nautinnollisuuteen ja mielihyvään (Kim & Han, 2008; Kwahk & Han, 2002; Rossi & Tuunanen 2010, Tuunanen ym. 2008) sekä käyttäjien tuotteen käyttöä koskeviin kokemuksiin ja vaikutelmiin (Han ym. 2001). 3) Tarve kehittää kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vaatimusmäärittelyä vastaamalla seuraaviin ongelmiin: a) Tulevien käyttäjien on vaikea ilmaista ja visioida täysin uudentyyppiseen ohjelmistotuotteeseen liittyviä tarpeita ja vaatimuksia (Casab ym. 2010). b) Kehitettäessä täysin uudentyyppistä ja innovatiivista ohjelmistotuotetta ei välttämättä edes tiedetä, keitä loppukäyttäjät ovat ja mihin tarkoitukseen he tulevat sovellusta käyttämään (Mayhew, 1999). c) Vastaavaa tuotetta, jota voitaisiin käyttää vaatimusten määrittelyn lähtökohtana, ei tyypillisesti ole olemassa (Dearden & Howard, 1998). d) Tällaisessa suunnittelutilanteessa ei osata ennakoita varmuudella, kuinka uusi tuote vaikuttaa loppukäyttäjien käyttäytymiseen, kuinka kilpailukykyinen tuote todellisuudessa on, ja kenelle se itse asiassa tuo hyötyä.

Edelleen, käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittely asettaa innovatiivisen kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämiseksi haasteita seuraavilla osa-alueilla: 1) kvantitatiivisten käytettävyyksvaatimusten määrittäminen, 2) käytettävyyks- ja käyttäjäkokemusvaatimusten sisällön analyysi, 3) selkeä prosessi, joka yhdistää käytettävyyks- ja

käyttäjäkokemusvaatimukset osana ohjelmistokehitystä ja vaatimusmäärittelyä ja 4) korkean abstraktitason käytettävyyksivaatimusten aiheuttamat ongelmat käytettävyyden arvioinnille.

Käyttäjakeskeisen suunnittelun periaatteita (ISO 13407, 1999) hyödyntävä vuorovaikutussuunnittelu (Preece, 2002) antaa hyvät lähtökohdat kuluttajamarkkinoille suunnatun ohjelmistotuotteen kehittämiseksi, koska se pyrkii huomioimaan käyttäjän useista eri näkökulmista ja yhdistää käyttäjäkokemuksen näkökulman käytettävyyteen jo vaatimusmäärittelyn tasolla. Käyttäjakeskeinen suunnittelu ei kuitenkaan sovellu parhaalla mahdollisella tavalla innovatiivisen teknologian suunnitteluun esimerkiksi käyttäjätarpeiden tunnistamisen näkökulmasta (katso esim. Leikas, 2009; Parkkola, 2006; Viukari, 2010). Tässä väitöskirjassa käyttäjätarpeiden tunnistaminen nähdään osana vaatimusten määrittelyprosessia. Kehitettäessä innovatiivista ohjelmistotuotetta tulisi ihmistä tarkastella laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Lisäksi vaatimusten määrittelyssä tulisi pyrkiä mahdollisimman teknologiariippumattomaan lähestymistapaan, jossa tunnistetut käyttäjätarpeet pidetään käytettävyyks- ja käyttäjäkokemusvaatimusten perustana läpi ohjelmistotuotteen suunnitteluprosessin. Tällöin tulee välttää tilannetta, jossa teknologia lähtee ohjaamaan suunnitteluprosessia käyttäjätarpeiden sijasta (Bergvall-Kåreborn & Ståhlbröst, 2010; Patnaik & Becker, 1999; Vredenburg ym. 2002). Tässä väitöskirjassa pyritään varmistamaan, että käyttäjätarpeet ohjaavat aidosti käytettävyyks- ja käyttäjäkokemusvaatimusten määrittelyä kehitettäessä innovatiivista ohjelmistotuotetta. Tällaista lopputulosta tavoitellaan ottamalla käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006) käyttäjätarkastelun lähtökohdaksi ja intentio vaatimusmäärittelyn lähtökohdaksi. Käyttäjäpsykologia ja intentio tarjoavat lupaavan lähestymistavan vastata edellä esitettyihin vaatimusmäärittelyn haasteisiin. Perinteiset vaatimusmäärittelyn lähestymistavat, menetelmät ja tekniikat eivät tarjoa selkeitä vastauksia esitettyihin haasteisiin. Seuraavassa luvussa esitellään intentiolähtöinen lähestymistapa vuorovaikutussuunnittelun vaatimusten määrittelyn tueksi.

3 INTENTIO-LÄHTÖINEN VUOROVAIKUTUSSUUNNITTELU

Informaatioteknologian alalla tietämystä intentiosta on hyödynnetty pääosin olemassa olevan teknologian hyväksymiseen, käyttöönottoon ja käyttämiseen liittyen. Huomattavasti vähemmän tutkimusta on siitä, kuinka intentiotietämystä voitaisiin hyödyntää käyttäjätarpeiden tunnistamisessa ja vaatimusten määrittelyssä. Intention ollessa ihmisen toimintaa ohjaava tekijä on luonnollista pyrkiä sisällyttämään intentiota koskeva tietämys myös vuorovaikutussuunnittelun lähtökohdaksi. Erityisen hyödyllistä tämä on tilanteissa, joissa ollaan kehittämässä innovatiivista ohjelmistotuotetta.

Tämän luvun tarkoituksena on rakentaa käsitystä intentiolähtöisestä vuorovaikutussuunnittelusta. Luvussa perehdytään intention käsitteeseen sekä aiheen kannalta relevantteihin teorioihin. Tällaisia teorioita ovat fenomenologia, hajautettu kognitio, toiminnan teoria, kognitiivinen ja hierarkkinen tehtävänälyysi sekä käyttäjäpsykologia. Teorioiden yhteydessä selvitetään kyseisten tutkimusalueiden avoimia kysymyksiä heijastaen edelleen tämän tutkielman vaikutuksia niihin. Luvun lopussa kuvataan intentiolähtöinen lähestymistapa vuorovaikutussuunnitteluun.

3.1 Intentionio

Intention käsitteellä on syvät juuret filosofian ja psykologian tutkimuksessa yhtenä ihmisen käyttäytymistä ja toimintaa selittävänä tekijänä (Ajzen, 1985; Fishbein & Ajzen, 1975; Bratman, 1987; Gollwitzer & Bargh, 1996; Hampshire, 1965; Irwin, 1971; James, 1890; Lewin, 1951; Meiland, 1970; Miller ym. 1960; Shultz, 1980; Tolman, 1932). Käsitteen tarkka määrittely on vaikeaa, koska sitä on käytetty hyvin laajasti soveltaen eri tieteenalojen ja koulukuntien mukaan. Voidaan kuitenkin sanoa, että *intentionilla* viitataan tyypillisesti siihen mentaaliseen tilaan, joka ohjaa ja kontrolloi ihmisen toimintaa ja käyttäytymistä (Brentano,

1874/1973; Shultz, 1980). Esimerkiksi, ”Mikko aikoo avata verhot, jotta voi pestä ikkunan” on intentio, joka saattaa ohjata Mikon avaamaan verhot.

Ihmisellä on mielessään lukuisia intentioita, joiden mukaan hän voi toimia ja käyttäytyä. On kuitenkin selvää, että ihminen ei toteuta jokaista mielessään olevaa intentiota (Malle & Knobe, 1997). Esimerkiksi edellä mainittu intentio voi jäädä toteuttamatta, jos Mikko tekeekin valinnan lähteä kuntosalille ikkunan pesun sijasta. Joidenkin tutkimusten mukaan (esim. Miller ym. 1960) voidaan sanoa, että motiiviin toimia tietyllä tavalla vaikuttavat intentioiden lisäksi arvot, jotka määräävät käytännössä toteutettavat intentiot. Toiminnan psykologian (Gollwitzer & Bargh, 1996) näkökulmasta puolestaan motivaatio nähdään keskeisenä käsitteenä, kun pyritään ymmärtämään ihmistä toimijana, joka tavoittelee päämääriään. Tästä näkökulmasta motivaatiota on tutkittu esimerkiksi sitä edistävien ja haittaavien tekijöiden näkökulmasta. Esimerkiksi ”control over outcomes” (Rotter, 1966), ”response-outcome dependence” (Seligman, 1975), ”self-efficacy” (Bandura, 1977) ja ”a high expectancy-valence product” (Vroom, 1964) ovat edellytyksiä korkealle motivaatiolle. Myöhemmin motivaatiota on pyritty ymmärtämään syvällisemmin esimerkiksi itsensä määräämismotivaatioteorian (engl. Self-Determination Theory) (Deci ym. 1989) kautta tutkimalla sitä, millainen vaikutus ihmisen toimintaan on sillä, kuinka toiminnan tavoitteet asettaa (vrt. sisäinen motivaatio vs. täysin kontrolloitu ja ulkoapäin ohjattu toiminta). Myös intentiolla itsessään viitataan joissakin tutkimuksissa (esim. Ajzen, 1991; Chen ym. 2002; Hasan, 2006; Warshaw & Davis, 1985) siihen määrään, suuruuteen tai innokkuuteen, jonka perusteella ihminen muodostaa suunnitelman joko toteuttaa tai olla toteuttamatta tietty toiminta.

Informaatioteknologian alalla tietämystä intentiosta on pyritty soveltamaan erityisesti teknologian hyväksymistä ja käyttöönottoa koskevan tutkimuksen yhteydessä (Agarwal & Karahanna, 2000; Cho, 2006; Szajna, 1996; Venkatesh ym. 2003; Yi & Hwang, 2003). Intention ja ihmisen toiminnan välinen suhde on määritelty tällöin tyypillisesti perustellun toiminnan teorian (TRA) (Fishbein & Ajzen, 1975), suunnitellun käyttäytymisen teorian (TBP) (Ajzen, 1985) tai teknologian hyväksymismallin (TAM) (Davis, 1989; Davis ym. 1989) mukaisesti. Perustellun toiminnan teoriaa pidetään yhtenä eniten sovelletuista intentio-toimintamallina tietojärjestelmätieteiden alalla (Kuo & Young, 2008). Perustellun toiminnan teoriassa, suunnitellun käyttäytymisen teoriassa ja teknologian hyväksymismallissa intentiolla viitataan nimenomaan siihen määrään, suuruuteen tai innokkuuteen, jonka perusteella ihminen muodostaa suunnitelman joko toteuttaa tai olla toteuttamatta tietty toiminta (Ajzen, 1991; Chen ym. 2002; Hasan, 2006; Warshaw & Davis, 1985). Intention käsitettä on käytetty myös prosessien mallintamisessa. Rolland ym. (1999) esittävät prosessimallin, jossa prosesseihin liitetään alku- ja loppuintentiot. Intentio nähdään tavoitteena, joka toimijalla on mielessään tiettyä hetkenä ja jonka hän haluaa saavuttaa. Sovellettaessa tätä menetelmäkehitykseen toimija ymmärretään sovellussuunnittelijaksi.

Ihmisen ja koneen välisen vuorovaikutustutkimuksen (HCI) näkökulmasta on kehitetty useita ihmisen toimintaa määrittäviä malleja, joissa intentio

esiintyy yhtenä keskeisenä ihmisen toimintaa ohjaavana tekijänä (esim. Card ym. 1983; Miller ym. 1960; Norman, 1986). Esimerkiksi syklinen interaktio (Monk, 1998) on lähestymistapa, jonka mukaan intentio, toiminta ja ympäristö ovat jatkuvassa syklisessä vuorovaikutussuhteessa. Tällöin ihmisen intentiot yhdessä ympäristön tilan havainnoinnin kanssa (ja myös ilman sitä) johtavat toimintoihin, jotka vaikuttavat uudelleen havaittavaan ympäristöön ja sen tilaan. Monkin (1998) mukaan intention kohdistuvaa tutkimusta tarvitaan kuitenkin selvästi lisää HCI-näkökulmasta, jotta ihmisen käyttäytymistä voitaisiin ymmärtää paremmin vuorovaikutustilanteessa. Osa intention ominaisista piirteistä tekee kuitenkin vaikeaksi intentiotietämyksen sisällyttämisen osaksi vuorovaikutussuunnittelua. Intention ominaispiirteet ovat seuraavat (Haasbroek, 1993): 1) Intentiot eivät ole suoraan mitattavissa (esimerkiksi koetilanteessa tarkkailija joutuu tyypillisesti päättämään intentiot). 2) Ihmisten on vaikea ilmaista sanoin intentioneitaan. 3) Intentiot voidaan yhdistää sekä toistaiseksi toteutumattomiin päämääriin (esimerkiksi suunnitelma) että päämääriin, jotka on jo saavutettu. 4) Intentionilla on tietty alkutila ja lopputila. 5) Intentiot ovat hierarkkisessa ja heterarkkisessa suhteessa toisiinsa. Kolme ensimmäistä intention ominaispiirrettä asettavat haasteita erityisesti vaatimusten määrittelyprosessia kohtaan.

3.2 Ihmisen toiminta ja vuorovaikutustutkimus

Tässä tutkielmassa on sivuttu jo useita kertoja ihmisen toiminnan käsitettä ja tuotu edelleen esille intention käsite nimenomaan ihmisen toimintaa ohjaavana tekijänä. Saariluoman (2011) mukaan ihmisen toiminta on luonnollinen näkökulma tarkastella ihminen-teknologia-vuorovaikutussuunnittelua, koska ihminen käyttää teknologiaa nimenomaan toiminnallisten päämääriensä saavuttamiseen. Toiminnan käsitteen tutkimuksella on pitkät historialliset perinteet alkaen Aristoteleen Nikomakhoksen etiikasta (1094) kulkien Spinozan (1674/1994), Goethen (1808/1832/1965) ja Fichten (1794-1795/2006) kautta edelleen Weberiin (1922), Schütziin (1932/2007) ja Parsonsiin (1949). Tässä tutkielmassa toiminnan käsitteestä ollaan kiinnostuneita ihminen-teknologia-vuorovaikutustutkimuksen näkökulmasta. Tästä näkökulmasta keskeisiä lähestymistapoja toiminnan tutkimukseen ovat Kuutin (2011) mukaan hajautettu kognitio (Hutchins, 1996), toiminnan teoria (Engeström ym. 1999) ja fenomenologinen lähestymistapa (Dourish, 2001). Seuraavassa käydään läpi lyhyesti hajautettu kognitio ja fenomenologinen lähestymistapa sekä selvitetään perusteellisemmin toiminnan teoria.

Fenomenologia on tieteenfilosofinen suuntaus, jonka keskeisinä kehittäjinä pidetään yleisesti sellaisia henkilöitä kuten Georg Hegel, Edmund Husserl ja Martin Heidegger (katso esim. Heidegger, 1927/1962; Husserl, 1962). Fenomenologia korostaa välittömän subjektiivisen kokemuksen merkitystä ja tutkii tietoisuutta ilmiönä tavoitteenaan selvittää todellisuuden ilmenemistä yksilölle hänen kokemusmaailmassaan (Giorgi, 1997). Filosofi Edmund Husserl korosti

intention merkitystä osana ihmisen toimintaa, sillä hänen mukaansa tietoisella toiminnalla on aina jokin kohde (kohde tässä sama kuin intentio), joka toimii tällöin ihmisen toimintaa ohjaavana tekijä (Husserl, 1900/1970, 1913/1983). Husserlin mukaan ihmisen tietoisuus on siis intentionaalista (Giorgi, 2005). Intention ja intentionaalisuuden käsitteen Husserl otti käyttöön oppi-isältään Franz Brentanolta (Brentano, 1874/1973; Giorgi, 2005). Fenomenologista lähestymistapaa on sovellettu edelleen osaksi ihminen-teknologia-vuorovaikutustutkimusta erityisesti ihmisen toiminnan näkökulmasta (esim. Dourish, 2001; Winograd & Flores, 1987). Kuutti (2011) mukaan erityisesti Paul Dourishin teos ”Where the Action Is: The Foundations of Embodied Interaction” nousi merkittäväksi työksi fenomenologian soveltamisen näkökulmasta tuoden käsitteen ”ruumiillisuus” (engl. embodiment) osaksi käytännön vuorovaikutustutkimusta. *Ruumiillisuus* viittaa siihen, miten fyysiset ja sosiaaliset elämän ilmiöt kehittyvät osana maailmaa, jossa olemme ja josta olemme osa (Kuutti, 2011).

Hajautettu kognitio viittaa lähestymistapaan, jossa kognitiota tarkastellaan yhden tai useamman ihmistoimijan ja materiaalisen ympäristön välille (esim. työkalut ja laitteet) hajaantuneena prosessina (Hutchins, 1996; Salomon, 1993). Hajautetun kognition käsite laajentaa perinteisen kognitiotieteen näkökulmaa yksilön mielen sisäisten kognitiivisten prosessien tutkimisesta monimutkaisten sosioteknisten järjestelmien suuntaan (Hollan ym. 2000; Hutchins, 1995). Sosioteknisellä järjestelmällä tarkoitetaan kontekstia, jossa useat ihmiset ja useat koneet (erityisesti tietokoneet) toimivat ikään kuin yhtenä suurena järjestelmänä osallistuen monimutkaisten toimintojen suorittamiseen (Hoffman & Militello, 2009). Tällaisia sosioteknisiä järjestelmiä ovat esimerkiksi laivojen komentosillat (Hutchins, 1996) ja lentokoneiden ohjaamot (Hutchins & Klausen, 1996; Hutchins & Palen, 1997).

Toiminnan teoria on filosofinen ja monitieteinen viitekehys ihmisen toiminnan tutkimiseen sekä yksilön tasolla että sosiaalisesta näkökulmasta (Kuutti, 1996). Toiminnan teoria on kehitetty alun perin Vygotskyn (1978) toimesta 1920-luvun Venäjällä. Vygotsky (1978) määritteli toiminnan kolmiomallin, jonka mukaan toiminta koostuu tekijästä, välineestä ja kohteesta, jotka ovat vuorovaikutuksessa keskenään. Keskeinen ajatus Vygotskin teoriassa oli idea välittyneestä toiminnasta (Kuutti, 2011), jonka mukaan inhimillinen toimija on suhteessa ympäristöönsä aina kulttuurillisten välineiden ja työkalujen sekä merkki- ja symbolijärjestelmien kautta. Vygotskin (1978) mukaan ihminen ei siis reagoi eläimen tavoin sisäsyntyisten refleksien kautta suoraan ulkoisiin ärsykkeisiin vaan ihmisen ja tekemisen välissä on aina jokin välittäjänä toimiva työkalu. Vygotskyn työtä toiminnan teorian kehittäjinä jatkoivat Venäjällä myöhemmin hänen oppilaansa Alexander Lurija ja A.N. Leontjev (esim. Leont'ev 1974, 1978, 1981, 1989). Leontev (1981) esitti esimerkiksi teorian siitä, että ihmisen toiminnalla on kolmitasoinen hierarkkinen rakenne, jonka muodostavat toiminnot (engl. activity), teot (engl. action) ja operaatiot (engl. operation). Tällöin (Bertelsen & Bødker, 2003) toiminnot (engl. activity) ovat ihmisen motivaatioon sidoksissa olevia toimintoja, jotka vastaavat ”miksi?”

kysymykseen. Teot ovat tavoitteisiin (engl. goal) sidoksissa olevia tekoja, jotka vastaavat "mitä?" kysymykseen. Operaatiot ovat tekojen edellytyksiä vastaten "miten?" kysymykseen.

Yrjö Engeström on kansainvälisesti tunnustettu toiminnan teorian tutkija, joka uudisti myöhemmin (Engeström, 1987) Vygotskin mallin toiminnan teoriasta laajentamalla sitä kattamaan tekijän, välineen ja kohteen lisäksi säännöt, yhteisön ja työnjaon sekä näiden väliset suhteet toisiinsa. Toisena merkittävänä toiminnan teorian tutkijana suomessa voidaan pitää Kari Kuuttia, joka on soveltanut toiminnan teoriaa HCI-näkökulmasta Engeströmin tutkimuksiin perustuen. Skandinaviassa on vahvat perinteet toiminnan teorian soveltamisesta HCI-alalle, sillä myös tanskalainen Susanne Bødker (1989, 1991) ja ruotsalaiset Victor Kaptelinin (1992) sekä Bonnie Nardi (1992, 1996) ovat tunnettuja tutkimuksistaan toiminnan teorian ja HCI-näkökulman rajapinnassa (katso myös Kaptelinin & Nardi, 2006; Kaptelinin ym. 1999). Kaptelinin ja Nardi (1997) ovat esimerkiksi määritelleet viisi keskeistä periaatetta, joiden mukaisesti toiminnan teoria selittää ihmisen toimintaa. Nämä periaatteet ovat toiminnan hierarkkinen rakenne, kohteellisuus (engl. object-orientedness), sisäistäminen/ulkoistaminen (engl. internalization/externalization), välittyminen (engl. mediation) ja kehitys (engl. development).

Toiminnan hierarkkisella rakenteella viitataan toimintoon (engl. activity), tekoon (engl. action) ja operaatioon (engl. operation) (Kaptelinin & Nardi, 1997). Kohteellisuudella viitataan siihen, että toiminta kohdistuu aina johonkin kohteeseen (Kaptelinin & Nardi, 1997), joka ei rajoitu fyysisiin, kemiallisiin tai biologisiin ominaisuuksiin, vaan sillä voi olla myös sosiaalisia ja kulttuurillisia ominaisuuksia (Kaptelinin ym. 1999). Kuutin (1996) mukaan kohde voi olla materiaallinen, vähemmän konkreettinen tai jopa täysin abstrakti. Sisäistäminen/ulkoistaminen viittaa siihen, että toiminta voi olla joko sisäistä tai ulkoista siten, että nämä käsitteet ovat kuitenkin suhteessa toisiinsa (Kaptelinin & Nardi, 1997). Esimerkiksi uuden kielen opettelu tapahtuu aluksi sanakirjojen ja kielioppikirjojen avulla (ulkoistaminen), mutta kun uuden kielen oppii, sen tuottaminen alkaa tapahtua "pään sisällä" (sisäistäminen). Sisäistämisen jälkeen suorituksessa ei siis välttämättä enää tarvita ulkoisia apuvälineitä (Kaptelinin ym. 1999). Välittyminen tarkoittaa välineen keskeistä roolia toimijan ja ympäristön vuorovaikutuksessa (Kaptelinin & Nardi, 1997), jota käsiteltiin jo aiemmin tässä luvussa. Kehitys tarkoittaa sitä, että ihmisen toiminta muuttuu jatkuvasti ja ympäröivän maailman kehitys muokkaa ihmisen toimintaa (Kaptelinin ym. 1999).

Kaptelinin ym. (1999) kehittivät edellä esitettyjen periaatteiden pohjalta tarkastuslistan (engl. The Activity Checklist), jonka avulla toiminnan teoriaa voidaan soveltaa käytännön vuorovaikutussuunnittelussa. Taskastuslistasta on olemassa kaksi eri versiota, joita voidaan käyttää teknologian a) suunnitteluun ja b) arviointiin (Kaptelinin ym. 1999). Tarkastuslista sisältää neljä näkökulmaa teknologian suunnitteluun ja arviointiin (Kaptelinin ym. 1999): 1) Tarkoitus ja päämäärä, 2) Ympäristön sosiaaliset ja fyysiset näkökulmat, 3) Oppiminen, kognitio ja artikulaatio ja 4) Kehitys. Ensimmäinen näkökulma viittaa

teknologian käytön ymmärtämiseen ottaen huomioon toiminnan hierarkkisen rakenteen eri tasot (vrt. toiminnan hierarkkinen rakenne). Toinen näkökulma viittaa kohdeteknologian integroimista vaatimuksiin, välineisiin, resursseihin ja ympäristön sosiaalisiin sääntöihin (vrt. kohteellisuus). Kolmas näkökulma viittaa toimintojen sisäistämiseen ja ulkoistamiseen kohdeteknologiassa (vrt. sisäistäminen/ulkoistaminen). Neljäs periaate viittaa toimintojen historiallisen kehityksen analysointiin (vrt. kehitys).

Toiminnan teoriaa ja vuorovaikutustutkimusta yhdistävä tutkimus on ollut aktiivista. Kuutin (2011) mukaan toiminnan teorian, tietojärjestelmäsuunnittelun ja vuorovaikutustutkimuksen yhdistävä ensimmäinen oppi- ja käsikirjatyypinen teos on toimintakeskeinen suunnittelu (eng. activity-centered design) vuodelta 2004 (Gay & Hembrooke, 2004). Edelleen, Kuutin (2011) mukaan tuorein toiminnan teoriaan käsittelevä oppikirja on Kaptelinin ja Nardin (2006) teos "Acting with Technology: Activity Theory and Interaction Design". Sovellettaessa toiminnan teoriaa käytännössä sen vahvuudet nousevat esille monimutkaisten järjestelmien käytettävyydetutkimusten yhteydessä (Bertelsen & Bødker, 2003; Kuutti, 1996). Kuutin (1996) mukaan toiminnan teorian roolille voidaan nähdä kolme eri roolia käytettävyydetutkimuksessa: 1) monitasoisuus (engl. multilevelness), 2) vuorovaikutus sosiaalisessa kontekstissa (engl. interaction embedded in social context) ja 3) dynamiikan ja kehityksen käsittely (engl. Dynamics and development). Monitasoisuus tarkoittaa sitä, että toiminnan teorian avulla voidaan tarkastella ihmisen toimintaa useista eri näkökulmista ja tarkkuustasoista yhtenäisen viitekehyksen alla (Kuutti, 1996). Vuorovaikutus sosiaalisessa kontekstissa tarkoittaa toiminnan teorian vahvuuksia sosiaaliseen kontekstiin upotetun toiminnan tutkimisessa (Kuutti, 1996). Dynamiikan ja kehityksen käsittely viittaa toiminnan teorian mahdollisuuksiin tutkia erilaisia toiminto-teko-operaatio-ketjuja ja niiden muodostumista sekä muokkaamista tietotekniikan avulla (Kuutti, 1996).

Toiminnan teoriaa voidaan myös kritisoida useista eri näkökulmista ihmisen ja koneen välisen vuorovaikutuksen (HCI) tutkimusalueella. Esimerkiksi Kaptelinin (1996) on esittänyt kritiikkiä toiminnan teorian ja vuorovaikutustutkimuksen rajapintaan liittyen. Tämä kritiikki voidaan jakaa neljään pääkohtaan. Ensimmäiseksi, toiminnan teoria on kehitetty pääosin sellaisten psykologisten teorioiden varaan, jotka painottavat yksilön toimintaa. Tutkittaessa ihmistä käyttäjänä on kuitenkin huomioitava myös esimerkiksi ryhmät ja organisaatiot. Itse asiassa tämä on jonkin verran ristiriitainen näkemys Kuutin (1996) näkemän edun "Vuorovaikutus sosiaalisessa kontekstissa" kanssa. Toiseksi, toiminnan teoria ei nykyisissä muodoissaan painota kulttuurien merkitystä yhtä paljon kuin alun perin (vrt. Vygotski) eikä pysty tarjoamaan kokonaisvaltaista kuvausta siitä ilmiöstä, jonka kulttuuri, arvot, motivaatio, tunteet ja persoonallisuus yhdessä muodostavat. Kolmanneksi, välittyneisyys (engl. mediation), joka on yksi toiminnan teorian keskeisimmistä piirteistä, ei tue esimerkiksi virtuaalitodellisuuden näkökulmaa, jossa raja kontekstin, todellisuuden ja työkalun välillä ei ole perinteisellä tavalla määriteltävissä. Neljänneksi, toiminnan teoria on yleisen tason viitekehys,

mutta sen tueksi käytännön olosuhteissa tarvitaan erillisiä menetelmiä ja tekniikoita, joita ei ole vielä tarpeeksi olemassa. Kaptelinin (1996) mukaan toiminnan teorian soveltaminen käytännössä on tästä johtuen hidasta.

Edellä esitetty teoriatausta auttaa ymmärtämään, mikä on intention rooli osana ihmisen toiminnan tutkimusta ja kuinka intentiotietämystä on pyritty hyödyntämään osana HCI-tutkimusta. Tarkasteltaessa esitettyjä teorioita nimenomaan HCI-näkökulmasta yksi mielenkiintoinen havainto on se, että osa ihmisen toimintaa ja vuorovaikutustutkimusta yhdistävästä tutkimuksesta (esim. Card ym. 1983; Miller ym. 1960; Monk, 1998; Norman, 1986) käyttää selkeästi intention käsitettä omana itsenäisenä käsitteenään ja attribuuttinaan kuvatessaan ihmisen toimintaa. Toinen osa (esim. Bertelsen & Bødker, 2003; Chung ym. 2000; Lamsweerde, 2009) sen sijaan yhdistää enemmän tai vähemmän tavoitteen, motivaation, tarpeen, tekemisen kohteen (engl. object) ja intention käsitteet toisiinsa.

Tässä väitöskirjassa intentio huomioidaan itsenäisenä ihmisen toimintaa ohjaavana tekijänä. Tämä on yksi syy siihen, miksi emme ota tarkastelumme lähtökohdaksi toiminnan teoriaa (Engeström, 1987; Vygotsky, 1978; Leont'ev, 1974, 1978, 1981, 1989) tai suoraan siitä polveutuneita lähestymistapoja (esim. Bertelsen & Bødker, 2003; Gay & Hembrooke, 2004; Kaptelinin & Nardi, 2006). Sen sijaan lähdemme liikkeelle intention "filosofisista ja psykologisista" määrittelmistä (esim. Bratman, 1987; Brentano, 1874/1973; Miller ym. 1960; Shultz, 1980). Emme kuitenkaan poissulje muilta osin toiminnan teoriassa olevia hyväksi todettuja periaatteita, kuten esimerkiksi välittyneisyys ja toiminnan kolmitasoinen hierarkkinen rakenne. Myös tavoitelähtöiset menetelmät (Chung ym. 2000; Lamsweerde, 2009) ovat keskeisessä asemassa, kun intentiolähtöistä lähestymistapaa kehitetään, vaikka nekään eivät aina määrittele eroa intention ja sen lähikäsitteiden (esim. tavoite, motivaatio, tarve) välille. Tavoitelähtöisten lähestymistapojen näkökulmasta pyrimme siis pureutumaan syvällisemmin intention merkitykseen toimintaa ohjaavana tekijänä. Olemme kuitenkin erityisen varovaisia niin toiminnan teorian kuin tavoitelähtöisten menetelmien soveltamisessa, sillä yhtenä hypoteesinamme on se, että erityisesti intention ja tavoitteen käsitteet määrittävät usein myös eron ihmisen toiminnalliselle tavoitteelle ilman teknologiaa tai teknologian kanssa. Vaatimusten määrittelyn ja sen alkuvaiheiden näkökulmasta tämä tarkoittaa eroa aidosti ihmislähtöisen ja teknologia- lähtöisen vuorovaikutussuunnittelun välillä (esim. Bergvall-Kåreborn & Ståhlbröst, 2010; Leikas, 2009; Parkkola, 2006; Viukari, 2010). Esimerkiksi tavoitelähtöisissä lähestymistavoissa "tavoite" viittaa usein teknologian käytön tavoitteeseen (Chung ym. 2000; Lamsweerde, 2009). Hypoteesiamme vahvistaa toiminnan teorian periaatteisiin perustuva Kaptelinin ym. (1999) kehittämä tarkastuslista, joka ottaa heti lähtökohdaksi teknologian tehokkaan käytön eikä ihmisen toiminnallisten päämäärien saavuttamista.

3.3 Kognitiivinen tehtäväanalyysi

Kognitiivinen tehtäväanalyysi (engl. Cognitive task analysis/CTA) on ”sateenvarjokäsite”, joka viittaa sellaisten menetelmien joukkoon, joiden tarkoituksena on saavuttaa ymmärrys ihmisen kognitiivisista prosesseista ja useiden ihmisten välisestä yhteistyöstä tarkkailtaessa ihmisen työtä erityisesti adaptiivisesta (sopeutuva/sopeutumista edistävä) näkökulmasta monimutkaisessa sosioteknisessä ympäristössä (Hoffman & Militello, 2009). *Sosioteknisellä ympäristöllä* Hoffman ja Militello (2009) tarkoittavat kontekstia, jossa useat ihmiset ja useat koneet (erityisesti tietokoneet) toimivat ikään kuin yhtenä suurena järjestelmänä osallistuen monimutkaisten toimintojen suorittamiseen. Kognitiivinen tehtäväanalyysi pyrkii tuottamaan kuvauksen niistä kognitiivisista prosesseista, joita tarvitaan tietyn kognitiivisen tehtävän taitavaan suorittamiseen (Klein & Militello, 1998). Kognitiivisella tehtävällä viitataan puolestaan joukkoon toisiinsa liittyviä mentaalisia toimintoja, joiden kautta pyritään saavuttamaan tietty tavoite (Klein & Militello, 1998). Kognitiiviselle tehtäväanalyysille on olemassa myös lukuisia muita määritelmiä. Esimerkiksi Schraagen ym. (2000) ovat määritelleet kognitiivisen tehtäväanalyysin olevan laajennus perinteisistä tehtäväanalyysimenetelmistä siten, että kognitiivinen tehtäväanalyysi tuo lisätietoa tietämyksestä (engl. knowledge), ajatusprosesseista ja tavoitteiden ”syväisemmistä” rakenteista osana tietyn kognitiivisen tehtävän suorittamista.

Perinteisellä tehtäväanalyysillä (engl. task analysis) viitataan enemmän käyttäytymistä, suorituskykyä ja tehtävä-alitehtävä-prosessia analysoiviin menetelmiin, siinä missä kognitiivinen tehtäväanalyysi korostaa kognition merkitystä, asiantuntijuutta (engl. expertise) ja suurempien tehtäväkokonaisuuksien hallintaa (Wei & Salvendy, 2004). Keskeinen ero perinteisen tehtäväanalyysin ja kognitiivisen tehtäväanalyysin välillä on lisäksi se, että perinteinen tehtäväanalyysi ei huomioi mentaalisia malleja osana tehtävän analyysiä, mutta kognitiivinen tehtäväanalyysi huomioi mentaaliset mallit (Ryder & Redding, 1993; Seamster ym. 1997; Wei & Salvendy, 2004). Yhteisenä piirteenä kaikille kognitiivisen tehtäväanalyysin määritelmille voidaan nähdä se, että kognitiivinen tehtäväanalyysi tarkastelee tyypillisesti tehtävän ja työn käsitteitä kognitiivisten prosessien näkökulmasta säilyttäen tavoitteiden saavuttamisen keskeisenä osana analyysiä (Hoffman & Militello, 2009; Klein & Militello, 1998). Lisäksi kognitiivinen tehtäväanalyysi korostaa asiantuntijuutta ja eksperttimäistä osaamista osana tavoitteiden saavuttamista (Hoffman & Militello, 2009; Hollnagel, 2003).

Kognitiivisen tehtäväanalyysin lisäksi toinen keskeinen tehtäväanalyysin muoto tässä tutkielmassa on hierarkkinen tehtäväanalyysi. *Hierarkkinen tehtäväanalyysi* luokitellaan yksittäiseksi tekniikaksi, joka kuuluu perinteisen tehtäväanalyysin perheeseen (Kirwan & Ainsworth, 1992). Hierarkkinen tehtäväanalyysi kehitettiin 1960-luvulla erityisesti koulutuksen tarpeisiin (Hollnagel, 2003) ja sen keskeisinä kehittäjinä pidetään sellaisia henkilöitä kuin John Annett ja Keith Duncan (katso esim. Annett & Duncan, 1967; Annett ym. 1971; Duncan, 1972; Duncan, 1974). Hierarkkinen tehtäväanalyysi jakaa yksittäiset tehtävät

alitehtäviksi, joiden muodostamalla hierarkialla kuvataan tavoitteen saavuttamista tarvittavan tarkkuustason mukaisesti (Hollnagel, 2003). Tarkemmin määriteltynä hierarkkinen tehtäväänalyysi tarkastelee tiettyjen tehtävien suorittamista sellaisten yksittäisten päämäärien kautta, jotka ovat yhteydessä siihen, mitä ihmisen oletetaan tekevän, ja siihen, kuinka hän yksittäisten suunnitelmien (engl. plan) avulla saa kunkin yksittäisten tehtävän suoritettua (Shepherd, 2000). Hierarkkista tehtäväänalyysiä on sovellettu paljon myös ihmisen ja koneen välisen vuorovaikutuksen tutkimuksessa (Macleod & Rengger, 1993; Pinelle ym. 2003; Stanton, 2006) ja erityisesti käytettävyyden arvioinnin tukena (Macleod, 1994; Stanton & Baber, 2002).

Kognitiiviseen tehtäväänalyysiin kuuluvien menetelmien joukko on laaja. Esimerkiksi Wei ja Salvendy (2004) ovat käyneet läpi 40 keskeistä kognitiivisen tehtäväänalyysin menetelmää ja jakaneet menetelmät neljään eri menetelmäperheeseen: 1) havainnointi ja haastattelu, 2) prosessin jäljittäminen (engl. Process tracing), 3) käsitteelliset tekniikat (engl. Conceptual techniques) ja 4) formaalit menetelmät (engl. Formal methods). Tämän jälkeen he ovat arvioineet jokaisen menetelmän heikkouksia ja vahvuuksia sekä selvittäneet menetelmille soveltuvia käyttökonteksteja ja laatineet konkreettisia ohjeistuksia sopivien menetelmien valintaan. Esimerkiksi, jos tehtävää analysoi useampi kuin yksi analyysoija eikä tehtävän aikaista tarkkailua tai verbaalista kommunikaatiota tarkkailevien kanssa tarvita, ohjeistetaan valitsemaan menetelmäperheeseen 3 (Käsitteelliset tekniikat) kuuluvia menetelmiä. Edelleen, yksittäisiä esimerkkejä kyseiseen menetelmäperheeseen kuuluvista kognitiivisen tehtäväänalyysin menetelmistä olisivat: "conceptual graph analysis" (Gordon ym. 1993), "consistent component" (Fisk & Eggemeier, 1988), "diagramming" (Lesgold ym. 1988), "error analysis" (Norman, 1984; Rasmussen ym. 1987, Reason, 1987) ja "psychological scaling-rating and ranking" (Ryder & Zachary, 1991). Kognitiivisen tehtäväänalyysin menetelmien joukko (Wei & Salvendy, 2004) kattaa myös monta vaatimusten määrittelyn ja vuorovaikutussuunnittelun näkökulmasta keskeistä menetelmää ja tekniikkaa, kuten esimerkiksi strukturoitu, puolistrukturoitu ja strukturoimaton haastattelu, kognitiivinen läpikäynti (Clarke, 1987) ja GOMS (Goals, Operations, Methods, and Selection rules) (esim. Card ym. 1983; Kieras, 1988; Carroll & Olson, 1988).

Kirjallisuudessa termiä "kognitiivinen tehtäväänalyysi" käytettiin ensimmäistä kertaa (Hoffman & Militello, 2009) opetuksen suunnittelun (engl. instructional design) yhteydessä 1970-luvulla, tarkkaan ottaen vuonna 1979 (Gallagher, 1979). Modernin kognitiivisen tehtäväänalyysin syntyyn 1970 ja 1980-lukujen välillä oli kuitenkin jo ennen tätä johtanut pitkä kehitys, joka oli alkanut jo 1870-luvun tienoilla kokeellisen psykologian isänä pidetyn saksalaisen Wilhelm Wundt toimesta (Clark & Estes, 1996; Crandall ym. 2006). Hoffman ja Militello (2009) ovat kuitenkin määritelleet modernin kognitiivisen tehtäväänalyysin alkaneeksi 1970-luvun lopulla. Modernin kognitiivisen tehtäväänalyysin voimakas kehittyminen 1980-luvulla johti siihen, että kyseisen lähestymistavan käytäntöjä alettiin soveltaa eri tiedesuuntausten mukaisesti kahdeksassa eri tiedeyhteisössä. Nämä tiedeyhteisöt olivat Hoffmannin & Militellon (2009) mu-

kaan: 1) Kognitiivinen psykologia, 2) Inhimilliset tekijät ja ergonomia, 3) Kognitiivinen tekniikka, 4) Opetuksen suunnittelu, 5) Kokeellinen psykologia, 6) Etnografia ja kognitiivinen antropologia, 7) Luonnollinen päätöksenteko ja 8) Eurooppalainen työn analyysi ja ergonomia. Edellä mainittuja suuntauksia voidaan tarkastella edelleen Hoffmanin ja Militellon (2009) mukaan kuudesta eri näkökulmasta, jotka ovat a) kognitiivinen teknologia (CSE), b) eksperttiys, c) luonnollinen päätöksenteko, d) työn analyysi, e) sosiologia ja etnografia ja f) ihmiskeskeinen suunnittelu.

Edellä esitettyjen suuntausten, tiedeyhteisöjen ja näkökulmien sisällä on vuosien kuluessa määritelty useita haasteita kognitiivisen tehtävänälyysin kehittämiseksi (Hollnagel, 2003). Hoffmanin ja Militellon (2009) laajojen ja syvälisten tutkimusten perusteella yhdeksi keskeisimmäksi tutkimusongelmaksi kognitiiviselle tehtävänälyysille, erityisesti informaatioteknologian ja ihmisen-teknologia-vuorovaikutustutkimuksen suunnittelun näkökulmasta, on muodostunut innovatiivisen teknologian suunnittelu. Tästä näkökulmasta viisi tärkeintä avointa kysymystä käydään läpi seuraavassa. Ensimmäinen avoin kysymys on seuraava: Kuinka tutkimustietoa, joka on kerätty kognitiivisen tehtävänälyysin avulla olemassa olevista monimutkaisista kognitiivisista järjestelmistä, voitaisiin hyödyntää osana uuden teknologian kehittämisen innovaatioprosessia? (Carroll & Campbell, 1989; Hoffman & Militello, 2009; Hollnagel, 2003). Kognitiivisella järjestelmällä viitataan tässä tilanteeseen, jossa yksittäinen henkilö toimii yhteistyössä sosiaalisen ryhmän kanssa suorittaen tiettyjä toimintoja työvälineiden avulla ja noudattaen samalla kulttuurista omaksumiaan käytäntöjä (Hoffman & Militello, 2009). Tätä ongelmaa kutsutaan innovaatio-ongelmaksi (engl. innovation problem) (Hoffman & Militello, 2009; Hollnagel, 2003). Toinen avoin kysymys on seuraava: Kuinka sellaisia kognitiivisen tehtävänälyysin avulla tuotettuja tutkimustuloksia, jotka kohdistuvat juuri tiettyihin toimintoihin (koskien erityisesti kognitiota ja yhteistyötä) tiettyssä kontekstissa ja erityisesti tiettyinä hetkenä, voidaan soveltaa uuden teknologian kehittämiseen, koska uusi teknologia muuttaa toiminnot, kontekstin, ”tietyn hetken” ja niihin liittyvät käytännöt? Toisin sanoen, vaikka suunnittelijoilla ja tutkijoilla olisi hyvä tietämys nykyisistä tehtävistä, uusi teknologia muuttaa uuden (usein siis myös samankin) tehtävän rakennetta. Tätä ongelmaa kutsutaan visioidun maailman ongelmaksi (engl. envisioned world problem) (Hollnagel & Woods, 2005; Woods, 1998).

”Visioidun maailman ongelmaa” seuraa ”käyttöön liittyvä sopeutumisen ongelma” (engl. the adaptation through use problem) (Hoffman & Militello, 2009). Tämä kolmas avoin kysymys on tarkemmin ilmaistuna seuraava. Kuinka voidaan ennustaa ja hahmotella muutos- ja mukautumisprosessi, joka seuraa teknologista muutosta? (Hoffman & Militello, 2009; Hollnagel, 2003). Esimerkiksi, kuinka uuden teknologian tuoma muutos vaikuttaa sellaiseen sosiotekniseen järjestelmään, jossa on tietyt vakiintuneet käytännöt ja menetelmät, ja kuinka uuden teknologian tuomiin muutoksiin voidaan sopeutua. Neljäs avoin kysymys kuuluu: Kuinka voidaan hallita niitä odottamattomia ja ennalta-arvaamattomia jälkivaikutuksia, joita uusi teknologia aiheuttaa käytännössä?

(Carroll & Rosson, 1992; Woods & Dekker, 2000). On esimerkiksi mahdollista, että innovatiivinen tuote sisältää mahdollisuuksia, jotka käyttäjät tunnistavat ja oivaltavat käytännössä, vaikka suunnittelijat eivät ole osanneet niitä ennakoita (Roth ym. 2006). Viides avoin kysymys viittaa ongelmaan nimeltä ”suunnittelijoiden vaikutusvalta” -ongelma (engl. leverage problem) (Hoffman & Militello, 2009; Hollnagel, 2003). Tällä ongelmalla viitataan kysymykseen, kuinka tutkimustietoa, joka on kerätty kognitiivisen tehtävänälyysin avulla monimutkaisista kognitiivisista järjestelmistä, voitaisiin hyödyntää kohdistamaan suunnittelun resursseja oikein osana suunnitteluprosessia. Tämä ongelma viittaa nimenomaan suunnittelijan ja analysoijan kykyyn ohjata suunnittelua ja suunnitteluprosessia keskittymään keskeisimpiin asioihin (Langdon ym. 2010). Tässä yhteydessä voidaan nostaa esille myös kognitiivisen tehtävänälyysin vahva sidos asiantuntijajärjestelmiin, ekspertteihin ja asiantuntijoihin, jotka toimivat tarkoin määritellyssä työympäristössä tehden tarkoin määriteltyjä työtehtäviä (Hoffman & Militello, 2009). Langdon ym. (2010) nostavatkin esille merkittävän seikan - kuinka kognitiivinen tehtävänälyysi ja sen menetelmät ja tekniikat toimivat tästä poikkeavissa suunnittelutilanteissa (esim. erityisolosuhteet, heikentyneet kognitiiviset kyvyt, noviisikäyttäjät ja vammaiset).

3.4 Käyttäjäpsykologia

Käyttäjäpsykologia on lähestymistapa, joka tarkastelee ihmistä laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja, tavoitteenaan tuottaa tieteelliseen psykologiaan perustuva analyysi ihmisestä käyttäjänä tietyssä vuorovaikutustilanteessa (Oulasvirta & Saariluoma, 2004, 2006; Saariluoma, 2004; Saariluoma, 2011; Saariluoma & Oulasvirta, 2010). Termi ”käyttäjäpsykologia” (engl. user psychology) otettiin virallisesti käyttöön (Saariluoma, 2004, 2011) vuonna 1981 Thomas P. Moranin tieteellisen artikkelin ”An applied psychology of the user” myötä (Moran, 1981). Ihmisen ja tietokoneen välistä vuorovaikutusta oli toki lähestytty aiemminkin psykologian näkökulmasta, esimerkiksi osana ohjelmoinnin psykologian tutkimusta (Adelson, 1981; Anderson ym. 1984; Anderson & Jeffries, 1985; Carroll ym. 1980; McKeithen ym. 1981; Pennington, 1987). Nämä varhaiset tutkimukset tuottivat merkittävästi uutta tietoa ihmisestä käyttäjänä psykologian näkökulmasta, mutta teknologian suunnittelun näkökulmasta ne johtivat myöhemmin tilanteeseen, jossa keskityttiin olemassa olevien teknologioiden käyttöön ja iteratiiviseen testaamiseen, pyrkien etsimään uusia käyttötarkoituksia jo olemassa olevalle teknologialle (Rosson & Carroll, 2002). Käyttäjäpsykologian uusi aalto (Saariluoma, 2004, 2011; Saariluoma & Oulasvirta, 2010) pyrkii sen sijaan lähestymään uuden teknologian suunnittelua näkökulmasta, jossa huolellinen käyttäjien käyttäjäpsykologinen analysointi vasta mahdollisesti johtaa uuden teknologian suunnitteluun ollen aidommin ihmislähtöistä. Tästä näkökulmasta käyttäjäpsykologian ”uuden aallon” soveltaminen osana nimenomaan innovatiivisen ohjelmistotuotteen suunnittelun alkuvaihetta on perusteltua.

Käyttäjäpsykologiassa on myös selvä yhteys ihmisen toiminnan analysointiin osana ihminen-teknologia-vuorovaikutusta, sillä käyttäjäpsykologia pyrkii selvittämään, kuinka ihminen toimii tietyssä kontekstissa (Saariluoma, 2004; Saariluoma & Oulasvirta, 2010; Parkkola, 2006). Esimerkiksi Parkkola (2006) on tutkinut käyttäjäpsykologiaa soveltaen tieto- ja viestintätekniikoiden suunnittelua äideille. Parkkolan (2006) mukaan käyttäjäpsykologia eroaa muista ihmisen toiminnan huomioivista lähestymistavoista HCI-alueella ottamalla ihmisen kokonaisvaltaisemmin huomioon. Esimerkiksi Donald Normanin (1998) toimintakeskeisestä (engl. activity-centered) lähestymistavasta käyttäjäpsykologia erottuu keskittymällä suunnittelun alkuvaiheisiin eikä painota käytettävyyden iteratiivista testaamista (Parkkola, 2006). Käyttäjäpsykologian mukaan (Saariluoma, 2004; Saariluoma & Oulasvirta, 2010) analysoitaessa ihmistä käyttäjänä tulisi osana analyysiä pyrkiä huomioimaan paitsi psykologia, mahdollisesti myös muut ratkaistavan ongelman kannalta tärkeät osa-alueet, kuten esimerkiksi sosiologia ja mielenfilosofia. Käyttäjäpsykologia on siis aidosti poikkitieteellinen lähestymistapa ihmisen toiminnan analyysiin ihminen-teknologia-vuorovaikutustilanteissa.

Tässä tutkimuksessa sovelletaan käyttäjäpsykologiaa vain psykologian näkökulmasta, erityisesti soveltaessamme hierarkkista ja kognitiivista tehtäväanalyysiä osana innovatiivisen ohjelmistotuotteen vaatimusmäärittelyä. Toisaalta, käyttäjäpsykologian ”henki” näkyy väitöskirjassa siten, että psykologian lisäksi tutkielmassa sovelletaan myös tieteellisin menetelmin kehitettyjä ja testattuja tietojenkäsittelytieteiden menetelmiä ja tekniikoita.

3.5 Kohti intentiolähtöistä vaatimusten määrittelyä

Tässä tutkielmassa intentiota lähestytään kahden eri näkökulman ja määrittelyn mukaan. Ensimmäisen näkökulman muodostaa intentio ihmisen toimintaa motivoivana ja kontrolloivana tekijänä (Brentano, 1874/1973; Bratman, 1987; Shultz, 1980). Toisen näkökulman mukaan intentio viittaa sellaisiin toistaiseksi toteutumattomiin osiin ihmisen suunnitelmia, joiden toteuttaminen on kokonaisuudessaan kuitenkin jo alkanut (Miller ym. 1960). Seuraavaksi näkökulmat käydään läpi intentiolähtöisen vaatimusmäärittelyn perspektiivistä. Sen jälkeen kuvataan intentiolähtöinen lähestymistapa.

Intentio ihmisen toimintaa ohjaavana tekijänä

Tässä tutkielmassa kehitetään vaatimusten määrittelyn tueksi viitekehys, joka huomio intention vaatimusmäärittelyn lähtökohtana. Intention ollessa ihmisen toimintaa ohjaava tekijä on luonnollista pyrkiä sisällyttämään intentiota koskeva tietämys suoraan vaatimusmäärittelyn alkuvaiheiden, erityisesti käyttäjätarpeiden tunnistamisen lähtökohdaksi. Intention viitatessa mentaaliseen tilaan, joka motivoi ja kontrolloi ihmisen toimintaa (Brentano, 1874/1973; Bratman, 1987; Shultz, 1980), se vastaa lopulta kysymykseen, miksi joku haluaa tehdä

jotakin ja miksi me toimimme juuri niin kuin me toimimme? Tavoiteltaessa vastauksia tällaisiin perustavalaatuisiin kysymyksiin edistetään samalla perustavalaatuisten tarpeiden (engl. fundamental human needs) tunnistamista. *Perustavalaatuilla tarpeilla* viitataan käyttäjätarpeiden tunnistamisen yhteydessä ihmisen perustarpeisiin (esim. turvallisuuden tarve, sosiaalisuuden tarve) (Kraft, 2012). Intentionien operationalisointi voidaan tehdä esimerkiksi haastattelujen avulla painottaen sopivasti ”minkä takia? ja miksi?” -tyyppisiä kysymyksiä. Tämän tyyppinen ihmisen toiminnan analysointi edellyttää haastattelijalta ja aineiston analysoijalta käyttäytymistieteiden ja modernin psykologian tunnettujen lainalaisuuksien osaamista. Tässä väitöskirjassa intentionilähtöisen lähestymistavan ja siihen sisältyvän käyttäjätarkastelun tueksi otetaan käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006).

Kehitettäessä intentionilähtöistä lähestymistapaa vaatimusten määrittelyn tueksi erotetaan kaksi erityyppistä intentionia toisistaan. Ensimmäinen viittaa mitä-intentioniin, joka ohjaa ja kontrolloi ihmisen toimintaa ja käyttäytymistä mahdollisimman teknologiariippumattomasti. Mitä-intentionio vastaa kysymykseen, ”mitä toimija aikoo tehdä?”. Toinen viittaa kuinka-intentioniin, joka ohjaa ja kontrolloi ihmisen toimintaa ja käyttäytymistä olemassa olevan teknologian huomioiden. Tämä intentionio vastaa kysymykseen, ”kuinka ’mitä-intentionio’ voidaan toteuttaa teknologian avulla?”. Toimintakontekstin analysointi nähdään osana intentionilähtöistä lähestymistapaa. Tämä on perusteltua, sillä konteksti esiintyy kaikissa tunnetuissa ja hyväksi koetuissa intentionia ja ihmisen toimintaa yhdistävissä teorioissa (esim. Dourish, 2001; Engeström, 1987; Husserl, 1962; Hutchins, 1996; Monk, 1998; Salomon, 1993; Vygotsky, 1978; Winograd & Flores, 1987). Intentionit realisoituvat konkreettiseksi toiminnaksi aina jossakin toimintakontekstissa (Gollwitzer & Bargh, 1996). Intentionilähtöisen lähestymistavan näkökulmasta konteksti on keskeinen attribuutti sekä mitä -intentionille että kuinka -intentionille.

Intentionio osana ihmisen suunnitelmia

Toisen näkökulman intentioniin tässä tutkimuksessa muodostaa määritelmä, jonka mukaan intentioniolla viitataan sellaisiin toistaiseksi toteutumattomiin osiin ihmisen suunnitelmia, joiden toteuttaminen on kokonaisuudessaan jo alkanut (Miller ym. 1960). Tätä määritelmää tarkennetaan interaktion intentionin määritelmällä. *Interaktion intentionio* viittaa matalan tason tavoitteisiin (engl. subgoal), jotka ovat osana korkeamman tason tavoitteiden (engl. long-term goal) toteuttamisesta (Xie, 2000). Tässä tutkielmassa interaktion intentioniolla tarkoitetaan erityisesti intentionioita, joiden ohjaamana ihminen toimii suorittaessaan yksittäisiä vuorovaikutustehtäviä käyttöliittymällä. Esimerkiksi maksuessaan laskua verkkopankin avulla käyttäjän vuorovaikutustehtävä koostuu yksittäisistä operaatioista, kuten verkkopankkiin kirjautuminen, uuden laskupohjan valitseminen, tarvittavien tietojen syöttäminen lomakkeeseen ja maksun hyväksyminen henkilökohtaisella salasanalla. Monimutkaiset, yksittäisten toimintojen ja intentionien muodostamat ketjut ja niistä suoriutuminen ovat edellytyksenä korkeam-

man tason tavoitteen saavuttamiselle, joka on tässä esimerkissä laskun maksaminen. Toisin sanoen, suunnitelmien yksityiskohtainen toteuttaminen ja toteutumisen kontrollointi on monimutkainen prosessi, joka edellyttää suunnitelmiin kuuluvien yksittäisten toimintojen ja niiden muodostamien hierarkkisten kokonaisuuksien tarkkaa tuntemista (Miller ym. 1960). Yksittäisten toimintojen tarkasteluun sovelletaan tässä tutkimuksessa hierarkkista tehtävänälyysiä (HTA) (Annett, 2000) ja kognitiivista tehtävänälyysiä (CTA) (Hoffman & Militello, 2009) siten, että toimintojen yksityiskohtainen analysointi viedään läpi käyttäjäpsykologian (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma 2004, 2006) mukaisesti. Tästä näkökulmasta lähestytään a) käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyä ja b) käytettävyyden arviointia.

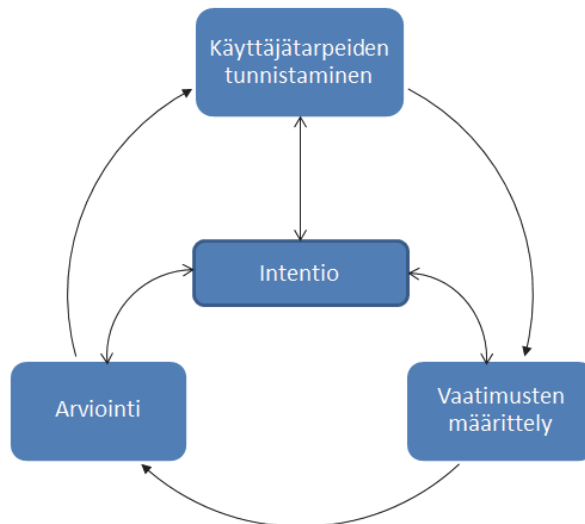
Intentionalähtöinen lähestymistapa

Seuraavassa kuvataan intentionalähtöinen lähestymistapa tiivistetysti kahdessa osassa. Ensiksi kerrotaan sen käyttötarkoitus ja -kohde sekä teoreettiset perusteet. Tämän jälkeen lähestymistapaa operationalisoidaan esittämällä yksinkertainen prosessimalli sen soveltamisesta.

Intentionalähtöinen lähestymistapa on tarkoitettu käytettäväksi käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyn tukena kehitettäessä innovatiivisia ohjelmistotuotteita kuluttajamarkkinoille. Tiedeperustaltaan lähestymistapa nojaa pääasiassa kognitiotieteeseen ja tietojärjestelmätieteeseen, osin myös filosofiaan ja psykologiaan. Yksittäisistä osa-alueista lähestymistavan rakentamisessa on hyödynnetty intention käsitettä, käyttäjäpsykologiaa, hierarkkista tehtävänälyysiä ja kognitiivista tehtävänälyysiä. Lähestymistapa tarkastelee intention käsitettä kahdesta näkökulmasta. Ensimmäisen näkökulman mukaan intentio nähdään ihmisen toimintaa ja käyttäytymistä ohjaavana ja kontrolloivana tekijänä (Brentano, 1874/1973; Bratman, 1987; Shultz, 1980). Toisen näkökulman mukaan intentio viittaa sellaisiin toistaiseksi toteutumattomiin osiin ihmisen suunnitelmia, joiden toteuttaminen on kokonaisuudessaan alkanut (Miller ym. 1960). Lähestymistapa soveltaa käyttäjäpsykologiaa (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006) ihmisten tarkasteluun laitteiden käyttäjinä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Hierarkkista- ja kognitiivista tehtävänälyysiä hyödynnetään niiden tarjoamien menetelmien ja tekniikoiden (Hoffman & Militello, 2009; Wei & Salvendy, 2004) osalta.

Intentionalähtöisen lähestymistavan mukainen vaatimusten määrittely voidaan kuvata prosessina (Kuvio 5), jonka keskeiset vaiheet ovat: 1) käyttäjätarpeiden tunnistaminen, 2) vaatimusten määrittely ja 3) käytettävyyden arviointi. Käyttäjätarpeiden tunnistamisessa lähdetään liikkeelle mahdollisimman teknologiariippumattomasta näkökulmasta (Bergvall-Kåreborn & Ståhlbröst, 2010; Kankainen & Oulasvirta, 2003; Patnaik & Becker, 1999; Vredenburg ym. 2002) analysoimalla ihmisen intentioista johdettuja perustavanlaatuisia tarpeita (esim. Kraft, 2012). Käyttäjätarpeiden

tunnistamisen tukena käytetään tarkoituksenmukaisia haastattelu-, fokusryhmä-, havainnointi- ja osallistumistekniikoita sekä kysely- ja tutkimustekniikoita ja dokumentaatioon perustuvia tekniikoita (vrt. Sutcliffe, 2002). Vaatimusten määrittelyssä keskitytään käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyyn. Tavoitteena on hyödyntää mahdollisimman laajasti olemassa olevaa tietämystä, jota ihmisen ja tietokoneen vuorovaikutustutkimuksessa on saavutettu eri vuosikymmeninä (Bødker, 2006; Kamppuri, 2011; Kaptelinin ym. 2003). Käytännössä tämä tarkoittaa käytettävyyssuunnittelun, käyttäjäkeskeisen suunnittelun ja käyttäjäkokemussuunnittelun periaatteiden soveltamista. Vaatimusten määrittelyssä sovelletaan edellä mainittujen suuntausten lisäksi hierarkkisen ja kognitiivisen tehtäväanalyysin menetelmiä ja tekniikoita (vrt. Hoffman & Militello, 2009; Wei & Salvendy, 2004). Arvioinnissa keskitytään todentamaan, että kehitettävä järjestelmä täyttää sille asetetut käyttäjätarpeet, käytettävyy- ja käyttäjäkokemusvaatimukset sekä pystyy vastaamaan käyttäjien intentioihin. Arviointia tulee tehdä sekä korkean että matalan abstraktitason vaatimuksille. Matalan abstraktitason vaatimukset voidaan arvioida esimerkiksi käytettävyydestauksella (Rubin, 1994). Korkean abstraktitason vaatimuksia tulee pyrkiä tarkentamaan määrittämällä niille kriittisiä rajoja tai käyttämällä menetelmiä, joiden avulla voidaan paikantaa käytettävyysongelmia tarkasti korkean abstraktitason vaatimuksista. Prosessi on iteratiivinen. Ellei arvioinnilla pystytä osoittamaan käyttäjän intention täyttymistä, siirrytään käyttäjätarpeiden tunnistamisen ja vaatimusten määrittelyn kautta uudelleen arviointiin. Iteraatiota jatketaan, kunnes ohjelmistotuote pystyy vastaamaan sille asetettuihin käyttäjätarpeisiin, vaatimuksiin ja käyttäjien intentioihin.



KUVIO 5 Intentionalähtöisen lähestymistavan mukainen vaatimusten määrittelyn prosessi

Prosessi eroaa muista lähestymistavoista monessa suhteessa. Prosessin lähtökohdaksi otetaan intentio, koska se määrittää yhdessä arvojen ja motivaation kanssa, miksi ihminen toimii tietyssä tilanteessa juuri tietyllä tavalla (Gollwitzer & Bargh, 1996). Prosessi ei tue toiminnan teoriaa (Engeström, 1987; Vygotsky, 1978; Leont'ev, 1974, 1978, 1981, 1989) tai siitä suoraan polveutuneita lähestymistapoja (esim. Bertelsen & Bødker, 2003; Gay & Hembrooke, 2004; Kaptelinin & Nardi, 2006), koska ne eivät määrittele eroa intention, tavoitteen, motivaation ja tarpeen välille. Prosessissa lähdetään liikkeelle intention ”filosofisista ja psykologisista” määritelmistä (esim. Bratman, 1987; Brentano, 1874/1973; Miller ym. 1960; Shultz, 1980), joissa intentio huomioidaan itsenäisenä ihmisen toimintaa ohjaavana tekijänä. Hypoteesina on se, että erityisesti intention ja tavoitteen käsitteet määrittävät usein myös eron ihmisen toiminnalliselle tavoitteelle ilman teknologiaa tai teknologian kanssa. Vaatimusten määrittelyn ja sen alkuvaiheiden näkökulmasta tämä tarkoittaa eroa teknologiariippumattoman ja teknologialähtöisen vuorovaikutussuunnittelun välillä (Bergvall-Kåreborn & Ståhlbröst, 2010; Leikas, 2009; Parkkola, 2006; Viukari, 2010).

3.6 Yhteenveto intentiolähtöisestä vuorovaikutussuunnittelusta

Tämän luvun tarkoituksena oli kuvata vuorovaikutussuunnitteluun kehitetyn intentiolähtöisen lähestymistavan peruskäsitteitä ja taustateorioita sekä esittää tiivistetysti, mitä intentiolähtöisellä lähestymistavalla tarkoitetaan. Lähestymistavan keskeisin peruskäsite on intentio, jolla tarkoitetaan mentaalista tilaa, joka ohjaa ja kontrolloi ihmisen toimintaa ja käyttäytymistä. Toinen, vaikkakin vähemmän merkityksellinen käsite on toiminta. Toiminnan käsitteen tutkimuksesta esiteltiin fenomenologiaa, hajautettua kognitiota ja toiminnan teoriaa. Näiden osalta todettiin, että toiminnan teoriaa ja siitä suoraan polveutuneita lähestymistapoja ei voida käyttää pääasiallisina teoreettisina lähtökohtina tälle tutkimukselle, koska niissä ihmisen intentiot eivät ole selkeästi näkyvissä. Yksittäisiä teorian periaatteita kuitenkin hyödynnetään, joista esimerkkeinä ovat välittyneisyys ja toiminnan kolmitasoinen hierarkkinen rakenne. Muina teorioina kuvattiin kognitiivista ja hierarkkista tehtäväanalyysiä. Kognitiivinen tehtäväanalyysi viittaa menetelmien joukkoon, joiden tarkoituksena on saavuttaa ymmärrys ihmisen kognitiivisista prosesseista ja useiden ihmisten välisestä yhteistyöstä. Hierarkkinen tehtäväanalyysi jakaa yksittäiset tehtävät alitehtäviksi, joiden muodostamalla hierarkialla kuvataan tavoitteen saavuttamista tarvittavan tavoitetason mukaisesti.

Luvun lopussa johdettiin ja kuvattiin intentiolähtöinen lähestymistapa käytettäväksi käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyn tukena. Se on monitieteinen konstruktio, jonka pääasiallisina tukipuina ovat intention käsite, käyttäjäpsykologia ja kognitiivinen sekä hierarkkinen tehtäväanalyysi. Lähestymistavan soveltamista havainnollistettiin ihmisen ja koneen välistä vuo-

rovaikutusta koskevien vaatimusten määrittelyprosessilla. Seuraavassa luvussa (luku 4) esitetään tutkimusongelma ja tarkoin määritellyt tutkimuskysymykset.

4 TUTKIMUSONGELMA JA -METODOLOGIA

Tässä tutkielmassa keskitytään käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyyn tilanteessa, jossa kehitetään kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta. Tässä luvussa määritellään ensin tutkimusongelma ja sitä jäsentävät tutkimuskysymykset. Sen jälkeen kerrotaan, millaisia tutkimusmenetelmiä työssä on käytetty. Lopuksi käydään läpi tutkimuksen reliabiliteetti ja valideetti.

4.1 Tutkimusongelma ja -kysymykset

Tämän tutkielman tutkimusongelma on: ”Kuinka käyttäjätarpeiden tunnistaminen ja vaatimusten määrittely tulisi suorittaa osana vuorovaikutussuunnittelua kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta?” Tutkimusongelma voidaan jäsentää neljään tutkimuskysymykseen. Ensimmäinen tutkimuskysymys keskittyy vaatimusmäärittelyn yleisiin haasteisiin kehitettäessä innovatiivista ohjelmistotuotetta. Toisessa tutkimuskysymyksessä rajaudutaan erityisesti intentiolähtöiseen käyttäjätarpeiden tunnistamiseen. Kolmas tutkimuskysymys käsittelee käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyä. Neljäs tutkimuskysymys rajautuu kahteen innovatiivisen ohjelmistotuotteen käytettävyyksivaatimusten määrittelyn erityisongelmaan: a) Kuinka voidaan paikantaa käytettävyysoongelmia käyttöliittymäprototyypin käyttöliittymästä, ellei käytettävyyksivaatimuksia ole määritelty? b) Kuinka voidaan määrittellä kriittisiä rajoja kvantitatiivisille käytettävyyksivaatimuksille tilanteissa, joissa vastaavaa tuotetta tai tuotteen edellistä versiota ei ole olemassa? Seuraavassa kutakin tutkimuskysymyksen valintaa taustoitetaan ja motivoidaan ennen sen määrittämistä.

Intentiolähtöinen vaatimusten määrittely

Vaatimusmäärittely on erityisen haasteellista tilanteissa, joissa vaatimuksia määritellään kuluttajamarkkinoille suunnatulle innovatiiviselle ohjelmistotuotteelle. Vaatimusmäärittelyn näkökulmasta ongelmia aiheuttavat seuraavat seikat. Ensimmäiseksi, tulevien käyttäjien on vaikea ilmaista ja visioida täysin uudentyyppiseen ohjelmistotuotteeseen liittyviä tarpeita ja vaatimuksia (Casab ym. 2010). Toiseksi, kehitettäessä täysin innovatiivista ohjelmistotuotetta ei välttämättä edes tiedetä keitä loppukäyttäjät ovat ja mihin tarkoitukseen he tulevat sovellusta käyttämään (Mayhew, 1999). Kolmanneksi, vastaavaa tuotetta, jota voitaisiin käyttää vaatimusten määrittelyn lähtökohtana, ei tyypillisesti ole olemassa (Dearden & Howard, 1998). Neljänneksi, tällaisessa suunnittelutilanteessa ei osata ennakoida varmuudella, kuinka uusi tuote vaikuttaa loppukäyttäjien käyttäytymiseen, kuinka kilpailukykyinen tuote todellisuudessa on ja kenelle se itse asiassa tuo hyötyä.

Edellä esitettyihin haasteisiin ja ongelmiin on esitetty ratkaisuehdotuksia useista eri vaatimusmäärittelyn näkökulmista. Esimerkkejä näkökulmista ratkaista ongelmia ovat keskittyminen vaatimusmäärittelyn alkuvaiheisiin (Yu, 1997, 1999), tavoitelähtöisten vaatimusmäärittelymenetelmien kehittäminen (Chung ym. 2000; Dardenne ym. 1993; Lapouchnian, 2005), luovuutta korostavien vaatimusmäärittelymenetelmien ja -tekniikoiden kehittäminen (Grube & Schmid, 2008; Maiden & Robertson, 2005a, 2005b; Maiden ym. 2004a, 2004b; McFadzean, 1998) ja käyttäjäkeskeisen lähestymistavan kehittäminen. Näiden lisäksi on pyritty lähestymään eri näkökulmista innovatiivisuutta osana ohjelmistotuotteen kehittämistä (Moore & Benbasat, 1991; Quinn ym. 1996; Swanson, 1994; von Hippel & Krogh, 2003; Von Krogh ym. 2003; Von Krogh ym. 2003; Vujovic & Ulhøi, 2008).

Innovatiivisten ohjelmistotuotteiden kehittämisen näkökulmasta ollaan kuitenkin edelleen tilanteessa, jossa tarvitaan uusia menetelmiä, tekniikoita ja ajattelutapoja ilmenneiden haasteiden selvittämiseksi (Fisher ym. 2005; Rose, 2011; Tang, 1998; Von Krogh ym. 2003). Esimerkiksi vaatimusmäärittelyn haasteiden on ennustettu 2010-luvun kuluessa liittyvän yhä useammin juuri innovatiivisten ohjelmistotuotteiden suunnitteluun (Sommerville, 2005). Nämä haasteet (Sommerville, 2005) ovat tyypillisesti yhteydessä tarpeeseen ymmärtää syvällisemmin olemassa olevan ja kehittävän teknologian suhdetta ja tarpeeseen kehittää innovatiivisia ohjelmistotuotteita hyvin nopeasti reagoiden ympäröivään markkinatilanteeseen ja käyttäjien muuttuneisiin tarpeisiin. Esimerkiksi käyttäjäkeskeinen suunnittelu ei sovellu parhaalla mahdollisella tavalla innovatiivisen ohjelmistotuotteen suunnitteluun siinä havaittujen käyttäjätarpeiden tunnistamisen ja vaatimusmäärittelyn puutteiden takia (Leikas, 2009; Parkkola, 2006; Viukari 2010). Myös kognitiivisen tehtäväanalyysin keskeiset haasteet liittyvät usein innovatiivisen teknologian suunnitteluun (Hoffman & Militello, 2009).

Tässä väitöskirjassa edellä esitettyjä haasteita lähestytään ottamalla vaatimusmäärittelyn lähtökohdaksi intentio. Intentiolla viitataan tyypillisesti men-

taaliseen tilaan, joka ohjaa ja kontrolloi ihmisen toimintaa ja käyttäytymistä (Brentano, 1874/1973; Shultz, 1980) ja määrittää yhdessä arvojen ja motivaation kanssa tarpeen sille, miksi ihminen toimii tietyissä tilanteissa juuri tietyllä tavalla (Gollwitzer & Bargh, 1996). Tästä syystä intention ottaminen vaatimusmäärittelyn lähtökohdaksi on hyvin perusteltua kehitettäessä innovatiivista ohjelmistotuotetta. Intention lähtöisyyden lisäksi tässä väitöskirjassa pyritään huomioimaan tietyissä määrin myös liiketoimintalähtöisyys ja teknologialähtöisyys. Tämä on välttämätöntä, jotta voidaan saavuttaa kokonaisvaltainen ymmärrys siitä, miksi uutta teknologiaa ollaan todella kehittämässä ja millaisiin tarpeisiin sen todella tulee vastata. Edellä esitettyjen lähtökohtien perusteella määritellään väitöskirjan ensimmäinen tutkimuskysymys.

Tutkimuskysymys 1: Kuinka vaatimusten määrittely voidaan suorittaa intention lähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta?

Intention lähtöisyys ja käyttäjäpsykologia osana käyttäjätarpeiden tunnistamista

Käyttäjätarpeiden tunnistaminen on keskeinen osa vaatimusten määrittelyä sijoittuen vaatimusmäärittelyn alkuvaiheisiin (Anastassova ym. 2007; Coble ym. 1997; Kujala ym. 2001; Kotonya & Sommerville, 2002; Lindgaard ym. 2006; Mayhew, 1999; Preece, 2002; Robertson, 2001; Sutcliffe, 2002). Käyttäjätarpeiden tunnistaminen on erityisen merkittävässä roolissa kehitettäessä innovatiivista ohjelmistotuotetta (Lindgaard ym. 2006; Macintosh, 1978). Tällöin käyttäjätarpeiden tunnistamisen tavoitteena on mahdollistaa, että a) kehitettävä tuote vastaa loppukäyttäjien tarpeita, b) loppukäyttäjät hyväksyvät kehitettävän tuotteen ja sen käyttötarkoituksen ja c) lopputuotteen käyttö olisi mahdollisimman hyödyllistä (Anastassova ym. 2007; McKeown ym. 1994).

Käyttäjätarpeiden tunnistaminen ei kuitenkaan ole ongelmaton. Siihen liittyvät menetelmät ja tekniikat eivät esimerkiksi anna selkeitä vastauksia seuraaviin kysymyksiin (Lindgaard ym. 2006): 1) millaisia vaiheita käyttäjätarpeiden tunnistamiseen tulisi sisältyä, 2) mitä käyttäjätarpeiden tunnistamisen lopputuloksena tulisi syntyä, 3) kuinka käyttäjätarpeiden tunnistamisen tulokset tulisi esittää ja dokumentoida ja 4) kuinka käyttäjätarpeiden tunnistuksen tuloksista tulisi keskustella ja kenen kanssa (Lindgaard ym. 2006). Lisäksi, koska käyttäjätarpeiden tunnistaminen nähdään usein kiinteänä osana vaatimusten määrittelyä (Lindgaard ym. 2006), käyttäjätarpeiden tunnistamisella ja vaatimusten määrittelyllä on myös yhteisiä haasteita. Esimerkiksi tutkimuskysymyksen 1 yhteydessä esitetyt neljä haastetta voidaan nähdä myös käyttäjätarpeiden tunnistamisen ongelmana. Käyttäjätarpeiden tunnistamisen haasteisiin onkin pyritty vastaamaan keskeisessä kirjallisuudessa (esim. Kotonya & Sommerville, 2002; Mayhew, 1999; Preece, 2002; Sutcliffe, 2002) kehittämällä nimenomaan vaatimusten määrittelyn menetelmiä ja tekniikoita. Käyttäjätarpeiden tunnistamista (esim. Lindgaard ym. 2006) ja vaatimusmäärittelyn alkuvaiheita (esim. Kotonya & Sommerville, 2002; Mayhew, 1999; Preece, 2002; Sutcliffe, 2002) käsittelevä keskeinen kirjallisuus vaikuttaa ohittavan kuitenkin lähes ko-

konaan tarpeiden käsittelyn psykologian näkökulmasta. Aiemmin esitetty Habbo Hotel -esimerkki havainnollistaa konkreettisesti, kuinka perustavanlaatuiset ihmisen tarpeet (Kraft, 2012) ovat yhteydessä innovatiivisen ohjelmistotuotteen menestykseen. Tällöin käyttäjätarpeiden tunnistamisessa korostuvat intentio, motivaatio ja arvot, jotka ovat toiminnan psykologian (Gollwitzer & Bargh, 1996) peruskäsitteitä.

Käyttäjätarpeiden tunnistamisen näkökulmasta keskeistä on myös tarve laajentaa perinteistä käyttäjänäkökulmaa kuluttajakeskeiseen suuntaan (Han ym. 2001; Kim & Han, 2008; Kwahk & Han, 2002; Tuunanen ym. 2008). Käyttäjänäkökulman laajentamisen tarve on tunnistettu myös muissa tutkimuksissa. Esimerkiksi Lamb ja Kling (2003) ovat täydentäneet käyttäjänäkökulmaa sosiaalisen toimijan näkökulmalla. Tällöin käyttäjää tarkastellaan muun muassa sosiaalisen identiteetin näkökulmasta, jolla viitataan sosiaalipsykologiassa ryhmäjäsenyyksien pohjalta muodostuvaan minäkäsitykseen (Helkama ym. 2010). Sosiaalinen identiteetti ilmenee esimerkiksi Habbo Hotel -ympäristössä siten, että palvelua käyttävät teini-ikäiset perustavat virtuaalimaailmaan erilaisia ryhmiä (esim. armeijahenkiset, gootit, punkkarit), joihin liittyminen edellyttää ryhmän määrittämien kriteerien täyttämistä muun muassa pukeutumisen suhteen. Tiettyihin ryhmiin kuulumisen avulla nuoret voivat korostaa ja tuoda esille omaa identiteettiään, mutta ryhmissä pysyminen edellyttää toisaalta sitoutumista ryhmän mukana jatkuvasti muuttuviin kuluttajavalintoihin. (Lehdonvirta ym. 2009). Tämän tyyppisten seikkojen huomioiminen on ymmärrettävästi keskeinen osa onnistunutta käyttäjätarpeiden tunnistamista.

Myös käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004, Saariluoma & Oulasvirta, 2010) on perinteistä käyttäjänäkökulmaa laajentava lähestymistapa. Käyttäjäpsykologia pyrkii tarkastelemaan ihmistä laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja (Oulasvirta & Saariluoma, 2004, 2006; Saariluoma & Oulasvirta, 2010). Habbo Hotel -esimerkissä sosiaalisen identiteetin käsitteen ja teorian linkittäminen osaksi innovatiivisen ohjelmistotuotteen käyttäjätarpeiden tunnistamista nähdään tässä väitöskirjassa osana käyttäjäpsykologiaa.

Tässä väitöskirjassa oletetaan, että kehitettäessä innovatiivista ohjelmistotuotetta käyttäjätarpeiden tunnistaminen tulisi aloittaa mahdollisimman teknologiariippumattomasta näkökulmasta analysoimalla perustavanlaatuisia ihmisen tarpeita (esim. Kraft, 2012). Intentio (Esim. Ajzen 1985; Fishbein & Ajzen, 1975; Bratman, 1987; Gollwitzer & Bargh, 1996; Hampshire, 1965; Irwin, 1971; James, 1890; Lewin, 1951; Meiland, 1970; Miller ym. 1960; Shultz, 1980; Tolman, 1932) on tällöin perusteltu lähtökohta innovatiivisen ohjelmistotuotteen käyttäjätarpeiden tunnistamiselle. Käyttäjätarpeiden tunnistaminen ja vaatimusten määrittely tulee lisäksi viedä läpi hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Tästä syystä käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006, Saariluoma & Oulasvirta, 2010) soveltuu taustateoriaksi intentiolähtöiselle käyttäjätarpeiden tunnistamiselle. Edellä esitettyjen lähtökohtien perusteella voimme määritellä väitöskirjan toisen tutkimuskysymyksen.

Tutkimuskysymys 2: Kuinka käyttäjäpsykologiaa ja intentiolähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusten määrittelyä?

Käytettävyys- ja käyttäjäkokemusvaatimusten määrittely: kuluttajamarkkinoille suunnattu innovatiivinen ohjelmistotuote

Kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen kehittäminen asettaa käytettävyys- ja käyttäjäkokemusvaatimusten määrittelylle neljä haastetta: 1) Kvantitatiivisten käytettävyysvaatimusten määrittäminen. 2) Käytettävyys- ja käyttäjäkokemusvaatimusten sisällön psykologinen analyysi. 3) Selkeä prosessi, joka yhdistää käytettävyys- ja käyttäjäkokemusvaatimukset osana ohjelmistokehitystä ja vaatimusmäärittelyä. 4) Korkean abstraktitason käytettävyysvaatimusten aiheuttamat ongelmat käytettävyyden arvioinnille. Näiden haasteiden lisäksi yksi keskeisimmistä haasteista liittyy käyttäjäkokemukseen: tarve keskittyä perinteiseksi koetun käytettävyyden tehokkuustuottavuus-näkökulman sijasta enemmän ohjelmistotuotteen tuomiin hedonisiin piirteisiin sekä käyttäjien tuotteen käyttöä koskeviin kokemuksiin ja vaikutelmiin (Han ym. 2001; Kim & Han, 2008; Kwahk & Han, 2002; Rossi & Tuunanen, 2010; Tuunanen ym. 2008).

Edellä esitettyjä haasteita on pyritty ratkaisemaan kehittämällä erilaisia lähestymistapoja, menetelmiä ja tekniikoita käytettävyyteen liittyvien vaatimusten määrittelyn tueksi (esim. Gould & Lewis, 1985; Mayhew, 1999; Nielsen, 1993; Preece, 2002; Stone, 2005; Wixon & Wilson, 1997). Esimerkiksi Preece (2002) on pyrkinyt yhdistämään käytettävyys- ja käyttäjäkokemusvaatimusten määrittelyn osana vuorovaikutussuunnittelua. Tyypillisesti näitä ratkaisuehdotuksia ei ole kuitenkaan kehitetty erityisesti kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen näkökulmasta, vaan ne toimivat enemmänkin yleisinä ohjeistuksina kaiken tyyppisten ohjelmistotuotteiden kehittämiseksi. Käytettävyysvaatimusten määrittely on kuitenkin sekä teoriassa että käytännössäkin edelleen yksi vuorovaikutussuunnittelun ongelmallisimmista ja laiminlyödyimmistä vaiheista (Heiskari ym. 2009; Jokela ym. 2006, Kantola & Jokela, 2007).

Yksi eniten käytetyistä ja tunnetuimmista lähestymistavoista käytettävyysvaatimusten määrittelyyn on ISO 13407-standardin mukainen käyttäjäkeskeinen suunnittelu (Maquire & Bevan, 2002; Väänänen-Vainio-Mattila, 2011; Jokela ym. 2003; Jokela ym. 2006; Kantola & Jokela, 2007). ISO 13407-standardi kuvaa vuorovaikutteisten järjestelmien suunnitteluprosessin kuitenkin vain yleisellä tasolla ja tarvitsee käytännössä rinnalleen yksityiskohtaisemmat menetelmät ja tekniikat (Jokela ym. 2003; Väänänen-Vainio-Mattila, 2011). Sama koskee käyttäjäkeskeisen suunnittelun uudistettua ISO 9241-210 standardia (ISO 9241-210, 2010). Esimerkiksi ISO/TR 16982 -standardissa esitellään joukko tällaisia menetelmiä ja arvioidaan sekä määritellään myös yksittäisten menetelmä- ja tekniikkaperheiden soveltuvuutta tuotekehityksen eri vaiheisiin. ISO/TR 16982-standardin mukaan vaatimusten määrittelyyn parhaiten sopivia käyttäjäkeskeisiä menetelmiä ovat käyttäjien tarkkailu, kysely, haastattelu ja ääneen

ajattelu. Edelleen, ISO/TR 16982 -standardin (ISO/TR 16982, 2002) mukaan vaatimusten määrittelyyn sopivia (mutta ei parhaiten sopivia) käyttäjäkeskeisiä menetelmiä ovat suorituksiin liittyvät mittaukset, kriittisten tapahtumien analyysi, yhteistyössä tapahtuva suunnittelu ja arviointi, luovat menetelmät, dokumenttien analyysi, mallipohjaiset analyysit ja asiantuntija-arvioinnit.

Tarkasteltaessa erityisesti innovatiivisen ohjelmistotuotteen vaatimusmäärittelyä voidaan huomata, että sitä on lähestytty usein luovien menetelmien kuten aivoriihien ja luovien ideapajojen näkökulmasta (Grube & Schmid, 2008; Maiden & Robertson, 2005a, 2005b; Maiden ym. 2004a, 2004b; McFadzean, 1998). Tämä on luonnollista, sillä vaatimusmäärittely nähdään luovana prosessina, jossa sidosryhmät ja kehittäjät työskentelevät yhdessä luodakseen ideoita uudelle tuotteelle, jotka voidaan lopulta kuvata vaatimuksina (Maiden ym. 2004a, 2004b). ISO/TR 16982-standardissa esitetyn luokittelun mukaan aivoriihet ja luovat pajat eivät kuitenkaan sovellu parhaalla mahdollisella tavalla vaatimusten määrittelyn menetelmiksi. Suurimman ongelman muodostaa kuitenkin se, että lähes kaikki käyttäjäkeskeisen suunnittelun menetelmät ja tekniikat ohittavat käytettävyyteen liittyvien vaatimusten määrittelyn hyödyntäen psykologista tietämystä. Esimerkiksi Habbo Hotel (katso myös Johnson ym. 2005; Lehdonvirta ym. 2009; Mäntymäki & Salo, 2012) ja Angry Birds (katso myös Mauro, 2011) -esimerkit havainnollistivat konkreettisesti, kuinka keskeistä sovelluksen menestymisen kannalta on liittää muun muassa sosiaalipsykologiaa ja ihmisen kognitiivisen kapasiteetin rajoituksia ilmentävä tietämys osaksi vaatimusmäärittelyä. Käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006) on lähestymistapa, joka pyrkii vastaavaan suunnitteluajatteluun. Tämän jälkeen oman haasteensa muodostaa sellaisten vaihe vaiheelta etenevien menetelmien puute, joiden avulla käyttäjäpsykologinen tietämys saataisiin yhdistettyä osaksi käytettävyys- ja käyttäjäkokemusvaatimusten määrittelyä.

Tässä väitöskirjassa edellä esitettyjä haasteita lähestytään keskittyen kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vaatimusmäärittelyyn. Väitöskirjassa käyttäjätarpeiden tunnistaminen nähdään osana vaatimusmäärittelyä ja sen lähtökohdaksi otetaan intentio, joka on ihmisen toimintaa ohjaava tekijä. Intentionlähtöisen vaatimusmäärittelyn lähtökohtana on hyödyntää mahdollisimman laajasti olemassa olevaa tietämystä, jota ihmisen ja tietokoneen vuorovaikutustutkimuksessa on saavutettu eri vuosikymmeninä (Bødker, 2006; Kamppuri, 2011; Kaptelinin ym. 2003). Näistä lähtökohdista määrittelemme väitöskirjan kolmannen tutkimuskysymyksen.

Tutkimuskysymys 3: Kuinka käytettävyys- ja käyttäjäkokemusvaatimuksia voidaan määrittää osana kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vuorovaikutussuunnittelua?

Kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen käytettävyyden arviointi

Useiden tutkimusten perusteella nähdään, että käytettävyyden todentaminen on haasteellista kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta (Aaen, 2008; Casab ym. 2010; Dearden & Howard, 1998; Maiden & Robertson, 2005a, 2005b; Mayhew, 1999; Tuunanen ym. 2008). Yksi erityinen haaste, joka nousee tämän ongelman asetelun taustalta, on korkean abstraktitason vaatimusten aiheuttamat ongelmat käytettävyyden arvioinnille.

Tämän tyyppisiin ongelmiin on etsitty vastauksia eri tavoilla eri aikakausien ja lähestymistapojen mukaisesti. Esimerkiksi 1980-luvulla syntynyt käytettävyyssuunnittelu (engl. usability engineering) korosti kvantitatiivisten käytettävyyksivaatimusten määrittämistä ja niiden mukaista arviointia (Tyldesley, 1988; Wixon & Wilson, 1997). Samoihin aikoihin kehitetyt käytettävyyden arviointimenetelmät, kuten käytettävyytestaus (Rubin, 1994), toivat selvästi esille käytettävyyksiarvioinnin ensisijaisena tavoitteena selvittää, missä määrin testattava tuote täyttää sille asetetut kvantitatiiviset käytettävyyksivaatimukset. 2000-luvulla käytettävyyden arviointimenetelmät (Ivory & Hearst, 2001; Paganelli & Paternò, 2002; Paternò & Ballardini, 1999, 2000) ovat kehittyneet suuntaan, joka ei korosta käytettävyyksivaatimukseen perustuvaa käytettävyyden arviointia tai käytettävyyksiongelmiin tarkkaa paikantamista ennalta määriteltyjen vaatimusten perusteella. Uudet menetelmät pyrkivät enemmänkin keräämään mahdollisimman paljon käyttäjää ja käytettävyyttä koskevaa dataa, jonka analysointi toimii käytettävyyksiarvioinnin perustana. Yhtenä syynä tällaiselle kehityssuunnalle saattaa olla se, että vaatimusmäärittelyn lähestymistavat, menetelmät ja tekniikat eivät pysty vastaamaan innovatiivisen ohjelmistotuotteen kehittämisen haasteisiin (Sommerville, 2005). Käytännössä tämä tarkoittaa sitä, että ellei määrällisiä matalan abstraktitason käytettävyyksivaatimuksia pystytä määrittämään suunnittelun alkuvaiheessa, on käytettävyyksiarviointi rakennettava perustumaan laadullisiin korkean abstraktitason käytettävyyksivaatimuksiin. Käytettävyyksiarvioinnin haasteita liittyen korkean abstraktitason käytettävyyksivaatimusten aiheuttamiin ongelmiin voidaankin lähestyä myös tavoitelähtöisen lähestymistavan näkökulmasta.

Tavoitelähtöisessä lähestymistavassa (esim. Cheng & Atlee, 2007; Lamsweerde, 2000; Sommerville, 2005; Sutcliffe, 1998) vaatimusmäärittelyä lähestytään tarkastelemalla vaatimuksia erilaisten tarkkuustasojen mukaisesti. Tavoitteet voidaan jakaa korkean tason tavoitteisiin (engl. high-level goal) ja matalan tason tavoitteisiin (engl. low-level goal), joista tarkennetaan edelleen varsinaisia vaatimuksia (katso esim. Chung ym. 2000; Lamsweerde, 2009). Kirjallisuuden mukaan korkean tason tavoitteista voidaan jalostaa matalan tason tavoitteita tarkentamalla niitä jatkuvilla "kuinka?" kysymyksillä

(Lamsweerde, 1995, 2000). Vastaavasti kysymällä ”miksi?” kysymyksiä saadaan yksityiskohtaiset vaatimukset ja matalan tason tavoitteet purettua korkean tason tavoitteiksi (Lapouchnian, 2005). Käytettävyyssarvioinnin näkökulmasta keskeisiä ovat matalan tason käytettävyystavoitteet ja niistä johdetut käytettävyystvaatimukset (Mayhew, 1999).

Kuten edellä todettiin, kehitettäessä nimenomaan kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta yksityiskohtaisten ja tarkkojen käytettävyyttä koskevien vaatimusten määrittely on kuitenkin ongelmallista (katso esim. Aaen 2008; Casab ym. 2010; Dearden & Howard, 1998; Maiden & Robertson 2005a, 2005b; Mayhew, 1999; Tuunanen ym. 2008). Erityisen ongelmallista on määrittellä kriittisiä rajoja käytettävyyden vaatimuksille (Jokela ym. 2006; Mayhew, 1999; Nielsen, 1993; Preece, 2002; Stone, 2005; Wixon & Wilson, 1997). Tällöin käytettävyyden arviointi joudutaan suorittamaan lähtökohdista, joissa kvantitatiivisia käytettävyystvaatimuksia ei ole käytettävissä käytettävyyssarvioinnin lähtökohtana. Käytettävyyden arviointiin tulisikin olla selkeä vaihe vaiheelta etenevä ja suhteellisen kevyt menetelmä, jonka avulla käytettävyyden arviointi pystyttäisiin suorittamaan täsmällisesti tilanteessa, jossa kvantitatiivisia käytettävyystvaatimuksia ei ole ennalta määritetty. Menetelmä olisi erityisen hyödyllinen, jos sen avulla voitaisiin käyttöliittymän prototyypin käytettävyyssarvioinnin yhteydessä määrittellä kriittiset rajat käytettävyystvaatimuksille. Tällaisessa tapauksessa menetelmää voitaisiin soveltaa käytettävyyssarvioinnin lisäksi myös ”käänteisellä tavalla” käytettävyystvaatimusten määrittelyn tukena. Näistä lähtökohdista määritellään väitöskirjan neljäs tutkimuskysymys.

Tutkimuskysymys 4: (a) Kuinka käytettävyysongelmia voidaan paikantaa täsmällisesti tilanteissa, joissa kvantitatiivisia käytettävyystvaatimuksia ei ole määritetty? b) Kuinka voidaan määrittää kriittisiä rajoja kvantitatiivisille käytettävyystvaatimuksille?

4.2 Tutkimusmetodologia

Tutkimusaihe, tutkimuskysymykset ja tutkimuksen teoreettiset lähtökohdat vaikuttavat jokaisessa tutkimuksessa tutkimusstrategian ja -menetelmien valintaan. Tutkimusaiheen liittyessä ihmisen ja teknologian väliseen vuorovaikutukseen tutkimusta koskevat oletukset ja hypoteesit on pystyttävä osoittamaan oikeiksi empiirisiin kokeisiin, joissa ihminen ja teknologia kohtaavat. Tässä tutkielmassa vuorovaikutusta tarkastellaan erityisesti suunnittelun näkökulmasta (vrt. vaatimusmäärittely, innovatiivinen ohjelmistotuote) ja kehitetään uudentyyppinen lähestymistapa käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyyn ja arviointiin. Tutkimus on aiheeltaan aidosti poikkitieteellinen asettuen ihminen-teknologia-vuorovaikutuksen, vaatimusmäärittelyn, käyttäjäpsykologian ja innovatiivisen ohjelmistotuotteen suunnittelun leikkausalueelle. Tutkimuksen teoreettiset lähtökohdat huomioon ottaen tutkimuskohteesta on aluksi

hahmoteltava teoreettisen tutkimuksen keinoin käsitteellisiä malleja, selityksiä ja rakenteita olemassa olevan tutkimuskirjallisuuden pohjalta. Vasta tämän jälkeen voidaan tarkastella empiirisiin kokein kehitettyjen mallien toimivuutta käytännön olosuhteissa.

Tutkimusstrategian tälle tutkimukselle muodostaa teoreettisen, kvalitatiivisen ja kvantitatiivisen lähestymistavan soveltaminen. Teoreettista lähestymistapaa sovelletaan tutkimuskysymysten Q1, Q2 ja Q3 yhteydessä. Kvalitatiivista lähestymistapaa sovelletaan artikkelissa A3 etsittäessä ratkaisua tutkimuskysymykseen Q2. Kvantitatiivista lähestymistapaa käytetään artikkelissa A6 ratkaistaessa tutkimuskysymystä Q4. Tässä väitöskirjan johdanto-osassa käytetään artikkeleista numerointia A1-A6 ja alaluvussa 4.1 esitetyistä tutkimuskysymyksistä lyhenteitä Q1-Q4. Taulukossa 3 on esitetty yhteenveto tutkimusmenetelmien käytöstä tutkimusongelmien ja artikkelien yhteydessä.

Teoreettinen lähestymistapa

Teoreettisessa tutkimuksessa pyritään hahmottamaan rajattua reaali maailman osaa kuvaava teoria, malli tai viitekehys perustuen aikaisempiin tieteellisiin tutkimustuloksiin (Järvinen & Järvinen, 2004). Teoreettinen tutkimus ei kuitenkaan synnytä teorioita, malleja tai viitekehyskäsitteitä ”itsestään” aiempien tutkimustulosten pohjalta, vaan ne ovat inhimillisen tutkimustoiminnan tuloksia (Hirsjärvi ym. 2007). *Teoria* viittaa yleistyksenä siitä, kuinka luontoon tai ihmiseen liittyvän maailman tietty määritelty osa toimii ja mitä keskeisiä käsitteitä siihen liittyy (Repko, 2012). Teorioiden tarkoituksena voidaan pitää Hirsjärven ym. (2007 s.138) mukaan: 1) tarjota oikotie kommunikoinnille, 2) järjestää ideoita ja paljastaa piileviä oletuksia, 3) luoda uusia ideoita, 4) tuoda esille ongelman monimutkaisuus, 5) luoda selityksiä ja ennusteita, 6) osoittaa näennäisesti erilaisien ongelmien yhteenkuuluvuus. Teorioiden muodostaminen on kuitenkin joskus käytännössä hankalaa, jolloin tutkimustyön apuvälineenä voidaan käyttää malleja (Hirsjärvi ym. 2007). Malli viittaa kuvitelmaan tai abstraktioon todellisuudesta, jonka avulla pyritään yksinkertaistamaan näkemystä kokonaisuudesta tuomalle esille vain olennaiset piirteet (Hirsjärvi ym. 2007 s.141-142). Mallin kuvaamat olennaiset piirteet voidaan kuvata joukolla propositioita tai lauseita, jotka ilmaisevat käsitteiden väliset suhteet (Järvinen & Järvinen, 2004). Tarkastelemalla mallin ja teorian suhdetta voidaan nähdä, että malli on ikään kuin teorian jalostamaton esiaste, josta on mahdollista kehittyä ja kehittää teoria yhdessä hypoteesin, tutkimuksen toteutuksen ja tulosten analysoinnin seurauksena (Hirsjärvi ym. 2007). Viitekehys tarkoittaa tutkimusongelman täsmällistä esittämistä käsitteellisesti (Alkula ym. 1994 s.34). Osana teoreettista tutkimusta viitekehyskäsitteelle voidaan määritellä ainakin kaksi tehtävää (Alkula ym. 1994): 1) viitekehyskäsitteellä voidaan rajata ja täsmentää tutkimuksessa tarkasteltavia asioita ja 2) viitekehys auttaa muodostamaan tietyn näkökulman ja juonen tutkimukselle sulauttaen tutkimuksen konkreettiseksi osaksi yhteiskuntaa. Teoreettinen tutkimuksen lähestymistapa on laaja kokonaisuus, joka sisältää ainakin seuraavat tutkimussuuntaukset (Uusitalo, 1999): teoreettinen katsaus, käsi-

teanalyttinen tutkimus, teoreettinen formaali mallintaminen, olemassa olevan teorian kehittäminen ja uuden teorian kehittäminen.

TAULUKKO 3 Tutkimusmenetelmät tutkimuskysymysten ja artikkeleiden yhteydessä

Tutkimusmenetelmä	Tutkimuskysymys	Artikkeli
Teoreettinen lähestymistapa	(Q1) Kuinka vaatimusten määrittely voidaan suorittaa intentiolähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta?	A1 A framework for intention-driven requirements engineering of innovative software products
Teoreettinen lähestymistapa	(Q2) Kuinka käyttäjäpsykologiaa ja intentiolähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusten määrittelyä?	A2 User psychology in interaction design - The role of design ontologies
Kvalitatiivinen lähestymistapa	(Q2) Kuinka käyttäjäpsykologiaa ja intentiolähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusten määrittelyä?	A3 Ontologies for human-technology interaction design
Teoreettinen lähestymistapa	(Q3) Kuinka käytettävyyttä ja käyttäjäkokemuksvaatimuksia voidaan määrittää osana kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vuorovaikutussuunnitelua?	A4 Towards Action-Oriented user interface design A5 Interaction deconstruction method for usability requirements engineering
Kvantitatiivinen lähestymistapa	(Q4) (a) Kuinka käytettävyysongelmia voidaan paikantaa täsmällisesti tilanteissa, joissa kvantitatiivisia käytettävyyksvaatimuksia ei ole määritelty? (b) Kuinka voidaan määrittää kriittisiä rajoja kvantitatiivisille käytettävyyksvaatimuksille?	A6 A quantitative method for localizing user interface problems: D-TEO Method

Tässä tutkimuksessa teoreettista tutkimusta käytetään artikkeleissa A1, A2, A4 ja A5. Artikkelissa A1 tutkitaan, millaisia haasteita innovatiivisen ohjelmistotuotteen vaatimusmäärittelyyn liittyy ja kehitetään olemassa olevien tutkimusten pohjalta viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi. Tutkimuksessa hyödynnetään teoreettista katsausta ja käsiteanalyttistä tutkimusta. Artikkelissa A2 selvitetään olemassa olevaan kirjallisuuteen perustuen ihmisen-teknologia-vuorovaikutuksen järjestystä ja rakennetta vuorovaikutustilan-

teessa ja hahmotellaan ihminen-teknologia-vuorovaikutusontologiaa. Tutkimuksessa sovelletaan teoreettisen tutkimuksen näkökulmasta käsiteanalyttistä tutkimusta ja uuden teorian kehittämistä. Artikkeleissa A4 ja A5 perehdytään käytettävyy- ja käyttäjäkokemusvaatimusten määrittämisen ongelmiin sekä kehitetään menetelmää, jonka avulla aikaisempien tutkimusten kautta esille nousseisiin haasteisiin pyritään vastamaan. Tutkimuksessa hyödynnetään teoreettisen tutkimuksen näkökulmasta teoreettista katsausta ja uuden teorian kehittämistä.

Kvalitatiivinen lähestymistapa

Kvalitatiivisen tutkimuksen päämääränä on ymmärtää tutkimuskohdetta ja tutkittavaa ilmiötä mahdollisimman kokonaisvaltaisesti (Hirsjärvi ym. 2007). Kvalitatiivinen tutkimus soveltuu erityisen hyvin tutkimukseen, jossa määritellään ja luokitellaan erilaisia luokkia ja kategorioita sekä tutkitaan niiden välisiä suhteita (Dey, 1993; McCracken, 1988). Kvalitatiivista tutkimusta voidaan myös käyttää tutkimusstrategiana teorioita testaavassa tutkimuksessa joko itsenäisenä strategiana tai vaihtoehtoisesti esimerkiksi kvantitatiivisen tutkimuksen rinnalla (Järvinen & Järvinen, 2004). Kvalitatiivisen tutkimuksen juuret ovat antropologiassa ja sosiologiassa pyrkien tuomaan esille, että inhimillistä toimintaa ei voida ymmärtää pelkästään kvantitatiivisen ja tilastollisen tutkimuksen perusteella (Holliday, 2008). Inhimillistä käytöstä hallitsevien subjektiivisten ominaisuuksien tutkiminen on tällöin välttämätöntä (Holliday, 2008). Hirsjärven ym. (2007) mukaan kvantitatiivisen tutkimusstrategian kehittymistä ja laajuutta kuvaa hyvin se, että sosiaalitieteilijät käyttivät ennen termin ”kvalitatiivinen” yleistymistä tutkimustoimiensa yhteydessä käsitteitä ”kenttätyö” ja ”osallistuva havainnointi”, jotka nähdään nykyisin vain osana kvalitatiivisen tutkimusstrategian kokonaisuutta.

Tässä tutkimuksessa kvalitatiivista lähestymistapaa sovellettiin artikkelin A3 yhteydessä siten, että aineistonhankintamenetelmänä käytettiin ITEA2-hankkeen Easy Interactions -projektissa mukana olleilta yrityksiltä kerättyjä valmiita dokumentteja ja aineiston analyysimenetelmänä sovellettiin kvalitatiivista sisällönanalyysiä. Kvalitatiivinen sisällönanalyysi koostui Metsämuurosen (2008) ja Syrjäläisen (1994) mukaisesti seitsemästä eri vaiheesta: 1) Tutkijan ”herkistyminen” (tällä viitataan oman aineiston perinpohjaiseen tuntemiseen ja keskeisten käsitteiden haltuunottoon teoreettisen kirjallisuuden avulla), 2) Aineiston sisäistäminen ja teoretisointi, 3) Aineiston karkea luokittelu, keskeiset luokat/teemat, 4) Tutkimustehtävän täsmennys, käsitteiden täsmennys, 5) Ilmiöiden esiintymistiheyden toteaminen, poikkeusten toteaminen, uusi luokittelu, 6) Ristiinvalidointi, saatujen luokkien puoltaminen ja horjuttaminen aineiston avulla sekä 7) Johtopäätökset ja tulkinta. Artikkelin A3 keskeisenä tavoitteena oli testata, kuinka hyvin artikkelissa A2 esitetty teoreettisista lähtökohdista kehitetty ihminen-teknologia-vuorovaikutusontologia vastasi todellisuutta siinä esitettyjen luokkien ja kategorioiden sekä niiden määrittelyjen ja luokittelujen suhteen. Tällöin edellä esitetyn sisällönanalyysin vaiheet 1-3 oli aloitettu jo työ-

tettäessä artikkeleita A2 teoreettisista lähtökohdista. Vaiheet 4-7 kohdistuivat puolestaan ihminen-teknologia-vuorovaikutusontologian luokkien ja kategorioiden testaamiseen ja teoreettiseen kehittämiseen päämääränä kuvata täsmällisemmin ihmisen toiminnan keskeiset tekijät osana ihminen-teknologia-vuorovaikutustilannetta.

Kvantitatiivinen lähestymistapa

Kvantitatiivinen tutkimus on määrällistä tutkimusta, jossa tutkimuskohdetta tarkastellaan numeerisen mittaamisen ja tilastollisen analyysiin kautta (Goodwin, 2008; Repko, 2012). Käyttäytymistieteiden näkökulmasta kvantitatiivisessa lähestymistavassa korostuvat (Kline, 2009): 1) ihmisen käyttäytymisen numeerinen mittaaminen, 2) ihmisen käyttäytymistä kuvaavan havaintoaineiston tilastollinen analysointi ja 3) tutkijan rooli passiivisena tarkkailijana. Kvantitatiivinen tutkimus ei kuitenkaan rajoitu pelkkiin numeroihin ja analyysihin, eivätkä numerot ole itsessään kvantitatiivisen tutkimuksen päämäärä. Päämääränä on pikemminkin liittää numerot osaksi todellisuutta, jossa elämme tuoden syvällistä tutkimusongelman mukaista uutta tietoa tutkittavasta ilmiöstä. (Jokivuori & Hietala, 2007) Kvantitatiivisen tutkimuksen vahvuutena on mahdollisuus kuvata tutkittavan ilmiön rakenne ja koostumus tarkalla tasolla sekä selvittää, minkälaisia yhteyksiä ilmiön eri osien välillä on ja määritellä minkälaisia muutoksia tutkittavassa ilmiössä tapahtuu (Alkula ym. 1994). Toinen kvantitatiivisen tutkimuksen vahvuus on menettelytapojen edellyttämä eksplikointi (Alkula ym. 1994). Käytännössä eksplikointi, eli asioiden selvä ja perusteellinen selittäminen edellyttää esimerkiksi tutkimuskohteeseen liittyvien käsitteiden, teorioiden ja aiempien tutkimusten (Hirsjärvi ym. 2007) tarkkaa tuntemusta, jotka osaltaan parantavat tutkimuksen luotettavuutta (Alkula ym. 1994). On kuitenkin muistettava, että kvantitatiivisessa tutkimuksessa tietojen kerääminen keskittyy tyypillisesti tutkittavan ilmiön rakenteisiin ja prosesseihin vain rajoitetuin osin, sillä ilmiöstä eristetään vain mitattavat osat. Tällöin tutkimus ei anna kuvaa tutkimuskohteesta kokonaisuutena. (Alkula ym. 1994).

Tässä tutkimuksessa kvantitatiivista lähestymistapaa sovellettiin artikkelin A6 yhteydessä, kun etsittiin vastausta tutkimuskysymykseen Q4. Kyseisessä artikkelissa arvioitiin D-TEO -menetelmän soveltuvuutta käytännön olosuhteisiin suorittamalla empiirinen käyttäjäkoe, johon osallistui 10 koehenkilöä. Osana käyttäjäkoetta mittasimme esimerkiksi aikoja, joita käyttäjät kuluttavat yksittäisten alitehtävien suorittamiseen, ja virheiden määriä, joita käyttäjät tekevät navigoidessaan WWW-sivustoilla. Edelleen, yksityiskohtaista mittausdataa ja lostettiin muun muassa todennäköisyyslaskennan menetelmien ja teorioiden avulla. Käyttäjäkokeessa käytettävyyttä tarkasteltiin aikojen mittaamisen ja virhemäärien näkökulmasta jättäen huomioimatta esimerkiksi käyttäjien subjektiiviset mielipiteet testattavasta järjestelmästä. Toinen kvantitatiiviselle tutkimukselle tyypillinen rajausta tehtiin siihen, että käyttäjäkokeessa testitettävät laadittiin vastaamaan vain eksplisiittisiä tiedonhakutehtäviä jättäen ulkopuolelle implisiittiset tiedonhakutehtävät. Tällä tavoin pyrittiin tutkimaan käytettävyyden

rakennetta ja yksittäisiä käytettävyyden attribuutteja sekä niiden välisiä suhteita vain tietystä näkökulmasta eristämällä muut käytettävyyden attribuutit ja käyttäjäkokemus käyttäjäkokeen ulkopuolelle.

4.3 Tutkimuksen reliabiliteetti ja validiteetti

Tutkimuksen luotettavuuden arvioinnissa voidaan käyttää monia erilaisia mittaus- ja tutkintatapoja, joista yleisimmät ovat reliabiliteetti ja validiteetti (Goodwin, 2008; Hirsjärvi ym. 2007). Reliabiliteetti ja validiteetti muodostavat yhdessä mittarin kokonaisluotettavuuden (Alkula ym. 1994). Edellinen viittaa tutkimustulosten toistettavuuteen, eli mittauksen tai tutkimuksen kykyyn antaa ei-sattumanvaraisia tuloksia (Hirsjärvi ym. 2007; Järvinen & Järvinen, 2004).

Validiteetti tarkoittaa mittarin tai tutkimusmenetelmän kykyä mitata juuri sitä, mitä on tarkoituskin mitata (Goodwin, 2008; Hirsjärvi ym. 2007). Validiutta voidaan tarkastella eri näkökulmista, joita ovat esimerkiksi (Alkula ym. 1994): ennustevalidius, sisällöllinen validius, rakennevalidius ja prosessivalidius. Ennustevalidius tarkoittaa mittarin kykyä mitata oikein jotakin mittarin ulkopuolista ilmiötä. Sisällöllinen validius viittaa siihen, että mittarin tulee olla tutkittavan käsitteen näkökulmasta järkevä ja perusteltu, jotta sillä voidaan mitata juuri tarkoitettua asiaa. Rakennevalidius on laaja käsite, jolla viitataan siihen, kuinka hyvin tutkimuksen tulokset vastaavat tutkimukselle asetettuja odotuksia. Prosessivalidius tarkoittaa kaikkien validiuden kannalta keskeisten vaiheiden (esim. käsitteanalyysi, operationalisointi, mittaus) mahdollisimman tarkkaa kuvausta.

Vuorovaikutusontologiaa kehitettäessä empiirisen tutkimuksen reliabiliteetti huomioitiin siten, että kolme tutkijaa suoritti kvalitatiivisen sisällön analyysin. Tällaista menettelytapaa voidaan kutsua myös tutkijatriangulaatioksi (Hirsjärvi ym. 2007). Lisäksi sisällön analyysin aineistona olevat dokumentit kerättiin ITEA2-hankkeen Easy Interactions -projektissa mukana olleilta yrityksiltä siten, että yritykset edustivat useita eri toimialoja sekä toimintakonteksteja. Validiteetin näkökulmasta tutkimuksen totuusarvoa tukee se, että vuorovaikutusontologiaan liittyvät käsitteet oli määritelty ja jäsennetty jo artikkelin A2 yhteydessä, joten käsitteiden teoreettinen määrittely (A2) ja empiirinen mittaaminen (A3) olivat selvästi toisistaan erilliset työvaiheet. Tällä tavoin pyrittiin suorittamaan erityisen huolellisesti keskeisten käsitteiden hahmottaminen ja määrittäminen, käsitteiden osa-alueiden määrittäminen ja käsitteiden välisten suhteiden määrittäminen. Tutkimuksen soveltuvuus tarkoitukseensa varmistettiin sekä keräämällä tutkimusaineisto eri toimialojen konteksteista että hyödyntämällä eri tutkimusalojen asiantuntijoita tutkimusprosessin eri vaiheissa. Silloin kun samaa ilmiötä lähestytään eri tieteenalojen teorioiden näkökulmasta, käytetään usein myös termiä teoreettinen triangulaatio (Hirsjärvi ym. 2007). Tulosten sovellettavuus innovatiivisen teknologian kehittämiseen arvioitiin hyväksi, sillä suurin osa analyysin aineistona olevista dokumenteista käsitteli nimenomaan

innovatiivisten teknologiaratkaisujen kehittämistä eri toimialoilla mukaan lukien ohjelmistotuotteiden kehittäminen.

D-TEO-menetelmää kehitettäessä tutkimuksen reliabiliteetti huomioitiin määrittämällä tutkittavan ilmiön kannalta keskeiset käsitteet sellaisiksi käsitteiksi, joita voidaan mitata. Tällaisesta käsitelmäärittelystä käytetään nimitystä operationalisointi (Alkula ym. 1994). Operationalisointi koostuu Alkulan ym. (1994) mukaan seuraavista vaiheista: 1) keskeisten käsitteiden hahmottaminen ja määrittäminen, 2) kokonaisuuden hahmottaminen määrittämällä keskeisten käsitteiden eri osa-alueet, koostumus ja niiden muodostama kokonaisuus, 3) siirtyminen teoreettisesta kielestä mitattaviin indikaattoreihin, 4) operationalisoinnin tarkka kuvaaminen, jonka kautta mittaus tehdään näkyväksi. D-TEO -menetelmää kehitettäessä operationalisointi suoritettiin seuraavasti. Keskeisten käsitteiden hahmottaminen ja määrittäminen aloitettiin perehtymällä aikaisempiin tutkimuksiin ja määritelmiin rakentaen kehitettävän menetelmän keskeiset vaiheet niiden pohjalta. Teorianäkökulmasta keskityttiin käytettävyyden käsitteen määritelmiin ja käytettävyyden koostumukseen sekä vastaavanlaisiin olemassa oleviin menetelmiin. Käytettävyyden attribuuteista keskityttiin tehokkuuteen korostaen eksplisiittisten tiedonhakutehtävien näkökulmaa (engl. search efficiency). Seuraavaksi tehokkuutta päätettiin mitata kahdella mittarilla, jotka olivat virheet ja aika. Tällä tavoin siirryttiin kohti konkreettista mittaria määritellen samalla käytettävyyden ja sen attribuuttien koostumus. Varsinaisessa käyttäjäkokeessa aikamittaukset kohdistettiin yksittäisiä alitehtäviä kohden kulutettuihin aikoihin. Vastaavasti virhemittaukset kohdistettiin virheiden määriin, joita käyttäjät tekivät navigoidessaan WWW-sivustolla suhteessa optimaaliseksi määriteltäviin navigointipolkuihin. Tällä tavoin määriteltiin joukko indikaattoreita, jotka sopivat mittausoperaation välineiksi. Operationalisoinnin tarkka kuvaus oli rakennettu sisään D-TEO -menetelmän neljään yksittäiseen vaiheeseen: 1) testin päämäärien ja käyttäjäprofiilien määrittäminen, 2) testitehtävien suunnittelu ja optimaalisten polkujen määrittäminen, 3) testien järjestäminen ja datan kerääminen ja 4) datan analysointi ja johtopäätösten tekeminen. D-TEO -menetelmää kehitettäessä operationalisointi tuotti helppokäyttöiset ja yksiselitteiset muuttujat, joita eri tutkijat voivat käyttää samalla tavalla tutkimustensa yhteydessä.

Tutkimuksen validiteetin näkökulmasta D-TEO -menetelmää voidaan lähestyä seuraavalla tavalla. Menetelmän avulla käytettävyysoongelmia paikannetaan yksittäisten operaatioiden tarkkuudella vertaamalla koehenkilöiden todellisia navigointipolkuja suhteessa optimaaliseksi määriteltäviin navigointipolkuihin. Vertailu navigointipolkujen välillä tehdään keskittyen käytettävyyden attribuuteista tehokkuuteen eksplisiittisten tiedonhakutehtävien näkökulmasta. Tämä on kirjallisuudessa tyypillinen ja perusteltu tapa lähestyä navigointitarkkuuteen ja tiedonhakuun liittyviä käytettävyyden erityiskysymyksiä (katso esim. Bernard, 2002; Norman & Chin, 1988). Käytettävyysongelmien paikantaminen D-TEO -menetelmällä rajoittuu ISO 9241-11 (1998) standardin määritelmään käytettävyydestä ja painottuu erityisesti käytettävyyden tehokkuusattribuuttiin. Käytettävyyttä ei tällöin lähestytä esimerkiksi opittavuuden tai tyyty-

väisyyden näkökulmasta. Toisaalta D-TEO -menetelmän kehittämiseen liittyvän empiirisen tutkimuksen tarkoitus on ollut havainnollistaa ja testata menetelmän käyttöä eikä tehdä sille kattavaa toimivuuden ja hyödyllisyyden arviointia.

5 TUTKIMUKSEEN SISÄLLYTETYT ARTIKKELIT

Tässä luvussa esitellään tiivistetysti väitöskirjaan sisältyvät tieteelliset artikkelit käymällä läpi kunkin tutkimuksen osalta tutkimuksen kohde, tavoite, tulokset ja väitöskirjan tekijän kontribuutio tutkimuksen työstämiseen. Ensimmäisessä tutkimuksessa (A1) kehitetään viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi vastaamaan tyypillisiin vaatimusmäärittelyn haasteisiin, joita kohdataan kehitettäessä innovatiivista ohjelmistotuotetta. Toisessa tutkimuksessa (A2, A3) hahmotellaan ihminen-teknologia-vuorovaikutusontologiaa, jota voidaan hyödyntää osana innovatiivisen teknologian suunnittelua erityisesti käyttäjätarpeiden tunnistamisen tukena. Kolmannessa tutkimuksessa (A4, A5) esitetään Interaction Deconstruction -menetelmä käytettävyyss- ja käyttäjäkokemusvaatimusten määrittämisen tueksi. Neljännessä tutkimuksessa (A6) esitetään D-TEO -menetelmä käytettävyysohjelmien paikantamisen ja käytettävyyttä koskevien kvantitatiivisten vaatimusten kriittisten raja-arvojen määrittelyn tueksi.

5.1 Tutkimus I: Viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi

Leppänen, M., Lamminen, J. & Saariluoma, P. 2011. A Framework for intention-driven requirements engineering of innovative software products. Teoksessa J. Pokorny, V. Repa, K. Richta, W. Wojtkowski, H. Linger, C. Barry & M. Lang (toim.) Information Systems Development. New York: Springer, 417-428.

Artikkelissa A1 tutkimuksen kohteena on innovatiivisen ohjelmistotuotteen vaatimusmäärittely. Tutkimuksen tavoitteena oli selvittää, kuinka vaatimusten määrittely voidaan suorittaa intentiolähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta. Motivaationa tutkimuksen taustalla olivat erityisesti innovatiivisen ohjelmistotuotteen asettamat haasteet vaatimusmäärittelylle sekä tarve hyödyntää intendentietämystä nykyistä laajemmin informaatioteknologian alalla ja lähestyä vaatimusmäärittelyä mahdollisimman teknologiariippumattomasta

näkökulmasta. Tutkimuksen tuloksena esiteltiin viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi.

Viitekehys koostuu viidestä vaatimusmäärittelyn aktiviteetista: visiointi, käyttäjäprofilointi, liiketoimintakontekstin analysointi, sosiaalisen kontekstin analysointi ja päämäärän asettaminen. Viitekehyksessä sovelletaan ideaa siitä, että vaatimukset tulisi johtaa toimijoiden intentioista siten, että huomioidaan sekä yksilö- että ryhmänäkökulma. Myöhemmässä vaiheessa prosessia intentionäkökulman lisäksi huomioidaan myös liiketoimintänäkökulma. Tällä tavoin innovatiivisen teknologian suunnitteluun on mahdollista integroida perusteellinen tietämys siitä, miksi uutta teknologiaa todella ollaan kehittämässä ja mihin sitä todella tarvitaan.

Tutkimuksessa esitetty viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi tarjoaa lupaavan mahdollisuuden vastata vaatimusmäärittelyn haasteisiin kehitettäessä innovatiivista ohjelmistotuotetta. Intentiolähtöisen vaatimusmäärittelyn kehittämiseksi on tarvetta erityisesti niiden vaatimusmäärittelyn haasteiden ratkaisemiseksi, jotka kohdistuvat vaatimusmäärittelyn alkuvaiheisiin. On kuitenkin huomattava, että tässä artikkelissa esitetty viitekehys on kehitetty pääosin kirjallisuuden perustuen eikä sitä ole empiirisesti arvioitu.

Väitöskirjan tekijänä osallistuin aktiivisesti tutkimukseen ollen mukana viitekehysten ideoinnissa ja kehittämisessä, laatimalla käytännön esimerkit viitekehysten soveltamisesta käytäntöön, toimimalla vaatimusmäärittelyn asiantuntijana, osallistumalla artikkelin kirjoitusprosessiin ja esittämällä tutkimuksen tulokset Prahassa järjestetyssä ISD 2010 -konferenssissa.

5.2 Tutkimus II: Käyttäjätarpeiden tunnistaminen ihmisen-teknologia-vuorovaikutusontologian avulla kehitettäessä innovatiivista ohjelmistotuotetta

Saariluoma, P., Parkkola, H., Honkaranta, A., Leppänen, M., & Lamminen, J. 2009. User psychology in interaction design – the role of design ontologies. Teoksessa P. Saariluoma & H. Isomäki (toim.) *Future Interaction Design II*, Berlin: Springer, 69-86.

Saariluoma, P., Lamminen, J., & Leppänen, M. 2009. Ontologies for human-technology interaction design. Teoksessa *Proceedings of the 16th International Product Development Management Conference*, Twente, The Netherlands: European institute for advanced studies in management, EIASM.

Artikkeleissa A2 ja A3 tutkimuksen kohteena on käyttäjätarpeiden tunnistaminen osana innovatiivisen teknologian kehittämistä. Tutkimuksen tavoitteena oli selvittää, kuinka käyttäjäpsykologiaa ja intentiolähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusmäärittelyä. Motivaationa tutkimuksen taustalla olivat myös kirjallisuus-

dessa esitetyt käyttäjätarpeiden tunnistamisen ongelmat (Lindgaard ym. 2006) ja pyrkimys ymmärtää perusteellisemmin tarpeen käsite ja ihmisen toiminnan rakenne. Tutkimuksen tuloksena esittelimme kevytrakenteisen (engl. light-weight) ontologian (katso esim. Gómez-Pérez ym. 2004) jäsentämään ihmisen ja teknologian välisen vuorovaikutuksen keskeiset käsitteet ja käsitteiden väliset suhteet ihmisen toiminnan näkökulmasta. Käytimme ensin teoreettista lähestymistapaa määrittellessämme toiminnan rakenteen vuorovaikutustilanteessa (A2). Tämän jälkeen valitsimme tutkimusmenetelmäksi kvalitatiivisen sisältöanalyysin (Metsämuuronen, 2008; Syrjäläinen, 1994) ja tarkensimme toiminnan rakennetta (A3). Tutkimus suoritettiin siten, että aineistonhankintamenetelmänä käytettiin ITEA2-hankkeen Easy Interactions -projektissa mukana olleilta yrityksiltä kerättyjä valmiita dokumentteja ja aineiston analyysimenetelmänä sovellettiin kvalitatiivista sisällönanalyysiä.

Tutkimus osoittaa, että käyttäjätarpeiden perusteellinen ymmärtäminen edellyttää ihmisen toiminnan ymmärtämistä ihminen-teknologia vuorovaikutustilanteessa yksittäisten toiminnan tekijöiden tarkkuudella. Esimerkiksi toiminnan psykologian näkökulmasta (Gollwitzer & Bargh, 1996) intentio, motivaatio ja arvot ovat yksittäisiä keskeisiä toiminnan tekijöitä, joiden avulla käyttäjätarpeita on mahdollista ymmärtää perusteellisemmin. Perustuen teoreettiseen lähestymistapaan (A2) ja kvalitatiiviseen sisältöanalyysiin (A3) nähdään, että ihminen-teknologia vuorovaikutustilanteen kannalta keskeiset toiminnan tekijät voidaan jakaa kahteen kategoriaan ihminen-teknologia-vuorovaikutusontologian avulla. Ensimmäinen kategoria koostuu neljästä käsitteestä (toimija, toiminnan kohde, väline, konteksti), jotka liittyvät suoraan ihmisen toimintaan. Toisen kategorian käsitteet ovat ensimmäisen kategorian käsitteiden attribuutteja. Tutkimuksemme tulokset tukevat ihmisen toimintaa käsittelevän psykologian alalla vallitsevaa käsitystä siitä, että intentio on motivaation, arvojen ja tarpeiden ohella yksi keskeisin ihmisen toimintaa ohjaava tekijä (esim. Ajzen, 1985; Fishbein & Ajzen, 1975; Bratman, 1987; Card ym. 1983; Gollwitzer & Bargh, 1996; Hampshire, 1965; Irwin, 1971; James, 1890; Lewin, 1951; Meiland, 1970; Miller ym. 1960, Monk, 1998; Norman, 1986; Shultz, 1980; Tolman, 1932).

Kvalitatiivisen sisältöanalyysin (A3) perusteella intentio kannattaa jakaa vuorovaikutustilanteessa edelleen mitä-intentioon ja kuinka-intentioon. Sisältöanalyysissä tämä ilmeni siten, että ihmisen toiminta voidaan eritellä vuorovaikutustilanteessa teknologiariippumattomaan ja teknologian avulla suoritettuun toimintaan. Mitä-intentio viittaa siihen, mitä toimija haluaa tehdä (erityisesti teknologiariippumattomasti). Kuinka-intentio viittaa siihen, millä tavoin mitä-intentio voidaan toteuttaa teknologian avulla. Tunnistettaessa käyttäjätarpeita ihminen-teknologia-vuorovaikutusontologian avulla on keskeistä selvittää myös sellaiset ihmisen toiminnat, joita ei kannata teknologian avulla tukea. Intention jakaminen mitä-intentioon ja kuinka-intentioon määrittelee eron aidosti ihmislähtöisen ja teknologianlähtöisen lähestymistavan välille osana vuorovaikutussuunnittelua. Tämä määritelmä auttaa jäsentämään monissa tutkimuksissa (esim. Leikas, 2009; Parkkola, 2006; Viukari, 2010) esille nostettua ongelmaa ihmislähtöisen ja teknologialähtöisen vuorovaikutussuunnittelun eroista. Kar-

toittaessamme toiminnan keskeiset attribuutit toiminnan analyysin avulla on mahdollista löytää uudenlaisia tapoja tukea ihmisen toimintaa. Tämä saattaa johtaa edelleen innovatiivisten ohjelmistotuotteiden kehittämiseen. Ihminen-teknologia-vuorovaikutusontologia tarjoaa ontologiapohjaisen ratkaisun innovatiivisen ohjelmistotuotteen vaatimusmäärittelyn tueksi. Kehittämämme ontologia perustuu käyttäjäpsykologiaan.

Väitöskirjan tekijänä osallistuin aktiivisesti ihminen-teknologia-vuorovaikutusontologian kehittämiseen ja ideointiin, tutkimusartikkelien kirjoittamiseen ja keskeisen kirjallisuuden läpikäyntiin. Toimin artikkelien työstämisen aikana myös tutkijana ITEA2-hankkeen Easy Interactions -projektissa, jonka keskeisiin tuloksiin ihminen-teknologia-vuorovaikutusontologia itse asiassa voidaan lukea.

5.3 Tutkimus III: Interaction Deconstruction -menetelmä käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyyn

Lamminen, J. & Saariluoma, P. 2010. Towards action-oriented user interface design. Teoksessa Proceedings of the 2010 International Conference on Management Science and Engineering, Hong Kong: ETP Engineering Technology Press, 58-61.

Lamminen, J., Rousi, R. & Saariluoma, P. 2011. Interaction deconstruction method for usability requirements engineering. Teoksessa K. Blashki (toim.) Proceedings of the IADIS International Conference Interfaces and Human Computer Interaction 2011, Portugali: IADIS Press, 161-168.

Artikkeleissa A4 ja A5 tutkimuksen kohteena on käytettävyy- ja käyttäjäkokemusvaatimusten määrittäminen kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta. Tutkimuksen tavoitteena oli selvittää, kuinka käytettävyy- ja käyttäjäkokemusvaatimuksia voidaan määrittää tällaisessa suunnittelutilanteessa. Artikkelissa A4 tutkimusta lähestyttiin ihmisen toiminnan näkökulmasta, ja artikkelissa A5 tutkimusta jatkettiin sekä syvennettiin kehittämällä käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyyn tarkoitettu käytännön menetelmä. Motivaationa tutkimukselle olivat haasteet, joita vaatimusmäärittelyssä kohdataan kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta. Motivaationa oli myös käyttäjäpsykologian ja intentiolähtöisyyden kehittäminen osana vuorovaikutussuunnittelua. Tutkimuksen tuloksena esiteltiin Interaction Deconstruction -menetelmä käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyn tueksi.

Interaction Deconstruction -menetelmä koostuu kolmesta vaiheesta: keskeisten toimintojen määrittely, hierarkkisen puurakenteen muodostaminen keskeisistä toiminnoista ja käytettävyyssattribuuttien asettaminen hierarkkisen puun solmuille.

Keskeisten toimintojen määrittämisellä (vaihe 1) viitataan ihminen-teknologia-vuorovaikutusta kuvaavien ihmisen toimintojen määrittämiseen. Interaction Deconstruction -menetelmän lähtökohtana on tilanne, jossa keskeiset toiminnot on jo selvitetty soveltuvalla menetelmällä tai tekniikalla. Tällaisia menetelmiä ja tekniikoita ovat esimerkiksi viitekehys intentiolähtöisen vaatimusmäärittelyn tueksi (A1) ja ihminen-teknologia-vuorovaikutusontologia (A2 ja A3). Hierarkkisen puurakenteen muodostamisella keskeisistä toiminnoista (vaihe 2) tarkoitetaan vaiheessa 1 määriteltyjen keskeisten toimintojen kuvaamista solmuista ja operaatioista koostuvan hierarkkisen puurakenteen avulla. Tällainen puurakenne määrittelee, kuinka ihminen suorittaa yksittäisiä vuorovaikutustehtäviä käyttöliittymällä ja millaisista yksittäisten toimintojen ja intentioiden ketjuista toiminnot koostuvat. Käsittelmällä intentioita yksittäisten operaatioiden tasolla annamme Millerin ym. (1960) intention määritelmälle operaatiotason merkityksen. Käytettävyyssattribuuttien asettaminen hierarkkisen puun solmuille (vaihe 3) on Interaction Deconstruction -menetelmän keskeisin vaihe, joka erottaa sen merkittävästi muista käytettävyyssvaatimusten määrittelymenetelmistä ja tekniikoista. Tällä vaiheella viitataan siihen, että hierarkkisen puun solmuille asetetaan relevantit käytettävyyden ja/tai käyttäjäkokemuksen attribuutit ja solmujen sisältö analysoidaan käyttäjäpsykologian näkökulmasta. Käytännössä tämä tarkoittaa käytettävyyss- ja käyttäjäkokemusvaatimusten määrittämistä rinnakkain tavalla, joka analysoi vaatimusten sisällön modernin psykologian tutkimustiedon valossa hyödyntämällä psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja.

Tutkimus osoittaa, että käytettävyyteen ja käyttäjäkokemukseen liittyvä vaatimusten määrittely on puutteellista sekä keskeisen kirjallisuuden että käytännön työn näkökulmasta. Interaction Deconstruction on lupaava menetelmä vastaamaan moniin vaatimusmäärittelyn haasteisiin tällä alueella. Menetelmän keskeisin parannus verrattuna olemassa oleviin menetelmiin ja tekniikoihin tulee esille vaiheessa 3. Tässä vaiheessa hierarkkisen puun solmuille asetetaan relevantit käytettävyyss- ja käyttäjäkokemusattribuutit sekä analysoidaan vaatimusten sisältö hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Käyttäjäkokemuksen suunnittelua on kirjallisuudessa pidetty hyvin haastavana suunnittelun näkökulmasta, koska ihmisen kokemukset ovat aina yksilökohtaisia. Interaction Deconstruction -menetelmä pyrkii kuitenkin liittämään käytettävyyttä kuvaavien attribuuttien lisäksi myös käyttäjäkokemusattribuutit osaksi käyttöliittymän suunnittelua. Tämä antaa lupaavan ja uudenlaisen mahdollisuuden suunnitella käytettävyyden lisäksi myös käyttäjäkokemuksia. Käyttäjäkokemuksien suunnittelu on puolestaan hyvin keskeistä, kun kyse on innovatiivisista ohjelmistotuotteista.

Väitöskirjan tekijänä ideoin ja kehitin Interaction Deconstruction -menetelmän ja olin päävastuussa kummankin artikkelin A4 ja A5 kirjoittamisesta. Artikkelin A5 tulokset esitin Roomassa järjestetyssä IADIS 2011-konferenssissa.

5.4 Tutkimus IV: D-TEO -menetelmä innovatiivisen ohjelmistotuotteen käytettävyyksvaatimusten määrittelyn tukena

Lamminen, J., Leppänen, M., Heikkinen, R., Kämäräinen, A. & Jokisuu, E. 2009. A quantitative method for localizing user interface problems: D-TEO Method. *Human Technology Journal*, 5 (2), 121-145.

Tässä artikkelissa tutkimuksen kohteena on käytettävyyksvaatimusten määrittelyn ja arvioinnin rajapinta vuorovaikutussuunnittelun näkökulmasta kehitettäessä innovatiivista ohjelmistotuotetta. Tutkimuksen tavoitteena oli selvittää kaksi innovatiivisen ohjelmistotuotteen käytettävyyksvaatimusten määrittelyn ja arvioinnin erityisongelmaa: a) käytettävyyksongelmien paikantaminen tilanteissa, joissa kvantitatiivisia käytettävyyksvaatimuksia ei ole määritelty, ja b) kriittisten rajojen määrittäminen kvantitatiivisille käytettävyyksvaatimuksille. Keskeisenä motivaationa tutkimukselle olivat ne vaatimusmäärittelyn haasteet, joilla on vaikutusta käytettävyyden arviointiin. Tutkimuksen tuloksena esiteltiin D-TEO (Decomposition of Tasks into Elementary Operations) -menetelmä, joka koostuu neljästä vaiheesta: 1) testin päämäärien ja käyttäjäprofiilien määrittäminen, 2) testitehtävien suunnittelu ja optimaalisten polkujen määrittäminen, 3) testien järjestäminen ja datan kerääminen ja 4) datan analysointi ja johtopäätösten tekeminen. D-TEO -menetelmä on erityisen hyödyllinen tilanteissa, joissa kehitetään kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta. D-TEO -menetelmän toimivuutta testattiin koetilanteessa, jossa suunniteltiin ja arvioitiin kuluttajamarkkinoille suunnatun WWW-sivuston käytettävyyttä. Käyttäjäkokeeseen osallistui 10 koehenkilöä.

D-TEO -menetelmä eroaa olemassa olevista käytettävyyden arviointimenetelmistä kolmella tavalla. D-TEO -menetelmä on kohtalaisen kevyt menetelmä verrattuna muihin menetelmiin, joilla käytettävyyksongelmia pyritään paikantamaan yksittäisten operaatioiden tarkkuudella. Tämä johtuu erityisesti siitä, että kokonaisvaltaisen tehtävämallinnuksen sijaan vain optimaaliset polut määritellään. Toiseksi, D-TEO -menetelmä tarjoaa keinon määrittellä kriittiset rajat yksittäisille operaatioille. Tämä on hyvin tärkeää tapauksissa, joissa kvantitatiivisia käytettävyyksvaatimuksia ei ole ennalta määritelty. Kolmanneksi, D-TEO -menetelmä tarjoaa keinon erottaa kohdesovelluksesta ne osat, joiden nähdään aiheuttavan eniten käytettävyyksongelmia. Tällöin resurssit (yleensä niukat) voidaan kohdistaa vain ongelmallisimpiin kohtiin käyttöliittymää.

Väitöskirjan tekijänä kehitin ja ideoin itsenäisesti D-TEO -menetelmän, vedin läpi käyttäjäkokeet ja osallistuin aktiivisesti tutkimustulosten analysointiin. Olin myös päävastuussa tutkimusartikkelin kirjoittamisesta, joka julkaistiin *Human Technology Journal* -lehdessä.

5.5 Tutkimusartikkelien suhteet toisiinsa

Tässä alaluvussa kuvataan, miten väitöskirjaan sisällytetyt artikkelit liittyvät tutkimusalueiltaan, -aiheiltaan ja -kysymyksiltään toisiinsa (taulukko 4). Kontekstina jokaisessa artikkelissa on tilanne, jossa kehitetään innovatiivista ohjelmistotuotetta. Artikkelit A4, A5 ja A6 rajaavat kontekstin lisäksi tarkemmin kulluttajamarkkinoille suunnattuun innovatiiviseen ohjelmistotuotteeseen.

TAULUKKO 4 Tutkimukseen sisällytetyt artikkelit

Artikkeli	Tutkimusalue	Tutkimusaihe	Tutkimuskysymys
A1	Vaatusmääritys	Intentionalähtöinen vaatimusten määrittely	Q1
A2	Käyttäjäpsykologia	Käyttäjätarpeiden tunnistaminen	Q2
A3	Käyttäjäpsykologia	Käyttäjätarpeiden tunnistaminen	Q2
A4	HCI	Käytettävyyden ja käyttäjäkokeemusvaatimusten määrittely	Q3
A5	HCI	Käytettävyyden ja käyttäjäkokeemusvaatimusten määrittely	Q3
A6	HCI	Käytettävyyden arviointi	Q4

Artikkelissa A1 keskitytään vaatimusmäärittelyyn. Artikkelissa vastataan tutkimuskysymykseen Q1 selvittämällä, kuinka vaatimusmäärittely voidaan suorittaa intentionalähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta. Tuloksia havainnollistetaan Web-pohjaisiin karttapalveluihin ja laajennettuun todellisuuteen liittyvillä esimerkeillä. Artikkeleissa A2 ja A3 keskitytään käyttäjätarpeiden tunnistamiseen käyttäjäpsykologian ja ihmisen toiminnan näkökulmasta. Verrattuna artikkeliin A1, artikkelit A2 ja A3 keskittyvät tarkemmin rajattuun vaatimusmäärittelyn osa-alueeseen (tässä väitöskirjassa käyttäjätarpeiden tunnistaminen nähdään vaatimusmäärittelyn osa-alueena). Niissä vastataan tutkimuskysymykseen Q2 selvittämällä, kuinka käyttäjäpsykologiaa ja intentionalähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusmäärittelyä. Artikkeleissa A2 ja A3 ihmisen toimintaa lähestytään myös laajemmasta toiminnan näkökulmasta, pyrkien kuvaamaan intention lisäksi myös muut keskeiset toiminnan tekijät eli attribuutit.

Artikkeleissa A4 ja A5 keskitytään vuorovaikutussuunnitteluun käytettävyyden ja käyttäjäkokeemusvaatimusten näkökulmasta ja vastataan tutkimuskysymykseen Q3 selvittäen, kuinka käytettävyyden ja käyttäjäkokeemusvaatimuksia voidaan määrittää osana innovatiivisen ohjelmistotuotteen vuorovaikutussuunnittelua. Näissä artikkeleissa rajaudutaan siis käytettävyyteen liittyvien vaatimusten määrittelyyn siinä missä artikkelit A1-A3 käsittelevät vaatimusmäärittelyä ja sen osa-alueita yleisemmästä näkökulmasta. Artikkelien yhtey-

dessä käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyä pohditaan myös finanssialan järjestelmien ja turvallisuuskriittisten järjestelmien kehittämisen näkökulmasta. Kognitiivisella tehtäväanalyysillä on keskeinen rooli, ja artikkelissa pyritään vastaamaan osittain myös kognitiivisen tehtäväanalyysin kehittämisen haasteisiin.

Artikkelissa A6 keskitytään vuorovaikutussuunnitteluun käytettävyyden arvioinnin näkökulmasta. Verrattuna artikkeleihin A1-A4, se syventyy perusteellisemmin käytettävyysvaatimusten määrittelyyn yksittäisten käyttövaatimuseriä tarkkuudella. Artikkelissa vastataan tutkimuskysymykseen Q4 selvittämällä, a) kuinka käytettävyysongelmia voidaan paikantaa täsmällisesti tilanteissa, joissa kvantitatiivisia käytettävyysvaatimuksia ei ole määritelty, ja b) kuinka voidaan määrittää kriittisiä rajoja kvantitatiivisille käytettävyysvaatimuksille. Kumpaakin ongelmaa lähestytään näkökulmasta, jossa kehitetään kuluttajamarkkinoille suunnattua innovatiivista teknologiaa soveltamalla käyttäjäpsykologiaa ja hierarkkista tehtäväanalyysiä.

6 YHTEENVETO

Tässä tutkielmassa tarkastellaan käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyä ja arviointia tilanteessa, jossa kehitetään kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta. Keskeisen kirjallisuuden perusteella tällaisen ohjelmistotuotteen kehittäminen asettaa vaatimusmäärittelylle useita haasteita. Näitä haasteita ei voida ratkaista perinteisten vaatimusmäärittelyn lähestymistapojen, menetelmien ja tekniikoiden avulla. Tässä tutkielmassa haasteisiin vastataan lähestymällä vaatimusmäärittelyä intentiolähtöisestä näkökulmasta. Seuraavaksi käydään läpi tutkimuksen kontribuutio, vahvuudet ja rajoitteet sekä jatkotutkimuskohteet.

6.1 Kontribuutio

Kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen suunnittelu on vaatimusmäärittelyn näkökulmasta monimutkainen prosessi. Keskeisen kirjallisuuden perusteella tämän tyyppisen ohjelmistotuotteen kehittäminen asettaa vaatimusmäärittelylle useita haasteita. Haasteet liittyvät perinteisen käyttäjänäkökulman laajentamiseen (Tuunanen ym. 2008), käyttäjäkokemuksen huomioimiseen (Kim & Han, 2008; Kwahk & Han, 2002; Rossi & Tuunanen, 2010; Tuunanen ym. 2008) ja käyttäjätarpeiden tunnistamiseen (Casab ym. 2010; Dearden & Howard 1998; Mayhew, 1999). Lopulta haasteet kiteytyvät neljään käytettävyyss- ja käyttäjäkokemusvaatimusten määrittämisen ongelmaan: 1) kvantitatiivisten käytettävyyssvaatimusten määrittäminen, 2) käytettävyyss- ja käyttäjäkokemusvaatimusten sisällön analyysi, 3) käytettävyyss- ja käyttäjäkokemusvaatimusten yhdistäminen yhtenäiseksi prosessiksi ja 4) korkean abstraktitason käytettävyyssvaatimusten aiheuttamat ongelmat käytettävyyden arvioinnille.

Perinteiset vaatimusmäärittelyn lähestymistavat, menetelmät ja tekniikat eivät tarjoa valmiita ratkaisuja edellä esitettyihin haasteisiin ja ongelmiin. Esi-merkiksi käyttäjäkeskeinen suunnittelu ei sovellu parhaalla mahdollisella taval-

la innovatiivisen ohjelmistotuotteen suunnitteluun, koska se lähtee liikkeelle teknologiakeskeisestä suunnittelutilanteesta, jossa pyritään jo lähtökohtaisesti parantamaan olemassa olevan teknologian käytettävyyttä, tai jossa ainakin kehitettävän tuotteen käyttötarkoitus on jo ennalta päätetty (Leikas, 2009; Parkkola, 2006; Viukari, 2010). Käyttäjakeskeinen suunnittelu ei myöskään tarkastele ihmistä laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja (Leikas, 2009; Parkkola, 2006; Viukari, 2010). Tässä väitöskirjassa oletetaan, että kehitettäessä innovatiivista ohjelmistotuotetta tulisi ihmistä tarkastella laitteiden käyttäjänä hyödyntäen psykologisia käsitteitä, teorioita, säännönmukaisuuksia ja metodeja. Lisäksi käyttäjätarpeiden tunnistamisessa ja vaatimusten määrittelyssä tulisi lähteä liikkeelle mahdollisimman teknologiariippumattomasta näkökulmasta (Bergvall-Kåreborn & Ståhlbröst, 2010; Kankainen & Oulasvirta, 2003; Patnaik & Becker, 1999; Vredenburg ym. 2002). Tästä syystä väitöskirjassa valitaan käyttäjätarkastelun lähtökohdaksi käyttäjäpsykologia (Moran, 1981; Saariluoma, 2004; Oulasvirta & Saariluoma, 2004, 2006) ja vaatimusmäärittelyn lähtökohdaksi intentionalähtöisyys. Tässä väitöskirjassa tällaisesta vaatimusmäärittelyn lähestymistavasta käytetään nimitystä intentionalähtöinen lähestymistapa.

Väitöskirjan päämääränä on vastata edellä esitettyihin haasteisiin. Tutkimusongelmaa tarkastellaan neljän fokusoidumman tutkimuskysymyksen kautta: 1) Kuinka vaatimusten määrittely voidaan suorittaa intensionalähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta? 2) Kuinka käyttäjäpsykologiaa ja intensionalähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusten määrittelyä? 3) Kuinka käytettävyy- ja käyttäjäkokemusvaatimuksia voidaan määrittää osana innovatiivisen ohjelmistotuotteen vuorovaikutussuunnittelua? 4) (a) Kuinka käytettävyysongelmia voidaan paikantaa täsmällisesti tilanteissa, joissa kvantitatiivisia käytettävyysovaatimuksia ei ole määritetty? (b) Kuinka voidaan määrittää kriittisiä rajoja kvantitatiivisille käytettävyysovaatimuksille?

Etsittäessä vastauksia tutkimuskysymyksiin tutkimusstrategiana sovellettiin teoreettista, kvalitatiivista ja kvantitatiivista lähestymistapaa. Perustuen kirjallisuuteen ja empiiriseen tutkimusdataan kirjoitettiin kuusi artikkelia (A1-A6), joiden avulla vastataan neljään tutkimuskysymykseen (Q1-Q4). Tutkimustulokset lisäävät tietämystämme vaatimusmäärittelyn haasteista kehitettäessä kuluttajamarkkinoille suunnattua innovatiivista ohjelmistotuotetta ja tarjoavat konkreettisia ratkaisuja haasteista selviämiseen. Tutkimustulokset sivuavat lisäksi useita kognitiivisen tehtäväanalyysin (CTA) haasteita liittyen innovatiivisen teknologian suunnitteluun. Seuraavaksi kuvataan tutkimustuloksia kunkin tutkimuskysymyksen osalta erikseen.

Kuinka vaatimusten määrittely voidaan suorittaa intensionalähtöisesti kehitettäessä innovatiivista ohjelmistotuotetta?

Ensimmäinen tutkielman päätulos on viitekehys intensionalähtöisen vaatimusmäärittelyn tueksi (A1). Viitekehys koostuu viidestä aktiviteetistä, jotka ovat

visiointi, käyttäjäprofilointi, liiketoimintakontekstin analysointi, sosiaalisen kontekstin analysointi ja päämäärän asettaminen. Tutkimuksessa todettiin, että intention ollessa vaatimusmäärittelyn lähtökohtana voidaan vastata useisiin vaatimusmäärittelyn haasteisiin (Chung ym. 2000; Dardenne ym. 1993; Grube & Schmid, 2008; Lapouchnian, 2000; McFadzean, 1998; Yu, 1997; Yu, 1999), joita kohdataan innovatiivista ohjelmistotuotetta kehitettäessä. Esimerkkejä tällaisista vaatimusmäärittelyn haasteista ovat vastaavien tuotteiden puuttuminen, käyttäjätarpeiden tunnistamisen vaikeus sekä loppukäyttäjien käyttäytymisen ja liiketoiminnan muotoutumisen ennustamisen vaikeus.

Viitekehys on merkittävä sekä yksittäisten toimijoiden että yritysten liiketoiminnan näkökulmasta. Liiketoiminnan näkökulmasta viitekehys tukee uudentyyppisten tuotteiden kehittämistä kuluttajille, joka on nykyään yleisesti tunnettu edellytys yritysten kilpailukyvyyn säilyttämiseksi (katso esim. Atuahe-Gima, 1996; Rose, 2011). Toimijan näkökulmasta viitekehysten avulla voidaan kehittää uusia sovelluksia, joita toimijat tarvitsevat toimintansa tueksi esimerkiksi toteuttaessaan erilaisia päivittäisiä suunnitelmiaan. Tekniikkaa tulisin kehittää ensisijaisesti ihmisen toiminnan tukemiseen, ei pelkästään tekniikan itsensä kehittämisen takia (Saariluoma, 2004). Intentionlähtöinen lähestymistapa vaatimusmäärittelyyn auttaa monilta osin saavuttamaan tätä päämäärää.

Intentionlähtöinen viitekehys eroaa muista vastaavista vaatimusmäärittelyn lähestymistavoista ja viitekehyksistä pääosin kahdesta näkökulmasta. Ensimmäiseksi se ottaa vahvasti intention vaatimusmäärittelyä ohjaavaksi tekijäksi. Esimerkiksi tavoitelähtöiset menetelmät (Chung ym. 2000; Lamsweerde, 2009) eivät huomioi intentiota yhtä suoraviivaisella tavalla. Toiseksi viitekehys korostaa sitä, että vaatimusmäärittelyssä tulee keskittyä suunnittelun näkökulmasta oikeisiin asioihin (engl. focus-based). Monet muut vaatimusmäärittelyn viitekehukset (esim. Kotonya & Sommerville, 2002) ovat enemmänkin elinkaarikeskisiä (engl. lifecycle-based) korostaen vaihe-vaiheelta etenevää prosessia.

Kuinka käyttäjäpsykologiaa ja intentionlähtöisyyttä voidaan soveltaa käyttäjätarpeiden tunnistamisessa osana innovatiivisen ohjelmistotuotteen vaatimusten määrittelyä?

Tutkielman toinen päätulos on toimintakeskeinen ihminen-teknologia-vuorovaikutusontologia (A2-A3). Ontologia jäsentää vuorovaikutustilanteen kannalta keskeiset toiminnan tekijät kahteen kategoriaan. Ensimmäinen kategoria muodostuu käsitteistä, jotka liittyvät suoraan toimintaan. Näitä ovat toimija, toiminnan kohde, väline ja konteksti. Toisen kategorian käsitteet ovat edellä mainittujen attribuutteja. Intention ollessa toimijan attribuutti se on yksi ontologian määrittämistä toisen kategorian käsitteistä. Intentio nähdään myös monissa muissa tutkimuksissa yhtenä osana toiminnan kokonaisuutta (esim. Ajzen, 1985; Fishbein & Ajzen, 1975; Bratman, 1987; Gollwitzer & Bargh, 1996; Hampshire, 1965; Irwin, 1971; James, 1890; Lewin, 1951; Meiland, 1970; Miller ym. 1960; Shultz, 1980; Tolman, 1932).

Ihminen-teknologia-vuorovaikutusontologiaa voidaan soveltaa käyttäjätarpeiden tunnistamisen tukena kehitettäessä innovatiivista ohjelmistotuotetta. Tyypillisesti käyttäjätarpeiden tunnistaminen on erityisen ongelmallista tällaisessa suunnittelutilanteessa (Lindgaard ym. 2006; Macintosh, 1978). Tällaisille tilanteille ovat tyypillisiä (Anastassova ym. 2007; McKeown ym. 2004) seuraavat vaatimukset: a) Kehitettävän tuotteen tulee vastata loppukäyttäjien tarpeita. b) Loppukäyttäjien tulee hyväksyä sekä kehitettävä tuote että sen käyttötarkoitus. c) Lopputuotteen käyttötarkoituksen tulee olla loppukäyttäjien näkökulmasta mahdollisimman hyödyllinen. Käyttäjätarpeiden tunnistamisen menetelmät ja tekniikat eivät kuitenkaan pysty vastaamaan täysin edellä esitettyihin seikkoihin. Lisäksi käyttäjätarpeiden analyysin rooli osana vaatimusmäärittelyä on monilta muiltakin osin epäselvä (Lindgaard ym. 2006). Epäselvyyksiä aiheuttaa esimerkiksi se, että jotkin vaatimusmäärittelyn lähestymistavat sisältävät käyttäjätarpeiden tunnistamisen kiinteänä osana, mutta joidenkin näkökulmien mukaisesti käyttäjätarpeiden analyysin tulisi olla vaatimusmäärittelystä erillinen osa (Lindgaard ym. 2006). Tässä tutkielmassa käyttäjätarpeiden tunnistaminen nähdään kiinteänä osana vaatimusmäärittelyä siten, että se on osa vaatimusten keräämisvaihetta.

Ihminen-teknologia-vuorovaikutusontologia (A2-A3) vastaa edellä mainittuun käyttäjätarpeiden tunnistamiseen liittyvään haasteeseen soveltamalla toiminnan psykologian näkökulmaa. Toiminnan psykologian (Gollwitzer & Bargh, 1996) mukaan intentio, motivaatio ja arvot ovat keskeisessä roolissa tarkasteltaessa tarpeen käsitettä. Ihminen-teknologia-vuorovaikutusontologia määrittelee ja jäsentää ihmisen toiminnan keskeiset tekijät (ts. attribuutit), jotta pystyisimme kehittämään innovatiivista teknologiaa teknologiariippumattomasta näkökulmasta tavalla, joka tukee ensisijaisesti ihmisen toimintaa. Ihminen-teknologia-vuorovaikutusontologian lähtökohdaksi on otettu käyttäjäpsykologia ja intentiolähtöinen ajattelu.

Ihminen-teknologia-vuorovaikutusontologia eroaa muista ihmisen toimintaa jäsentävistä ja kuvaavista malleista ja teorioista, kuten toiminnan teoriasta, määrittelemällä intension selvästi itsenäiseksi osaksi ihmisen toimintaa ja jakamalla sen edelleen mitä-intentioksi ja kuinka-intentioksi. Intention jakaminen mitä-intention ja kuinka-intention määrittelee eron aidosti ihmislähtöisen (engl. human-driven) ja teknologianlähtöisen (engl. technology-driven) lähestymistavan välille. Tämä auttaa jäsentämään monissa tutkimuksissa (esim. Leikas, 2009; Parkkola, 2006; Viukari, 2010) esille nostettua ongelmaa aidosti ihmislähtöisen ja teknologialähtöisen vuorovaikutussuunnittelun eroista. Verrattuna toiminnan teoriaan ihminen-teknologia-vuorovaikutusontologialla on myös enemmän ns. ennustavan teorian piirteitä (katso lisää HCI-teorioiden luokittelusta esim. Shneiderman, 2002). Ihminen-teknologia-vuorovaikutusontologia pyrkii esimerkiksi tietyissä määrin ennustamaan ainakin karkealla tasolla ihmisen toimintaa uudentyyppisissä vuorovaikutustilanteissa, sillä ontologian avulla suunnittelija voi hahmotella uusien käyttötilanteiden vaikutusta ihmiseen teknologian käyttäjänä.

Kuinka käytettävyys- ja käyttäjäkokemusvaatimuksia voidaan määrittää osana kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vuorovaikutussuunnittelua?

Tutkielman kolmas päätulos on Interaction Deconstruction -menetelmä käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyyn (A4-A5). Menetelmä koostuu kolmesta vaiheesta: keskeisten toimintojen määrittely, hierarkkisen puurakenteen muodostaminen keskeisistä toiminnoista ja käytettävyysattribuuttien asettaminen hierarkkisen puun solmuille.

Menetelmän kehittämisen motivaationa on ollut käytettävyysvaatimuksia koskevasta kirjallisuudesta löydetty puutteet liittyen esimerkiksi käyttäjäkokemusvaatimusten määrittämiseen ja psykologisen tietämyksen sisällyttämiseen osaksi vaatimusmäärittelyprosessia kehitettäessä innovatiivista ohjelmistotuotetta. Interaction Deconstruction -menetelmässä hyödynnetään ja sovelletaan olemassa olevaa tutkimusta liittyen hierarkkisen tehtäväanalyysin, kognitiivisen tehtäväanalyysin ja käytettävyyden sekä käyttäjäkokemuksen attribuuttien analyysin suhteen. Tällä tavoin toimimalla voidaan vastata moniin käytettävyyteen liittyviin vaatimusmäärittelyn haasteisiin (esim. Jokela ym. 2005; Kantola & Jokela 2007).

Artikkelissa A1 vaatimusmäärittelyä käsiteltiin intension näkökulmasta ja artikkeleissa A2 ja A3 tuotiin esille toiminnan kokonaisuuden merkitys osana käyttäjätarpeiden tunnistamista. Interaction Deconstruction -menetelmä hyödyntää näitä tutkimuksia pyrkien käsittelemään yksittäisiä tehtäviä alitehtävien tarkkuudella ja tavoitteita korkean tason tavoitteiden ja matalan tason tavoitteiden mukaisesti. Toimimalla näin päästään määrittämään käytettävyys- ja käyttäjäkokemusvaatimuksia, jotka ovat riittävän täsmällisiä toimimaan edelleen myös käytettävyydestauksen tukena.

Interaction Deconstruction -menetelmän keskeisin vaihe on käytettävyysattribuuttien asettaminen hierarkkisen puun solmuille. Tämä vaihe on merkittävä, koska se mahdollistaa käytettävyys- ja käyttäjäkokemusvaatimusten määrittämisen rinnakkain yksittäisten alitehtävien tarkkuudella. Interaction Deconstruction -menetelmä eroaa muista vastaavista menetelmistä (esim. Gould & Lewis, 1985; Mayhew, 1999; Nielsen, 1993; Wixon & Wilson, 1997) edukseen nimenomaan mahdollistamalla käytettävyys- ja käyttäjäkokemusvaatimusten rinnakkaisen määrittelyn. Myös Preece (2002) on yhdistänyt käytettävyys- ja käyttäjäkokemusvaatimusten määrittelyn, mutta Interaction Deconstruction -menetelmä tarjoaa siihen verrattuna selkeämmän vaihe vaiheelta etenevän prosessin käytettävyyttä koskevien vaatimusten määrittelyyn. Lisäksi Interaction Deconstruction -menetelmä eroaa Preece (2002) menetelmästä mahdollistamalla käytettävyys- ja käyttäjäkokemusvaatimusten sisällön analyysin psykologian näkökulmasta.

Käytännössä käytettävyys- ja käyttäjäkokemusvaatimusten määrä kasvaa helposti suureksi. Kokonaisuuden hallitsemiseksi kannattaa hyödyntää ei-toiminnallisten vaatimusten määrittelyssä käytettyä yleistä merkintätapaa (Chung ym. 2000). Interaction Deconstruction -menetelmä ei vielä sisällä tätä vaihetta, mutta jatkokehityksen kannalta se on välttämätöntä.

(a) Kuinka käytettävyyso ongelmia voidaan paikantaa täsmällisesti tilanteissa, joissa kvantitatiivisia käytettävyyso vaatimuksia ei ole määritelty? (b) Kuinka voidaan määrittää kriittisiä rajoja kvantitatiivisille käytettävyyso vaatimuksille?

Tämän tutkimuksen neljäs päätulos on D-TEO -menetelmä käytettävyyso ongelmien paikantamisen ja käytettävyyttä koskevien vaatimusten kriittisten rajojen määrittämisen tueksi (A6). D-TEO -menetelmä soveltuu erityisen hyvin käytettäväksi kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen käytettävyyttä koskevien vaatimusten määrittelyn tueksi. Menetelmää voidaan käyttää myös itsenäisenä menetelmänä käytettävyyden arviointiin tilanteissa, joissa kvantitatiivisia käytettävyyso vaatimuksia ei ole määritelty. Menetelmä koostuu neljästä vaiheesta: a) testin päämäärien ja käyttäjäprofiilien määrittäminen, b) testitehtävien suunnittelu ja optimaalisten polkujen määrittäminen, c) testien järjestäminen ja datan kerääminen ja d) datan analysointi ja johtopäätösten tekeminen.

D-TEO -menetelmä ratkaisee kirjallisuudessa esitettyjä käytettävyyso vaatimuksiin liittyviä ongelmia (esim. Jokela ym. 2006) sekä käytettävyyso vaatimusten määrittelyn (Kantola & Jokela, 2007) että käytettävyyso testauksen (Ivory & Hearst, 2001) näkökulmasta. Vaatimusmäärittelyn näkökulmasta D-TEO -menetelmää voidaan käyttää apuna, kun halutaan asettaa kriittiset rajat kvantitatiivisille käytettävyyso vaatimuksille käyttöliittymän prototyypin avulla. Käytettävyyso testauksen näkökulmasta D-TEO -menetelmä tarjoaa tukea tilanteissa, joissa käytettävyyso ongelmat halutaan paikantaa tarkasti käyttöliittymältä, vaikka tarkkoja käytettävyyso vaatimuksia ei ole määritelty.

D-TEO -menetelmä eroaa pääosin kolmella tavalla muista käytettävyyden arviointimenetelmistä. Ensimmäiseksi D-TEO -menetelmä on kevyt menetelmä verrattuna muihin vastaaviin menetelmiin, jotka pyrkivät paikantamaan käytettävyyso ongelmia yksittäisten operaatioiden tarkkuudella (esim. Paganelli & Paternò, 2002; Paternò & Ballardín, 1999; Paternò & Ballardín, 2000). Toiseksi D-TEO -menetelmä tarjoaa keinon määrittellä kriittiset rajat yksittäisten operaatioiden tarkkuudella. Kriittisten rajojen määrittely perustuen tilastollisiin menetelmiin tuo lisätietoa erityisesti kvantitatiivisten käytettävyyso vaatimusten määrittelyyn (Jokela ym. 2006; Kantola & Jokela, 2007) ja auttaa myös paikantamaan käytettävyyso ongelmia tilanteissa, joissa määrällisiä käytettävyyso vaatimuksia ei ole ennalta määritelty. Kolmanneksi D-TEO -menetelmän avulla on mahdollista paikantaa tarkasti arvioitavan käyttöliittymän ongelmallisimmat osat ja kohdistaa suunnitteluresurssit näihin ”pullonkauloihin”.

D-TEO -menetelmän merkittävyys nousee esille juuri edellä mainittujen eroavaisuuksien valossa. Kohtalaisen kevyenä menetelmänä sen soveltaminen käytännön ohjelmistokehitystyössä on lupaavaa. D-TEO -menetelmän nykyisen version integroiminen käytännön ohjelmistokehitystyöhön vaatii kuitenkin jatkokehitystä. Lisäksi menetelmää tulisi myös laajentaa kattamaan paremmin käyttäjäkokemuksen näkökulma (nykyisin menetelmä perustuu lähinnä käytettävyyden tehokkuus-näkökulmaan). Kriittisten rajojen määrittäminen yksittäisten operaatioiden tarkkuudella on merkittävää teoreettisesta ja käytännön nä-

kökulmasta. Teoreettisesta näkökulmasta se tuo lisätietoa käytettävyyksvaatimusten määrittämisen teoriaan. Käytännön näkökulmasta se mahdollistaa tarkan käytettävyyden arvioinnin prototyyppivaiheessa olevalle käyttöliittymälle, jolle ei ole määritelty aiemmin kvantitatiivisia käytettävyyksvaatimuksia. Myös käyttöliittymän ”pullonkaulojen” paikantaminen tekee D-TEO -menetelmästä merkittävän. Tällöin käytettävyyteen varatut resurssit voidaan kohdistaa paremmin kaikista ongelmallisimpiin käyttöliittymän osiin.

6.2 Tutkimuksen vahvuudet ja rajoitteet

Ihmisen ja tietokoneen välisen vuorovaikutustutkimuksen näkökulmasta tieteilisellä teoriolla voi olla viisi erilaista roolia (Shneiderman, 2002): deskriptiivinen, selittävä, ennustava, preskriptiivinen ja generatiivinen. Deskriptiivinen teoria määrittelee keskeiset käsitteet ja muuttujat sekä käsitteiden väliset erot. Selittävä teoria selittää ilmiöiden suhteet ja erilaiset prosessit. Ennustava teoria mahdollistaa tapahtumien ennustamisen määritellyissä konteksteissa. Preskriptiivinen teoria tarjoaa parhaiksi koettuihin käytäntöihin perustuvia ohjeistuksia. Generatiivinen teoria tukee luovuutta ja uuden keksimistä sekä löytämistä. (Shneiderman, 2002).

Tässä väitöskirjatutkimuksessa esitetty intentiolähtöinen lähestymistapa vaatimusten määrittelyyn on teorianäkökulmasta deskriptiivinen, selittävä, preskriptiivinen ja generatiivinen. Lähestymistapa on deskriptiivinen, koska siinä määritellään ja jäsennetään ihmisen toiminnan vuorovaikutustilanteeseen liittyvät keskeiset käsitteet (esim. intentio, väline ja konteksti). Se on myös selittävä, sillä se ehdottaa tiettyjen ilmiöiden olemassaoloa, suhdetta muihin ilmiöihin ja selittää niiden toimintaa (esimerkiksi mitä-intentio ja kuinka-intentio). Lähestymistapa on preskriptiivinen, koska se hyödyntää hyviksi koettuja käytäntöjä osana vaatimusmäärittelyä ja tarjoaa konkreettisia vaihe-vaiheelta eteneviä ohjeita käytettävyyttä koskevien vaatimusten määrittelyyn ja arviointiin (esim. Interaction Deconstruction -menetelmä ja D-TEO -menetelmä). Se on myös generatiivinen, sillä sen avulla pyritään kehittämään uusia ja innovatiivisia ohjelmistotuotteita lähestymällä vaatimusmäärittelyä intentiolähtöisestä näkökulmasta.

Edellä esitetyn perusteella voidaan sanoa, että intentiolähtöinen lähestymistapa vaatimusten määrittelyyn on teorianäkökulmiltaan varsin monipuolinen. Siinä yhdistetään syvällinen teoreettinen tutkimus käytännönläheisten menetelmien ja tekniikoiden kehittämiseen ja edelleen niiden konkreettiseen soveltamiseen käytännössä. Koska lähestymistapa on uusi, työ painottuu enemmän teoreettiseen tarkasteluun kuin empiiriseen näkökulmaan. Tutkimus on ollut vaativa monestakin syystä. Ensimmäinen syy oli tutkimusaiheen moninaisuus. Tutkimus sijoittuu vaatimusmäärittelyyn, ihminen-teknologia-vuorovaikutustutkimukseen (HCI), innovatiivisen ohjelmistotuotteen kehittämisen ja käyttäjäpsykologian leikkauspintaan ja siinä sovelletaan kognitiotieteen ja tietojärjestelmätieteen menetelmiä, lähestymistapoja, käsitteitä, teorioita ja

tekniikoita. Lisäksi tutkimusaihe edellyttää erityistietämystä hierarkkisesta tehtäväänalyysistä (HTA), kognitiivisesta tehtäväänalyysistä (CTA), toiminnan teoriasta ja ihmisen toiminnan psykologiasta. Toinen syy tutkimuksen vaativuuteen oli alkuvaiheen tutkimusidean kypsyysaste. Tutkimusta aloitettaessa tutkimusongelma ja -kysymykset eivät olleet täysin selvillä, joten määrittelin nämä tutkijana käymällä läpi laajasti monialaista kirjallisuutta laatien samalla tutkimussuunnitelman. Johtuen osittain siitä, että lähestymistapa on uusi, tutkimuksen rajauksen mukaisia valmiita kirjallisuuskatsauksia ja muita vastaavia tieteellisiä selvitystöitä ei ollut käytettävissä. Tästäkin syystä tutkimus painottui teoreettiseen lähestymistapaan.

Tutkimusongelmaan ja tutkimuskysymyksiin vastaaminen edellytti teoreettisen lähestymistavan lisäksi myös kvalitatiivisen ja kvantitatiivisen lähestymistavan soveltamista. Kvalitatiivisen lähestymistavan näkökulmasta sovelsimme aineiston analyysimenetelmänä kvalitatiivista sisällön analyysiä (Met-sämuuronen, 2008; Syrjäläinen, 1994) jäsentäessämme vuorovaikutusontologiaan kuuluvia ihmisen toiminnan tekijöitä. Kvantitatiivisen lähestymistavan näkökulmasta sovelsimme todennäköisyyslaskentaan ja tilastollisiin menetelmiin perustuvaa analyysiä paikantaessamme käyttöliittymän ongelmakohtia ja määrittellessämme kvantitatiivisten käytettävyyksivaatimusten raja-arvoja.

Väitöskirjatutkimuksessa esiintyy seuraavia rajoitteita. Ensinnäkin osa kehitetyistä menetelmistä ja tekniikoista (erityisesti A1, A4 ja A5) perustuu vain kirjallisuuteen eikä niiden toimivuutta käytännössä ole empiirisiin kokein todistettu. Teoreettisen lähestymistavan taustalla oleva aineisto olisi voinut olla kattavampi ja perustua järjestelmällisempään kirjallisuuskatsaukseen. Teoreettisen lähestymistavan taustalla oleva aineiston laajuus olisi voinut olla parempi myös kehitettäessä ihminen-teknologia-vuorovaikutusontologiaa (A2, A3) ja D-TEO -menetelmää (A6). Vuorovaikutusontologian tapauksessa olisimme voineet esimerkiksi käydä systemaattisemmin läpi olemassa olevia ihmisen toimintaa kuvaavia malleja ja ontologioita. D-TEO -menetelmän tapauksessa olisimme voineet käydä läpi laajemmin ja kattavammin vastaavia olemassa olevia menetelmiä. Toisaalta sekä vuorovaikutusontologian että D-TEO -menetelmän tapauksessa käytettiin teoreettisen lähestymistavan tukena myös empiiristä tutkimusta.

Toiseksi, tässä tutkimuksessa ei ole perusteellisesti huomioitu kehitettyjen menetelmien ja tekniikoiden integroimista olemassa oleviin ohjelmistokehityksen malleihin. Esimerkiksi intentiolähtöinen viitekehys ja D-TEO -menetelmä sopisivat hyvin ketterän ohjelmistokehityksen tueksi. Myöskään sitä ei ole määritelty, kuinka ihminen-teknologia-vuorovaikutusontologia voitaisiin integroida osaksi nykyisiä käyttäjätarpeiden tunnistamisen menetelmiä ja tekniikoita. Edelleen Interaction Deconstruction -menetelmän yhdistäminen osaksi muita vaatimusmäärittelyn vaiheita (esim. vaatimusten dokumentointi) voitaisiin määrittellä nykyistä perusteellisemmin, esimerkiksi vaihe-vaiheelta etenevänä prosessina.

Kolmanneksi, emme pystyneet vertaamaan tässä tutkimuksessa esitettyä intentiolähtöistä vaatimusmäärittelyn lähestymistapaa täysin kattavasti muihin vastaaviin lähestymistapoihin. Keskeisin syy tähän oli se, että juuri vastaavaan

käyttötarkoitukseen, eli kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen käytettävyy- ja käyttäjäkokemusvaatimusten määrittelyyn tarkoitettua ”vastaavaa lähestymistapaa” ei vaikuttanut olevan olemassa. Olisimme kuitenkin voineet käydä nykyistä perusteellisemmin läpi esimerkiksi niitä käytäntöjä, jotka on todettu toimiviksi määriteltäessä vaatimuksia erikseen joko kuluttajamarkkinoille suunnatuille tai innovatiivisille ohjelmistotuotteille. Käytettävyyksivaatimusten näkökulmasta ongelmaiseksi muodostui se, että samoja (ns. yleisen tason) vaatimusmäärittelyn lähestymistapoja, menetelmiä ja tekniikoita tunnuttiin käytettävän useissa eri konteksteissa riippumatta siitä, onko kyseessä esimerkiksi innovatiivinen tai kuluttajamarkkinoille suunnattu ohjelmistotuote. Käyttäjäkokemusvaatimusten määrittämisen näkökulmasta vaikeuksia menetelmien ja tekniikoiden väliseen vertailuun aiheuttaa se, että selkeästi tähän tarkoitukseen suunnattuja hyvin määriteltyjä menetelmiä ja tekniikoita ei vaikuta olevan olemassa.

6.3 Jatkotutkimuskohteet

Monk (1998) on todennut, että HCI-näkökulmasta intention huomioivaa tutkimusta tarvitaan lisää, jotta ihmisen käyttäytymistä voitaisiin ymmärtää paremmin vuorovaikutustilanteessa. Tässä väitöskirjatutkimuksessa on pyritty vastaamaan tähän haasteeseen sekä teorian että käytännön näkökulmasta kehittämällä intentiolähtöinen lähestymistapa käytettävyy- ja käyttäjäkokemusvaatimusten määrittämisen tueksi. Samalla tutkimus nosti esille jatkotutkimuskohteita edellä esitetyistä tutkimuksen rajoitteista että täysin uusista ideoista.

Tutkimuksen rajoitteista esille nousseet jatkotutkimuskohteet ovat: 1) perusteelliset empiiriset tutkimukset liittyen väitöskirjatutkimuksessa kehitettyjen menetelmien ja tekniikoiden soveltuvuuteen käytännön ohjelmistokehitykseen, 2) kehitettyjen menetelmien ja tekniikoiden integrointi olemassa oleviin ohjelmistokehityksen malleihin ja lähestymistapoihin ja 3) intentiolähtöisen lähestymistavan perusteellinen vertaaminen sitä lähinnä vastaaviin, olemassa oleviin lähestymistapoihin. Lisäksi olisi hyödyllistä tutkia systemaattisen kirjallisuuskatsauksen keinoin, mitä kuluttajamarkkinoille suunnatulla innovatiivisella ohjelmistotuotteella itse asiassa tarkoitetaan ja mitä vaatimusmäärittely tarkoittaa tässä yhteydessä nimenomaan käytettävyyden ja käyttäjäkokemuksen näkökulmasta. Tässä tutkimuksessa nämä asiat käytiin toki läpi, mutta syvällisemmän vastauksen saaminen edellyttäisi vieläkin laajempaa tutkimusta. Edelleen yksi tutkimuksen rajoitteista esille nousseista yksittäisistä jatkotutkimuskohteista on sekä teoreettinen että empiirinen tutkimus liittyen käytettävyy- ja käyttäjäkokemusvaatimusten dokumentointiin sopivalla tarkkuustasolla. Riippumatta toimialasta olisi hyvä vertailla erilaisia menetelmiä ja tekniikoita ei-toiminnallisten vaatimusten dokumentointiin käytettävyy- ja käyttäjäkokemusvaatimusten näkökulmasta.

Väitöskirjan työstämisen aikana syntyi myös useita uusia ideoita jatkotutkimuskohteiksi. Yksi mielenkiintoinen jatkotutkimuskohde olisi Interaction De-

construction -menetelmän ja D-TEO -menetelmän jatkokehittäminen turvallisuuskriittisten järjestelmien kehittämisen tarpeisiin. Molemmat menetelmät perustuvat yksittäisten tehtävien alitehtävien ja operaatioiden tarkkuuteen, joten ne mahdollistavat sekä vuorovaikutuksen suunnittelun että testauksen hyvin tarkalla toiminnan tasolla. Molemmat menetelmät tukevat lisäksi kognitiivista tehtäväänalyysiä, joka on hyväksi koettu lähestymistapa turvallisuuskriittisten järjestelmien tutkimuksessa. Myös ihminen-teknologia-vuorovaikutusontologian sopivuutta turvallisuuskriittisten järjestelmien suunnitteluun ja erilaisten ongelmien analyysiin tulisi tutkia lisää. Vuorovaikutusontologian avulla voitaisiin esimerkiksi syventää tietämystä ihminen-teknologia vuorovaikutuksesta osana tapahtumien tutkintaa turvallisuuskriittisissä organisaatioissa. Edelleen, intentioiden tutkimusta tulisi jatkotutkimusten yhteydessä syventää muun muassa mentaalisia malleja tutkivien psykologisten teorioiden ja viitekehysten kautta.

SUMMARY

This dissertation focused on usability and user experience requirements engineering from the viewpoint of human-technology interaction design, particularly in the context of designing innovative consumer information systems.

Firstly, the literature review was conducted to find out the challenges of requirements engineering in designing innovative software products. In order to elicit and develop better solutions to the challenges a framework for intention-driven requirements engineering of innovative software products was developed. This framework is strongly based on the earlier research on innovation and creativity as well as goal-oriented requirements engineering approaches. However, compared to the existing approaches our framework derives from the concept of intention as a starting point of requirements engineering. This provides a deeper understanding of how the desired software product would meet social, technological and business needs and why the product is actually needed. The framework guides the elaboration of high-level goals into low-level goals and further into concrete requirements.

Secondly, a light-weight ontology was constructed for understanding and describing human interaction behaviour in human-technology interaction in the context of innovative software product design. The main goal of this ontology is to clarify both the user needs analysis and the early phases of requirements engineering by increasing the understanding of human-technology interaction among technology designers. The human-technology interaction ontology divides the concepts related to human action into two categories. The first category contains the concepts that are directly related to action. These are human agent, object, instrument, and context. The concepts in the second category are considered attributes of the concepts of the first category. The human-technology interaction ontology was constructed based on the literature review and the content analysis within ten enterprises participating in the Easy Interactions project.

Thirdly, the Interaction Deconstruction method was devised to support engineering of usability and user experience requirements. This method favourably differs from the existing usability requirements engineering methods in many ways. Firstly, it supports defining both usability and user experience requirements. Secondly, it utilizes psychological knowledge to engineer scientifically justifiable requirements. Thirdly, the Interaction Deconstruction method supports defining the relevant user-related attributes at the level of sub-actions. The method can be used in designing both a radically new technology and new versions of existing technology. This method utilizes task analysis, hierarchical tasks analysis and psychological analysis of usability-related attributes to yield the usability and user experience requirements.

Fourthly, a novel quantitative usability evaluation method, the D-TEO method, was introduced and tested. This method was developed to fulfil the needs revealed by the review of the relevant usability engineering literature, as well as to meet the challenges of the practical usability evaluation methods and

techniques. The D-TEO method has several advantages compared to the corresponding usability evaluation methods. First, the method is rather lightweight. Second, the D-TEO method supports decision making on the limits of critical values. Thirdly, the D-TEO method distinguishes those part of user interface that are more problematic. The D-TEO method can be used in both existing and innovative technology design contexts, but it is especially suitable when usability-related requirements have not been defined in the quantitative way during innovative technology design. The D-TEO method was tested in a real life evaluation study with 10 participants.

To summarize, this dissertation study showed that the intention-driven approach provides a new viewpoint for usability-related requirements engineering of the consumer information systems. It also provides a new perspective for requirements engineering in general. Through considering the concept of intention as the core of the requirements engineering process requirements engineering is integrated with psychological understanding of human action. In order to give answers to the general research problem and the specific research questions the following methods and techniques mentioned were developed: a) the framework for intention-driven requirements engineering, b) the lightweight human-technology interaction ontology, c) the Interaction Deconstruction method for usability-related requirements engineering, and d) the D-TEO method for obtaining deeper understanding of the limits of critical values concerning quantitative usability requirements. In addition to these practical artefacts, this dissertation offers an extensive literature review of the challenges and solutions in interaction design and requirements engineering.

YHTEENVETO

Tässä tutkielmassa keskityttiin käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittelyyn ihminen-teknologia vuorovaikutussuunnittelun näkökulmasta. Tutkielmassa rajauduttiin erityisesti suunnittelutilanteisiin, joissa kehitetään kuluttajamarkkinoille suunnattuja innovatiivisia ohjelmistotuotteita. Ensimmäiseksi tunnistettiin kirjallisuuteen perustuen tyypilliset vaatimusmäärittelyn kohtaamat haasteet kehitettäessä innovatiivista ohjelmistotuotetta. Kirjallisuuteen perustuen selvitettiin, kuinka näihin haasteisiin on pyritty vastaamaan ja tuotiin esille myös ratkaisemattomat haasteet. Osaan ratkaisemattomista haasteista pyrittiin vastaamaan rakentamalla viitekehysten intensionalisen vaatimusmäärittelyn tueksi. Viitekehys perustuu vahvasti tavoittepohjaisten menetelmien ja tekniikoiden hyödyntämiseen osana vaatimusmäärittelyä korostaen kuitenkin vahvemmin intension merkitystä vaatimusten määrittelyn lähtökohdana. Ottamalla intension vaatimusten määrittelyn lähtökohdaksi, pyritään syvällisemmin ymmärtämään, miksi tiettyä ohjelmistotuotetta ollaan itse asiassa kehittämässä huomioiden loppukäyttäjän, liiketoiminnan ja teknologian näkökulmat. Viitekehysten mukaan intensionalisten jaloitetaan ensin korkean tason päämääriä, joita tarkennetaan edelleen matalan tason päämääriksi kunnes päästään ohjelmistotuotteen käyttöä koskevien vaatimusten tarkkuustasolle.

Toiseksi rakennettiin kevyt ontologia kuvaamaan ihmisen toimintaa ihmisen-teknologia vuorovaikutustilanteessa. Ontologian tarkoituksena on selkeyttää innovatiivisen ohjelmistotuotteen suunnittelua erityisesti käyttäjätarpeiden tunnistamisen ja vaatimusmäärittelyn alkuvaiheiden suhteen lisäämällä suunnittelijoiden ymmärrystä siitä, kuinka aiottu teknologia vaikuttaa ihmisen toimintaan ihminen-teknologia vuorovaikutustilanteessa. Ontologian mukainen ihmisen toiminnan analyysi perustuu neljän keskeisen käsitteen (toimija, toiminnan kohde, väline, konteksti) ja niiden attribuuttien analysointiin psykologista tietämystä hyödyntäen. Ontologian rakentaminen perustui sekä olemassa olevaan kirjallisuuteen että sisältöanalyysiin, joka suoritettiin osana ITEA2-hankkeen Easy Interactions projektia.

Kolmanneksi kehitettiin Interaction Deconstruction -menetelmä käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittämisen tueksi. Menetelmän avulla pyritään hyödyntämään aiempaa perusteellisemmin psykologista tietämystä käytettävyydestä ja käyttäjäkokemuksesta vaatimusten määrittelyn alkuvaiheissa. Menetelmän kehittämisen motivaationa olivat tämän lisäksi kirjallisuuden perusteella havaitut puutteet hyödyntää tietämystä käyttäjäkokemuksesta osana vaatimusmäärittelyä ja tarkastella käytettävyyttä ja käyttäjäkokemusta koskevia vaatimuksia matalan abstraktiotason näkökulmasta. Interaction Deconstruction -menetelmää voidaan käyttää kehitettäessä sekä innovatiivisia ohjelmistotuotteita että uusia versioita olemassa olevista sovelluksista. Menetelmän kehittämisessä on hyödynnetty erityisesti käytettävyyden ja käyttäjäkokemuksen attribuutteihin liittyvää teoreettista tietämystä sekä tehtäväanalyysiin liittyvää kirjallisuutta.

Neljänneksi tunnistettiin käytettävyyttä koskevasta kirjallisuudesta puute, jonka mukaan nykyisillä käytettävyyden arviointimenetelmillä on vaikea paikantaa täsmällisesti käytettävyyssongelmia arvioitaessa innovatiivisen ohjelmistotuotteen käyttöliittymän käytettävyyttä. D-TEO -menetelmä rakennettiin käytettävyyssongelmien paikantamisen ja käytettävyyttä koskevien vaatimusten kriittisten rajojen määrittämiseksi tueksi. Menetelmän avulla voidaan erotella kehitettävästä käyttöliittymästä käytettävyyden näkökulmasta ongelmallimmat osat ja keskittyä niiden parantamiseen. D-TEO -menetelmässä käytetään todennäköisyyslaskentaan perustuvaa menettelytapaa käytettävyyssongelmien paikantamiseen erityisesti tapauksessa, jossa tarkastellaan käytettävyyttä eksplisiittisten tiedonhakutehtävien ja tehokkuuden (aika, virheet) näkökulmasta. D-TEO -menetelmää voidaan käyttää sekä olemassa olevan tuotteen käyttöliittymän käytettävyyden arviointiin että innovatiivisen ohjelmistotuotteen käyttöliittymän prototyypin suunnittelun tukena. Menetelmä on erityisen hyödyllinen tilanteissa, joissa kehitetään täysin uudentyyppistä sovellusta eikä matalan abstraktiotason kvantitatiivisia käytettävyyssvaatimuksia voida johtaa esimerkiksi sovelluksen edellisestä versiosta tai kilpailijatuotteista. D-TEO -menetelmän toimivuutta arvioitiin käytännössä käytettävyyssitestauksen avulla.

Yhteenvetona voidaan sanoa, että tässä tutkielmassa kartoitettiin keskeiset haasteet ja ongelmat, joita kohdataan määriteltäessä käytettävyyttä ja käyttäjäkokemusta koskevia vaatimuksia osana kuluttajamarkkinoille suunnatun innovatiivisen ohjelmistotuotteen vuorovaikutussuunnittelua. Vastaukseksi haasteisiin kehitettiin a) viitekehys intentiolähtöisen vaatimusmäärittelyn alkuvaiheiden tueksi, b) ihminen-teknologia- vuorovaikutusontologia, jonka tarkoituksena on selkeyttää innovatiivisen ohjelmistotuotteen suunnittelua käyttäjätarpeiden tunnistamisen suhteen, c) Interaction Deconstruction -menetelmä käytettävyyttä ja käyttäjäkokemusta koskevien vaatimusten määrittämisen tueksi sekä d) D-TEO -menetelmä käytettävyyssongelmien paikantamisen ja käytettävyyttä koskevien vaatimusten kriittisten rajojen määrittämiseksi tueksi. Näiden tulosten lisäksi tutkielma tarjoaa laajan kirjallisuuskatsauksen ihminen-teknologia-vuorovaikutuksen ja vaatimusmäärittelyn haasteista ja ratkaisuksista.

LÄHTEET

- Aaen, I. 2008. Essence: facilitating software innovation. *European Journal of Information Systems* 17 (5), 543–553.
- Abrahamsson, P., Warsta, J., Siponen, M.T. & Ronkainen, J. 2003. New directions on agile methods: a comparative analysis. *Teoksessa Proceedings of the 25th International Conference on Software Engineering, Portland: IEEE, 244–254.*
- Abran, A., Khelifi, A., Suryan, W. & Seffah, A. 2003. Usability meanings and interpretations in ISO Standards. *Software Quality Journal* 11 (4), 325–338.
- Adelson, B. 1981. Problem-solving and the development of abstract categories in Programming languages. *Memory and Cognition* 9 (4), 422–433.
- Agarwal, R. & Karahanna, E. 2000. Time flies when you're having fun: cognitive absorption and beliefs about information technology usage. *MIS Quarterly* 24 (4), 665–694.
- Ahmad, A. 2011. Social network sites and its popularity. *International Journal of Research and Reviews in Computer Science* 2 (2), 522–526.
- Ajzen, I. 1985. From intentions to actions: a theory of planned behavior. *Teoksessa J. Kuhl & J. Beckmann (toim.) Action-control: From cognition to behavior. Heidelberg: Springer, 11–39.*
- Ajzen, I. 1991. The theory of planned behavior. *Organizational Behavior and Human Decision Processes* 50 (2), 179–211.
- Alkula, T., Pöntinen, S. & Ylöstalo, P. 1994. *Sosiaalitutkimuksen kvantitatiiviset menetelmät. Juva: WSOY.*
- Alves, C., Ramalho, G. & Damasceno, A. 2007. Challenges in requirements engineering for mobile games development: the meantime case study. *Teoksessa Proceedings of the 15th IEEE International Conference on Requirements Engineering 2007, Los Alamitos, California: IEEE Computer Society, 275–280.*
- Anastassova, M., Mégard, C. & Burkhardt, J.M. 2007. Prototype evaluation and user-needs analysis in the early design of emerging technologies. *Teoksessa Proceedings of the 12th International Conference on Human-Computer Interaction, Beijing, China 22–27 July 2007.*
- Anderson, J. R., Farrell, R. & Sauers, R. 1984. Learning to program LISP. *Cognitive Science* 8, 87–129.
- Anderson, J. R. & Jeffries, R. 1985. Novice LISP errors: undetected losses of information from working memory. *Human-Computer Interaction* 1, 107–131.
- Annett, J. 2000. Theoretical and pragmatic influences on task analysis methods. *Teoksessa J. M. Schraagen, S. F. Chipman, & V. L. Shalin (toim.) Cognitive Task Analysis. Mahwah, NJ: Erlbaum, 27–37.*
- Annett, J. & Duncan, K. D. 1967. Task analysis and training design, *Occupational Psychology* 41, 211 - 221.
- Annett, J., Duncan, K. D., Stammers, R. B. & Gray, M. J. 1971. *Task Analysis. London: HMSO.*

- Arhippainen, L. 2009. Studying user experience: issues and problems of mobile services - case ADAMOS: user experience (im)possible to catch? Oulun yliopisto. Oulu: Oulun yliopisto. Väitöskirja.
- Aristoteles 2012. Nikomakhoksen etiikka. (Teokset VII) Suomentaja ja selitysten laatija Simo Knuuttila. Helsinki: Gaudeamus (Alkuteos julk. 1094).
- Atuahene-Gima, K. 1996. Market orientation and innovation. *Journal of Business Research* 35, 93-103.
- Baddeley, A. 1997. *Human memory: theory and practice*. Hove: Psychology Press.
- Baddeley, A. 2002. Is working memory working? *European Psychologist* 7 (2), 85-97.
- Baddeley, A. 2007. *Working memory, thought and action*. Oxford: Oxford University Press.
- Bandura, A. 1977. Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review* 84 (2), 191-215.
- Baregheh, A., Rowley, J. & Sambrook, S. 2009. Towards a multidisciplinary definition of innovation. *Management Decision* 47 (8), 1323-1339.
- Bastek, M., Tuunanen, T., & Gardner, L. 2008. *Consumer information systems service families*. Auckland: The University of Auckland Business School.
- Beck, K. 1999. Embracing change with extreme programming. *IEEE Computer* 32 (10), 70-77.
- Bentley, T., Johnston, L. & von Baggo, K. 2002. Putting some emotion into requirements engineering. Teoksessa *The Seventh Australian Workshop on Requirements Engineering*. Melbourne, Australia: Deakin University, 227-244.
- Bergvall-Kåreborn B, Ståhlbröst A. 2010. User expressions translated into requirements. *Human Technology* 6 (2), 212 - 229.
- Bernard, M. 2002. Examining a metric for predicting the accessibility of information within hypertext structures. Doctoral dissertation, Wichita, KS, USA: Wichita State University.
- Bertelsen, O.W. & Bødker, S. 2003. Activity theory. Teoksessa J. Carroll (toim.) *HCI Theories and Frameworks: Toward a Multidisciplinary Science*. San Francisco: Morgan Kaufmann Publishers, 291-324.
- Bevan, N. 1995. Usability is quality of use. Teoksessa *Proceedings of the HCI International 1995 Conference on Human Computer Interaction*. Mahwah, NJ: Lawrence Erlbaum Associates, 349-354.
- Bevan, N. 2009a. Extending quality in use to provide a framework for usability measurement. Teoksessa M. Kurosu (toim.) *Proceedings of the HCI International 2009, San Diego, California, USA*: Springer, 13-22.
- Bevan, N. 2009b. What is the difference between the purpose of usability and user experience evaluation methods? Teoksessa *Proceedings of the UXEM'09 Workshop*.
- Beyer, H. & Holtzblatt, K. 1998. *Contextual design: A customer-centered approach to system design*. San Diego, CA, USA: Academic Press.

- Blythe, M., Overbeeke, K., Monk, A. & Wright, P. (toim.). 2003. *Funology: from usability to enjoyment*. Dordrecht: Kluwer.
- Boehm, B. 1981. *Software engineering economics*. Englewood Cliffs, NJ: Prentice Hall.
- Boehm, B. 1986. A spiral model of software development and enhancement. *ACM SigSoft Software Engineering Notes* 11 (4), 21-42.
- Booch, G., Rumbaugh, J. & Jacobson, I. 1998. *The unified modeling language user guide*. Reading, MA: Addison-Wesley.
- Brangier, E. 2007. *Besoin et interface*. Teoksessa J. Akoka & I. Comyn-Wattiau (toim.) *Encyclopédie des Nouvelles Technologies*. Paris: Vuibert, 1070-1084.
- Bratman, M. 1987. *Intention, plans, and practical reason*. Cambridge, Massachusetts and London, England: Harvard University Press.
- Brentano, F. 1973. *Psychologie vom empirischen Standpunkt [Psychology from an empirical point of view]*. Hamburg, Germany: Felix Meiner. (Alkuteos julk. 1874)
- Bødker, S. 1989. A human activity approach to user interfaces. *Human Computer Interaction* 4 (3), 171-195.
- Bødker, S. 1991. *Through the interface—a human activity approach to user interface design*. Hillsdale, NJ: Lawrence Erlbaum.
- Bødker, S. 2006. When second wave HCI meets third wave challenges. *Teoksessa Proceedings of the Nordic Conference on Human-Computer Interaction (NordiCHI)* Oslo: ACM Press, 1-8.
- Callele, D., Neufeld, E. & Schneider, K. 2006. emotional requirements in video games. *Teoksessa the proceedings of the 14th IEEE International Requirements Engineering Conference*, 299 - 302.
- Caplan, D., Waters, G. & DeDe, G. 2008. Specialized verbal working memory for language comprehension. *Teoksessa A. R. A. Conway, C. Jarrold, M. J. Kane, A. Miyake & J. Towse (toim.) Variation in Working Memory*. New York: Oxford University Press, 272-302.
- Card, S.K., Moran, T.P. & Newell, A. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum.
- Carroll, J. 1997. Human computer interaction: psychology as science of design. *Annual Review of Psychology* 48, 61-83.
- Carroll, J. M. 2002. *Human-Computer Interaction in the New Millennium*. New York: ACM Press.
- Carroll, J. M. 2003. *HCI models, theories, and frameworks: towards a multidisciplinary science*. San Francisco: Morgan Kaufmann.
- Carroll, J. M. & Campbell, R. L. 1989. Artefacts as psychological theories: the case of human-computer interaction. *Behaviour and Information Technology* 8 (4), 247-256.
- Carroll, J. M. & Olson, J. R. 1988. Mental models in human-computer interaction. *Teoksessa M. Helander (toim.) Handbook of Human-computer Interaction*. Amsterdam: Elsevier, 46-56.

- Carroll, J.M. & Rosson, M.B. 1992. Getting around the task-artifact cycle: how to make claims and design by scenario. *ACM Transactions on Information Systems* 10 (2), 181-212.
- Carroll, J. M., Thomas, J. C. & Malhotra, A. 1980. Presentation and representation in design problem solving. *British Journal of Psychology* 71 (1), 143-153.
- Cassab, H., Tuunanen, T., Peffers, K., Gengler, C., Hui, W. & Virtanen, V. 2010. Discovery of new service concepts for diverse markets. *Service Science* 2 (3), 178-196.
- Chen, L., Gillenson, M.L. & Sherrell, D.L. 2002. Enticing online consumers: an extended technology acceptance perspective. *Information and Management* 39 (8), 705-719.
- Cheng, B. & Atlee, J. 2007. Research directions in requirements engineering. *Teoksessa Proceedings of the Future of Software Engineering 2007*. Minneapolis, MN, USA: IEEE-CS Press, 285-303.
- Cho, V. 2006. A Study of the roles of trusts and risks in information-oriented online Legal Services using an integrated model. *Information & Management* 43 (4), 502-520.
- Chung L., Nixon B., Yu E. & Mylopoulos J. 2000. *Non-functional requirements in software engineering*. Boston: Kluwer Academic.
- Clark, R. E. & Estes, F. 1996. Cognitive task analysis. *International Journal of Educational Research* 25 (5), 403-417.
- Clarke, B. 1987. Knowledge acquisition for real-time knowledge-based systems. *Teoksessa Proceedings of the First European Workshop on Knowledge Acquisition for Knowledge Based Systems*, Reading University, UK.
- Coble, J. M., Karat, J. & Kahn, M. G. 1997. Maintaining a focus on user requirements throughout the development of clinical workstation software. *Teoksessa proceedings of the CHI'97*, New York: ACM Press, 170-177.
- Cockton, G. 2004. Value-centered HCI, *Teoksessa Proceedings of the NordiCHI Conference*. New York: ACM Press, 149-160.
- Coiera, E. 2003. Interaction design theory. *International Journal of Medical Informatics* 69 (2-3), 205-222.
- Cooper, R. B. 2000. Information technology development creativity: A case study of attempted radical change. *MIS Quarterly* 24 (2), 245-276.
- Crandall B, Klein G, Hoffman, R.R. 2006. *Working minds: a practitioner's guide to cognitive task analysis*. Cambridge: MIT Press.
- Damanpour, F. & Wischnevsky, J. D. 2006. Research on innovation in organizations: distinguishing innovation-generating from innovation-adopting organizations. *Journal of Engineering and Technology Management* 23 (4), 269-291.
- Davis, F.D. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly* 13 (3), 319-339.

- Davis, F.D., Bagozzi, R.P. & Warshaw, P.R. 1989. User acceptance of computer technology: a comparison of two theoretical models. *Management Science* 35 (8), 982-1003.
- Dardenne, A., van Lamsweerde, A. & Fickas, S. 1993. Goal-directed requirements acquisition. *Science of Computing Program* 20 (1-2), 3-50.
- Dearden, A. & Howard, S. 1998. Capturing user requirements and priorities for innovative interactive systems. *Teoksessa Proceedings of the Australasian Computer Human Interaction Conference*, 160-167.
- Deci, E.L., Connell, J.P. & Ryan, R.M. 1989. Self-determination in a work organization. *Journal of Applied Psychology* 74, 580-590.
- Desurvire, H., Caplan, M. & Toth, J. A. 2004. Using heuristics to evaluate the playability of games. *Teoksessa Extended Abstracts CHI'04*. New York: ACM, 1509-1512.
- Dey, I. 1993. *Qualitative data analysis: a user-friendly guide for social scientists*. London: Routledge.
- DiRomualdo, A. & Gurbaxani, V. 1998. Strategic intent for IT outsourcing. *Sloan Management Review* 39 (4) , 67-80.
- Dourish, P. 2001. *Where the action Is: the foundations of embodied interaction*. Cambridge, MA: MIT Press.
- Draper, S.W. 1999. Analysing fun as a candidate software requirement. *Personal Technologies* 3 (3), 117-122.
- Duncan, K. D. 1972. Strategies for the analysis of the task. *Teoksessa J. Hartley (toim.) Programmed Instruction: An Education Technology*. London: Butterworth.
- Duncan, K. D. 1974. Analytical techniques in training design. *Teoksessa E. Edwards & F. P. Leeds (toim.) The Human Operator and Process Control*. London: Taylor & Francis, 283 - 320.
- Elboushi, M. I. & Sherif, J. S. 1997. Object-oriented software design utilizing quality function deployment. *Journal of Systems and Software*, 38(2), 133-143.
- Engeström, Y. 1987. *Learning by expanding. An activity- theoretical approach to developmental research*. Helsinki: Orienta-Konsultit Oy.
- Engeström, Y., Miettinen, R. & Punamäki, R. 1999. *Perspectives on Activity Theory*. Cambridge: Cambridge University Press.
- Fenton N. & Pfleeger S. 1997. *Software metrics: A rigorous practical approach*. London: PWS Publishing.
- Fichte, J.G. 2006. *Tiedeopin perusta. Suomentaja ja selitysten laatija Ilmari Jauhiainen*. Helsinki: Gaudeamus (Alkuteos Grundlage der gesamten Wissenschaftslehre julk. 1794-1795).
- Fishbein, M. & Ajzen, I. 1975. *Belief, attitude, intention and behavior: An Introduction to Theory and Research*. Reading, MA: Addison-Wesley.
- Fisher, G., Giacardi, E., Eden, H., Sugimoto, M., & Ye, Y. 2005. Beyond binary choices: Integrating individual and social creativity. *International Journal of Human-Computer Studies* 63, 482-512.

- Fisk, A. D. & Eggemeier, F. T. 1988, Application of automatic/controlled processing theory to training tactical command and control skills: 1. Background and task analytic methodology. Teoksessa Proceedings of the Human Factors Society 32nd annual meeting. Santa Monica, CA: Human Factors Society, 1227 - 1231.
- Fogg, B. & Tseng, H. 1999. The elements of computer credibility. Teoksessa Proceedings of the Conference on Human Factors in Computing Systems (CHI). New York: ACM Press, 80-87.
- Folmer, E. & Bosch, J. 2004. Architecting for usability: a survey. *The Journal of Systems and Software* 70 (1-2), 61-78.
- Gallagher, J.P. 1979. Cognitive/information processing psychology and instruction: Reviewing recent theory and practice. *Instructional Science* 8, 393-414.
- Gay, G. & Hembrooke, H. 2004. Activity-centered design. Cambridge, MA: MIT Press.
- Gentner, D.R. & Grudin, J. 1990. Why good engineers (sometimes) create bad interfaces. Teoksessa Proceedings of CHI'90 Human Factors in Computing Systems, Seattle, 277-282.
- Giorgi, A. 1997. The theory, practice and evaluation of the phenomenological method as a qualitative research procedure. *Journal of Phenomenological Psychology* 28 (2), 235-260.
- Giorgi, A. 2005. The phenomenological movement and research in the human sciences. *Nursing Science Quarterly* 18 (1), 75-82.
- Goguen, J. & Jirotko, M. (toim.). 1994. Requirements engineering: social and Technical Issues. London: Academic Press.
- Goldstein, B. 1999. Sensation and perception, Pacific Grove, CA: Brooks/Cole.
- Gollwitzer, P. M. & Bargh, J. A. 1996. (toim.) *The Psychology of Action: Linking Motivation and Cognition to Behavior*. New York: Guilford.
- Gómez-Pérez, A., Fernández-López, M. & Corcho, O. 2004. Ontological engineering: with examples from the areas of knowledge management, e-commerce and the semantic web. 2.painos. London: Springer-Verlag.
- Good, M., Spine, T.M., Whiteside, J, & George, P. 1986. User-derived impact analysis as a tool for usability engineering. Teoksessa Proceedings of the CHI'86 Human Factors in Computing Systems Conference. New York: ACM Press.
- Goodwin, C.J. 2007. Research in psychology: methods and design. 5.painos. New York: Wiley.
- Gopalakrishnan, S. & Damanpour, F. 1997. A review of innovation research in economics sociology and technology management. *Omega* 25 (1), 15-28.
- Gordon, S. E., Schmierer, K. A. & Gill, R. T. 1993. Conceptual graph analysis: knowledge acquisition for instructional systems design. *Human Factors* 35 (3), 459 - 481.
- Gould J.D. & Lewis, C. 1985. Designing for usability: key principles and what designers think. *Communications of the ACM* 28 (3), 300-311.

- Greenbaum, J. & Kyng, M. (toim.). 1991. *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, NJ: Lawrence Erlbaum.
- Greenhalgh, T., Robert, G., Macfarlane, F., Bate, P. & Kyriakidou, O. 2004. Diffusion of innovations in service organizations: systematic review and recommendations. *Milbank Quarterly* 82 (4), 581-629.
- Grube P. & Schmid, K. 2008. Selecting creativity techniques for innovative requirements engineering. *Teoksessa Proceedings of Workshop on Multimedia and Enjoyable Requirements Engineering (MERE'08)*, 32-36.
- Grudin, J. 1990. The computer reaches out: the historical continuity of interface design. *Teoksessa Proceedings of the Conference on Human Factors in Computing Systems (CHI)*. New York: ACM, 261-268.
- Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J. & Cajander, Å. 2003. Key principles for user-centred system design. *Behaviour & Information Technology* 22 (6), 397-409.
- Haasbroek, L.J. 1993. Advanced human-computer interfaces and intent support: a survey and perspective. *Teoksessa Proceedings of SMC '93: 'Systems Engineering in the Service of Humans'*, 350-355.
- Hampshire, S. 1965. *Thought and action*. London: Chatto & Windus.
- Han, S.H., Yun, M.H., Kwahk, J. & Hong, S.W. 2001. Usability of consumer electronic products. *International Journal of Industrial Ergonomics* 28 (3-4), 143-151.
- Harrison, S., Tatar, D. & Sengers, P. 2007. The three paradigms of HCI. *Teoksessa Alt.chi of the SIGCHI Conference on Human Factors in Computing Systems*. New York: ACM.
- Hartson, H.R. 1998. Human-computer interaction: Interdisciplinary roots and trends. *The Journal of Systems and Software*, 43 (2), 103-118.
- Hasan, B. 2006. Delineating the effects of general and system-specific computer self-efficacy beliefs on IS acceptance. *Information and Management* 43 (5), 565-571.
- Hassenzahl, M. 2002. The effect of perceived hedonic quality on product appealingness, *International Journal of Human-Computer Interaction*, 13 (4), 479-497.
- Hassenzahl, M. 2003. The thing and I: understanding the relationship between user and product. *Teoksessa M. Blythe, C. Overbeeke, A. F. Monk, & P. C. Wright (toim.) Funology: From Usability to Enjoyment*. Dordrecht: Kluwer, 31-42.
- Hassenzahl, M. 2008. User experience (UX): towards an experiential perspective on product quality. *Teoksessa IHM '08 Proceedings of the 20th French-Speaking Conference on Human-Computer Interaction (Conférence Francophone sur l'Interaction Homme-Machine)*. New York: ACM, 11-15.
- Hassenzahl, M., Diefenbach, S. & Göritz, A. 2010. Needs, affect, and interactive products - Facets of user experience. *Interacting with Computers* 22 (5), 353-362.
- Hassenzahl, M. & Roto, V. 2007. Being and doing: A perspective on User Experience and its measurement. *Interfaces* 72, 10-12.

- Heeks, R. 2002. Information Systems and developing countries: failure, success and local improvisations. *The Information Society* 18 (2), 101-112.
- Heeks, R. & Kenny, C. 2002. ICTs and Development: Convergence or divergence for developing countries? Teoksessa S. Krishna & S. Madon (toim.) *Proceedings of 7th International Working Conference of IFIP WG9.4 Information and Communication Technologies and Development: New Opportunities, Perspectives and Challenges*, Bangalore : Indian Institute of Management, 29-44.
- Heidegger, M. 1962. Being and time. Kääntäjät John Macquarrie ja Edward Robinson. *Being and time*. New York: Harper & Row (Alkuteos julk. 1927).
- Heiskala, R. 2007. Social innovations: structural and power perspectives. Teoksessa T.J. Hämäläinen & R. Heiskala (toim.) *Social Innovations, Institutional Change and Economic Performance*, Cheltenham: Edward Elgar, 52-79.
- Heiskari, J., Kauppinen, M., Runonen, M. & Männistö, T. 2009. Bridging the gap between usability and requirements engineering. Teoksessa 17th IEEE International Requirements Engineering Conference. Washington, DC, USA: IEEE, 303-308.
- Helkama, K., Myllyniemi, R. & Liebkind, K. 2010. *Johdatus sosiaalipsykologiaan*. 3.-8.painos. Helsinki: Edita Prima Oy.
- Hertzum, M. & Clemmensen, T. 2012. How do usability professionals construe usability? *International Journal of Human-Computer Studies*, 70 (1), 26-42.
- Heymans P. & Dubois, E. 1998. Scenario-based techniques for supporting the elaboration and the validation of formal requirements. *Requirements Engineering Journal* 3 (3-4), 202-218.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. 2007. *Tutki ja kirjoita*. 13.painos. Keuruu: Otavan Kirjapaino Oy.
- Hoffman, R. R., & Militello, L. 2009. *Perspectives on cognitive task analysis: Historical origins and modern communities of practice*. Boca Raton, FL: CRC Press/Taylor & Francis.
- Hollan J. D., Hutchins E. & Kirsh D. 2000. Distributed cognition: A new foundation for human-computer interaction research. *ACM ToCHI* 7 (2), 174-196.
- Holliday, A. 2008. *Doing and writing qualitative research*. 2.painos. London: Sage.
- Hollnagel, E. (toim.). 2003. *Handbook of cognitive task design*. Mahwah, NJ: Lawrence Erlbaum.
- Hollnagel, E. & Woods, D.D. 2005. *Joint cognitive systems. Foundations of Cognitive Systems Engineering*. Boca Raton, FL: CRC Press, Taylor & Francis Group.
- Husserl, E. 1970. *Logical investigations*. Vols. I&II (Kääntäjä J. N. Findlay). New York: Humanities Press. (Alkuteos julk. 1900).
- Husserl, E. 1983. *Ideas pertaining to a pure phenomenology and to a phenomenological philosophy. First book* (Kääntäjä F. Kersten). The Hague: Martinus Nijhoff. (Alkuteos julk. 1913).

- Husserl, E. (Alkuteos julk. 1962). *Phenomenological psychology* (Kääntäjä J. Scanlon). The Hague: Nijhoff.
- Hutchins, E. 1995. How a cockpit remembers its speeds. *Cognitive Science* 19, 265-288.
- Hutchins, E. 1996. *Cognition in the Wild*. Cambridge, MA: MIT Press.
- Hutchins, E. L. & Klausen, T. 1996. Distributed cognition in an airline cockpit. Teoksessa Y. Engeström & D. Middleton (toim.) *Cognition and Communication at Work*. New York, NY: Cambridge University Press, 15-34.
- Hutchins, E. L. and Palen, L. 1997. Constructing meaning from space, gesture, and speech. Teoksessa L. B. Resneck, R. Saljo, C. Pontecorvo, and B. Burge (toim.) *Tools, and Reasoning: Essays in Situated Cognition*. Vienna, Austria: Springer-Verlag.
- IEEE Std IEEE-Std-830-1998. 1998. IEEE Recommended Practice for Software Requirements Specification, IEEE CS Press.
- Iivari, J. & Iivari, N. 2006. Varieties of user-centeredness. Teoksessa *Proceedings of the 39th Annual Hawaii International Conference on System Sciences*. Los Alamitos: IEEE Computer Society Press, 125-153.
- Irwin, F.W. 1971. *Intentional behavior and motivation*. Philadelphia, JB: Lippincott.
- ISO 9126-1. 2001. International Organization for Standardization [ISO] and International Electrotechnical Commission [IEC], 2001, June. *Software engineering: Product quality-Quality model*. (Standards No. 9126-1). Geneva, Switzerland: ISO.
- ISO 9241-1. 1997. *Ergonomic Requirements for Office Work with Visual Display Terminals (VDTS) - Part 1: General Introduction*. Geneva, Switzerland: International Organisation for Standardisation.
- ISO 9241-11. 1998. *Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability*. Geneva, Switzerland: International Organisation for Standardisation.
- ISO 9241-210. 2010. *Ergonomics for human-system interaction - Part 210: Human-centred design for interactive systems*. Geneva, Switzerland: International Organisation for Standardisation.
- ISO 13407. 1999. *Human-Centered Design for Interactive Systems*. Geneva, Switzerland: International Organisation for Standardisation.
- ISO/TR 16982. 2002. *Ergonomics of Human-System Interaction - Usability Methods Supporting Human-Centred Design*. Geneva, Switzerland: International Organisation for Standardisation.
- Ivory, M. Y. & Hearst, M. A. 2001. The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys* 33 (4), 470-516.
- Jacobson, I., Christerson, M., Jonsson, P. & Overgaard, G. 1992. *Object-oriented software engineering: A Use-Case Driven Approach*. Wokingham, UK: Addison-Wesley.
- James, W. 1890. *The principles of psychology* (Vol. II). New York: Holt.

- Johnson M. & Toiskallio, K. 2005. Fansites as sources for user research: Case Habbo Hotel. Teoksessa Proceedings of the 28th Information Systems Research Seminar in Scandinavia (IRIS'28).
- Jokela, T., Iivari, N., Matero, J. & Karukka, M. 2003. The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. Teoksessa Proceedings of the Latin American Conference on Human-Computer Interaction. New York: ACM, 53-60.
- Jokela, T., Koivumaa, J., Pirkola, J., Salminen, P. & Kantola, N. 2006. Methods for quantitative usability requirements: A case study on the development of the user interface of a mobile phone. *Personal Ubiquitous Computing* 10 (6), 345-355.
- Jokivuori, P. & Hietala, R. 2007. Määrällisiä tarinoita. Monimuuttujamenetelmien käyttö ja tulkinta. Porvoo: WSOY.
- Järvinen, P. & Järvinen, A. 2004. Tutkimustyön metodeista. Tampere: Opinpajan kirja.
- Kamppuri, M., 2011. Theoretical and methodological challenges of cross-cultural interaction design. Itä-Suomen yliopisto. Joensuu: Itä-Suomen yliopisto. Väitöskirja.
- Kane, M. J., Conway, A. R. A., Miura, T. K. & Colflesh, G. J. 2007. Working memory, attention control and the N-back task: A question of construct validity. *Journal of Experimental Psychology: Learning, Memory and Cognition* 33 (3), 615-622.
- Kankainen, A. & Oulasvirta, A. 2003. Design ideas for everyday mobile and ubiquitous computing based on qualitative user data. Teoksessa N. Carbonell & C. Stephanidis (toim.) *User Interfaces for All*. Berlin, Germany: Springer-Verlag, 458-464.
- Kantola, N. & T. Jokela. 2007. Determining high level quantitative usability requirements: a case study. Teoksessa Proceedings of the 12th International Conference on Human-Computer Interaction: Interaction Design and Usability, Heidelberg: Springer, 536-543.
- Kaptelinin, V. 1992. Human computer interaction in context: The activity theory perspective. Teoksessa J. Gornostaev (Ed) Proceedings of EWHCI'92 Conference. Moscow: ICSTI.
- Kaptelinin, V. 1996. Computer-mediated activity: functional organs in social and developmental contexts. Teoksessa B. A. Nardi (toim.) *Context and Consciousness: Activity theory and Human-Computer Interaction*. Cambridge, MA: MIT Press, 45-68.
- Kaptelinin, V. & Nardi, B. 1997. Activity theory: basic concepts and applications. Teoksessa Proceedings of the Computer-Human Interaction (CHI) interest group, Conference on Human Factors in computing systems, Atlanta, Georgia: ACM.
- Kaptelinin, V. & Nardi, B. 2006. Acting with technology: activity theory and interaction design. Cambridge, MA: MIT Press.
- Kaptelinin, V., Nardi, B., Bødker, S., Carroll, J., Hollan, J., Hutchins, E. & Winograd, T. 2003. Post-cognitivist HCI: second-wave theories. Teoksessa

- Proceedings of the Conference on Human Factors in Computing Systems (CHI). New York: ACM Press, 692-693.
- Kaptelinin, V., Nardi, B.A. & Macaulay, C. 1999. The activity checklist: a tool for representing the "space" of context. *Interactions* 6 (4), 27-39.
- Karat, J. 1997. Evolving the scope of user-centered design. *Communications of the ACM* 40 (7), 33-38.
- Kieras, D. E. 1988. Towards a practical GOMS model methodology for user interface design. Teoksessa M. Helander (toim.) *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier, 135-158.
- Kim, J. & Han, S.H. 2008. A methodology for developing a usability index of consumer electronic products. *International Journal of Industrial Ergonomics* 38 (3-4), 333-345.
- Kirwan, B. & Ainsworth, L. K. 1992. *A guide to task analysis*. London: Taylor & Francis.
- Klein, G. & Militello, L. G. 1998. Cognitive task analysis. Teoksessa Workshop of Human Factors and Ergonomics Society 42nd Annual Meeting, No.12. Chicago, Illinois.
- Kline, R.B. 2009. *Becoming a behavioral science researcher. A guide to producing research that matters*. New York: The Guilford Press.
- Koskela, T. & Väänänen-Vainio-Mattila, K. 2004. Evolution towards smart home environments: empirical evaluation of three user interfaces. *Personal and Ubiquitous Computing* 8 (3-4), 234 - 240.
- Kotonya, G. & Sommerville, I. 2002. *Requirements engineering, processes and techniques*. Chichester: Wiley.
- Kraft, C. 2012. *User experience innovation : user centered design that works*. New York: Apress L. P.
- Kujala, S., Kauppinen, M. & Rekola, S. 2001. Bridging the gap between user needs and user requirements. Teoksessa N. Avouris & N. Fakotakis (toim.) *Advances in Human-Computer Interaction I*, Typorama Publications, 45-50.
- Kuo, F.Y. & Young, M.L. 2008. A study of the intention-action gap in knowledge sharing practices. *Journal of the American Society for Information Science and Technology* 59 (8), 1224-1237.
- Kuutti, K. 1996. Activity theory as a potential framework for human computer interaction research. Teoksessa B.A. Nardi (toim.) *Context and Consciousness: Activity Theory and Human-Computer Interaction*, Cambridge: MIT Press, 17-44.
- Kuutti, K. 2011. Toiminnan teoria. Teoksessa A. Oulasvirta (toim.) *Ihmisen ja tietokoneen vuorovaikutus*. Helsinki: Gaudeamus Helsinki University Press, 62-87.
- Kwahk, J. & Han, S.H. 2002. A methodology for evaluating the usability of audiovisual consumer electronic products. *Applied Ergonomics* 33 (5), 419-431.
- Lamb, R. & Kling, R. 2003. Reconceptualizing users as social actors in information systems research. *MIS Quarterly* 27 (2), 197-235.

- Langdon, P., Persad, U. & Clarkson, P.J. 2010. Developing a model of cognitive interaction for analytical inclusive design evaluation. *Interacting with Computers* 22 (6), 510-529.
- Lapouchnian A. 2005. Goal-oriented requirements engineering: an overview of the current research, Depth Report, University of Toronto.
- Larman, C. & Basit, V. 2003. Iterative and incremental development. A Brief History. *IEEE Computer* 36 (6), 47-56.
- Law, E., Roto, V., Hassenzahl, M., Vermeeren, A. & Kort, J., 2009. Understanding, scoping and defining user experience: a survey approach. *Teoksessa Proceedings of the CHI 2009 Conference on Human Factors in Computing Systems*. New York: ACM, 719-728.
- Lehdonvirta, V., Wilska, T., & Johansson, M. 2009. Virtual consumerism: case Habbo Hotel. *Information, Communication & Society* 12 (7), 1059-1079.
- Leikas, J. 2009. Life-based design. Form of life as a foundation for ICT design for older adults. Jyväskylän yliopisto. Jyväskylä: Jyväskylän yliopisto. Väitöskirja.
- Leont'ev, A. N. 1974. The problem of activity in psychology. *Soviet Psychology* 13 (2), 4-33.
- Leont'ev, A. N. 1978. Activity, consciousness and personality. Englewood Cliffs: Prentice-Hall.
- Leont'ev, A. N. 1981. Problems of the development of the mind. Moscow: Progress.
- Leont'ev, A. N. 1989. The problem of activity in the history of Soviet psychology. *Soviet Psychology* 27 (1), 22-39.
- Lesgold, A., Rubinson, H., Feltovich, P., Glaser, R., Klopfer, D. & Wang, Y. 1988. Expertise in a complex skill: diagnosing x-ray pictures. *Teoksessa M. T.H. Chi, R. Glaser & M. J. Farr (toim.) The Nature of Expertise*. Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers, 311 - 342.
- Lewin, K. 1951. Intention, will, and need. *Teoksessa D. Rapaport (Eds. and Trans.) Organization and pathology of thought*. New York: Columbia University Press, 95-153.
- Liikkanen, L. 2007. Design cognition for conceptual design. *Teknillinen korkeakoulu TKK - Aalto yliopisto*. Espoo: Teknillinen korkeakoulu TKK. Väitöskirja.
- Lindgaard, G., Dillon, R., Trbovich, P., White, R., Fernandes, G., Lundahl, S. & Pinnamaneni, A. 2006. User needs analysis and requirements engineering: theory and practice. *Interacting with Computers* 18 (1), 47-70.
- Löwgren, J., Stolterman, E. 2005. Thoughtful interaction design: A design perspective on information technology. Cambridge, MA: MIT Press.
- Macintosh, N.B. & Daft, R.L. 1978. User department technology and information design. *Information and Management* 1 (3), 123-131.
- Macleod, M. & Rengger, R. 1993. The development of DRUM: a software tool for video-assisted usability evaluation. *Teoksessa Proceedings of The BCS HCI'93 Conference, Loughborough UK, Sept 1993*. Cambridge University Press, 293-309.

- Macleod, M. 1994. Usability: practical methods for testing and improvement. Teoksessa Proceedings of the Norwegian Computer Society Software '94 Conference.
- Maguire, M. & Bevan, N. 2002. User requirements analysis: a review of supporting methods. Teoksessa Proceedings of the IFIP 17th World Computer Congress - TC13 Stream on Usability, 133-148.
- Maiden N., Gizikis A. & Robertson, S. 2004a. Provoking creativity: imagine what your requirements could be like. *IEEE Software* 21 (5), 68-75.
- Maiden N., Manning S., Robertson S. & Greenwood, J. 2004b. Integrating creativity workshops into structured requirements processes. Teoksessa Proceedings of Designing Interactive Systems (DIS2004), 113-122.
- Maiden N. & Robertson, S. 2005 (a). Developing use cases and scenarios in the requirements process. Teoksessa Proceedings of the IEEE International Conference on Software Engineering (ICSE), 561-570.
- Maiden N. & Robertson, S. 2005 (b). Integrating creativity into requirements processes: Experiences with an air traffic management system. Teoksessa Proceedings of the IEEE International Conference on Requirements Engineering (RE), 105-116.
- Malle, B. F., & Knobe, J. 1997. The folk concept of intentionality. *Journal of Experimental Social Psychology* 33 (2), 101-121.
- Mauro, C. 2011. Why Angry Birds is so successful and popular: a cognitive teardown of the user experience. Web document. Available from <<http://www.mauronewmedia.com/blog/why-angry-birds-is-so-successful-a-cognitive-teardown-of-the-user-experience/>> November 20, 2012.
- Mayhew, D. J. 1999. The usability engineering lifecycle: A practitioner's handbook for user interface design. San Francisco: Morgan Kaufmann Publishers.
- McCracken, G. 1988. The long interview. Newbury Park: Sage.
- McFadzean E. 1998. The creativity continuum: towards classification of creative problem solving techniques. *Journal of Creativity and Innovation Management* 7(3), 131-139.
- McKeithen, K. B., Reitman, J. S., Reuter, H. H., & Hirtle, S. 1981. Knowledge organization and skill differences in computer programmers. *Cognitive Psychology* 13 (3), 307-325.
- McKeown, K., Kukich, K. & Shaw, J. 1994. Practical issues in automatic document generation. Teoksessa Proceedings of the Fourth Conference on Applied Natural-Language Processing (ANLP-1994), 7-14.
- Meiland, J.W. 1970. The nature of intention. London: Methuen.
- Metsämuuronen, J. 2008. Laadullisen tutkimuksen perusteet. 3.painos. Jyväskylä: Gummerus.
- Miller, G. A., Galanter, E. & Pribram, K. H. 1960. Plans and the structure of behavior. New York: Holt.
- Monk, A. 1998. Cyclic interaction: A unitary approach to intention, action and the environment. *Cognition*, 68 (2), 95-110.

- Moore, G.C. & Benbasat, I. 1991. The Development of an instrument to measure the perceived characteristics of adopting an information technology innovation. *Information Systems Research* 2 (3), 192-222.
- Moran, T. 1981. An applied psychology of the user. *Computing Surveys* 13 (1), 1-11.
- Mäntymäki, M. & Salo, J. 2012. Digital natives' purchasing behavior in Habbo Hotel. To appear in the proceedings of the 25th Bled eConference, Bled, Slovenia.
- Nardi, B. 1992. Studying context: a comparison of activity theory, situated action models and distributed cognition. *Teoksessa Proceedings East-West HCI Conference, St. Petersburg, Russia, August 4-8*, 352-359.
- Nardi, B. (toim.) 1996. *Context and Consciousness: Activity theory and Human-Computer Interaction*. Cambridge, MA: MIT Press.
- Nielsen, J. 1993. *Usability engineering*. San Francisco: Morgan Kaufmann.
- Nielsen, J. & Mack, R. 1994. *Usability inspection methods*. New York: Wiley.
- Nielsen, J. & Molich, R. 1990. Heuristic evaluation of user interfaces. *Teoksessa Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People 1990 (CHI 1990)*. New York: ACM, 249-256.
- Niiniluoto, I. 1987. *Truthlikeness*. Boston: D. Reidel.
- Nivala, A.-M., Brewster, S. & Sarjakoski, L. T. 2008. Usability evaluation of Web mapping sites. *The Cartographic Journal* 45 (2), 129-138.
- Norman, D. A. 1984. Stages and levels in human-machine interaction. *International Journal of Man-Machine Studies* 21 (4), 365 - 370.
- Norman, D. A. 1986. Cognitive engineering. In D.A. Norman & S. Draper (toim.) *User Centered System Design: New Perspectives on Human-Computer Interaction*. Hillsdale: Lawrence Erlbaum, 31-61.
- Norman, D. A. 1998. *The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution*. Cambridge, MA: MIT press.
- Norman, D. A. 2004. *Emotional design: why we love (or hate) everyday things*. New York: Basic Books.
- Norman, K. & Chin, J. 1988. The effect of tree structure on search performance in a hierarchical menu selection system. *Behaviour and Information Technology* 7 (1), 51-65.
- Nunamaker, J., Dennis, A., Valacich, J., Vogel, D. & George, J. 1991. Electronic meeting systems to support group work. *Communications of the ACM*, 34 (7), 40-61.
- Nuseibeh, B & Easterbrook, S. 2000. Requirements engineering: a roadmap. *Teoksessa The Future of Software Engineering, Special Issue 22nd International Conference on Software Engineering, ACM-IEEE*, 35-46.
- Obrist M., Roto, V. & Väänänen-Vainio-Mattila, K. 2009. User experience evaluation: do you know which method to use? *Teoksessa CHI Extended Abstracts: 2763-2766*.

- Olson, G. M. & Olson, J. S. 2003. Human-computer interaction: psychological aspects of the human use of computing. *Annual Review of Psychology* 54 (1), 491-516.
- Oulasvirta, A. & Saariluoma, P. 2004. Long-term working memory and interrupting messages in human computer interaction. *Behaviour & Information Technology* 23 (1), 53-64.
- Oulasvirta, A. & Saariluoma, P. 2006. Surviving task interruptions: investigating the implications of long term working memory theory. *International Journal of Human Computer Studies* 64 (10), 941-961.
- Owen, A. M., McMillan, K. M., Laird, A. R., & Bullmore, E. 2005. NBack working memory paradigm: A meta-analysis of normative functional neuroimaging studies. *Human Brain Mapping* 25 (1), 46-59.
- Paganelli, L., & Paternò, F. 2002. Intelligent analysis of user interactions with Web applications. Teoksessa *Proceedings of the 7th International Conference on Intelligent User Interfaces (ACM IUI'02)*. New York: ACM, 111-118.
- Parkkola, H. 2006. *Designing ICT for mothers. User Psychological Approach*. Jyväskylän yliopisto. Jyväskylä: Jyväskylän yliopisto. Väitöskirja.
- Parsons, T. 1949. *The structure of social action*. New York: Free Press.
- Pashler, H. 1998. *The Psychology of Attention*. Cambridge, Mass.: MIT Press.
- Paternò, F., & Ballardin, G. 1999. Model-aided remote usability evaluation. Teoksessa A. Sasse & C. Johnson (toim.) *Proceedings of the IFIP TC13 7th International Conference on Human-Computer Interaction (INTERACT '99)*. Amsterdam: IOS Press, 434-442.
- Paternò, F., & Ballardin, G. 2000. RemUSINE: A bridge between empirical and model-based evaluation when evaluators and users are distant. *Interacting with Computers* 13 (2), 229-251.
- Patnaik, D. & Becker, R. 1999. Needfinding: the why and how of uncovering people's needs. *Design Management Journal* 10 (2), 37-43.
- Pennington, N. 1987. Stimulus structures and mental representations in expert comprehension of computer programs. *Cognitive Psychology* 19 (3), 295-341.
- Pinelle, D., Gutwin, C. & Greenberg, S. 2003. Task analysis for groupware usability evaluation: modeling shared-workspace tasks with the mechanics of collaboration. *ACM TOCHI*, 10 (4), 281-311.
- Pinelle, D., Wong, N. & Stach, T. 2008. Heuristic evaluation for games: usability principles for video game design. Teoksessa *Proceedings CHI 2008*, 1453-1462.
- Pol, E. & Ville, S. 2009. Social innovation: buzz word or enduring term. *The Journal of Socio-Economics*, 38 (6), 878-885.
- Preece, J., Rogers, Y. & Sharp, H. 2002. *Interaction design : beyond human-computer interaction*. New York: Wiley.
- Propp, S., Buchholz, G. & Forbrig, P. 2009. Integration of usability evaluation and model-based software development. *Journal Advances in Engineering Software* 40 (12), 1223-1230.

- Quinn, J.B., Baruch, J.J. & Zien, K.A. 1996. Software-based innovation. *Sloan Management Review* 37 (4), 11-24.
- Rasmussen, J., Duncan, K. & Leplat, L. 1987. *New technology and human error*. New York: John Wiley & Sons.
- Read, J. C., MacFarlane, S. J. & Casey, C. 2001. Measuring the usability of text input methods for children. *Teoksessa Proceedings of HCI2001*, 559-572.
- Read, J. C., MacFarlane, S. J. & Casey, C. 2002. Endurability, engagement and expectations: measuring children's fun. *Teoksessa Proceedings of interaction design and children*, 189-198.
- Reason, J. 1987. Generic errors modeling systems (GEMS): a cognitive framework for locating common error forms. *Teoksessa J. Rasmussen, K. Duncan, & L. Leplat (toim.) New Technology and Human Error*, New York: John Wiley & Sons, 63 - 86.
- Repko, A.F. 2012. *Interdisciplinary research: process and theory*. 2.painos. London: Sage Publications.
- Robertson, S. 2001. Requirements trawling: techniques for discovering requirements. *International Journal of Human-Computer Studies* 55 (4), 405-421.
- Rolland, C., Prakash, N., & Benjamin, A. 1999. A Multi-model view of process modeling. *Requirements Engineering* 4 (4), 169-187.
- Rose, J. 2012. *Software innovation: Eight work-style heuristics for creative system developers*. Aalborg: Aalborg University.
- Rossi, M. & Tuunanen, T. 2010. A method and tool for rapid consumer application development. *International Journal of Organisational Design and Engineering* 1 (1), 109-125.
- Rosson, M. B., & Carroll, J. M. 2002. *Usability engineering: scenario-based development of human-computer interaction*. San Francisco: Morgan-Kaufmann.
- Roth, E., Scott, R., Deutsch, S., Kuper, S., Schmidt, V., Stilson, M. & Wampler, J. 2006. Evolvable work-centered support systems for command and control: creating systems users can adapt to meet changing demands. *Ergonomics*, 49(7), 688-705.
- Roto V., Obrist M., Väänänen-Vainio-Mattila K. 2009. User experience evaluation methods in academic and industrial contexts. *Teoksessa Proceedings of UXEM 09 workshop*.
- Rotter, J. B. 1966. Generalized expectancies for internal versus external control of reinforcement. *Psychological Monographs*, 80 (1, Whole No. 699).
- Royce, W. 1970. Managing the development of large software systems: concepts and techniques. *Teoksessa Proceedings of 9th International Conference on Software Engineering*. Francisco: IEEE Computer Society Press, 1-9.
- Rubin, J. 1994. *Handbook of usability testing: how to plan, design, and conduct effective tests*. New York: Wiley.
- Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. & Lorenson, W. 1991. *Object-oriented modelig and design*. Englewood Cliffs: Prentice-Hall.

- Ryder, J. M. & Redding, R. 1993. Integrating cognitive task analysis into instructional systems development. *Educational Technology Research and Development* 41(2), 75 - 96.
- Ryder, J. M. & Zachary, W. W. 1991, Experimental validation of the attention switching component of the COGNET framework. Teoksessa *Proceedings of the human factors society 35th annual meeting*, Santa Monica, CA, 72 - 76.
- Saariluoma, P. 2004. Käyttäjäpsykologia [User psychology]. Porvoo: WSOY.
- Saariluoma, P. 2011. Käyttäjä. Teoksessa A. Oulasvirta (toim.) *Ihmisen ja tietokoneen vuorovaikutus*. Helsinki: Gaudeamus Helsinki University Press, 45-61.
- Saariluoma, P., & Oulasvirta, A. 2010. User psychology: Re-assessing the boundaries of a discipline. *Psychology*, 1 (5), 317-328.
- Salomon, G. (toim.) 1993. *Distributed Cognitions: Psychological and Educational Considerations*. Cambridge: Cambridge University Press.
- Schraagen, J. M., Chipman, S. F. & Shute, V. J. 2000. State-of-the-art review of cognitive task analysis techniques. Teoksessa J. M Schraagen, S. F. Chipman & V. J. Shute (toim.) *Cognitive Task Analysis*. Mahwah, NJ: Lawrence Erlbaum Associates, 467-487.
- Seamster, T. L., Redding, R.E. & Kaempf, G. L. 1997. *Applied cognitive task analysis in aviation*. Aldershot: Ashgate.
- Seffah, A., Donyaee, M., Kline, R.B. & Padda, H.K. 2006. Usability measurement and metrics: a consolidated model. *Software Quality Control* 14 (2), 159-178.
- Seligman, M,E,P. 1975. *Helplessness: on depression, development, and death*. San Francisco: Freeman.
- Shackel, B., 1984. The concept of usability. Teoksessa J., Bennett, D., Case, J. Sandelin, & M. Smith (toim.) *Visual Display Terminals*. Englewood Cliffs: Prentice-Hall, 45-87.
- Shackel, B. 1991. Usability: Context, framework, definition, design and evaluation. Teoksessa B. Shackel & S. Richardson (toim.) *Human Factors for Informatics Usability*. Cambridge, MA, USA: Cambridge University Press, 21-38.
- Shepherd, A. 2000. HTA as a framework for task analysis. Teoksessa J. Annett & N.A. Stanton (toim.) *Task Analysis*. London: Taylor & Francis, 9-23.
- Shneiderman, B. 2002. *Leonardo's Laptop*. Cambridge: MIT Press.
- Shultz, T. R. 1980. Development of the concept of intention. Teoksessa W. A. Collins (toim.) *The Minnesota Symposium on Child Psychology (Vol.13)*. Hillsdale, NJ: Erlbaum, 131-164.
- Schütz, A. 2007. *Sosiaalisen maailman merkityksellinen rakentuminen*. Suomentaja Veikko Pietilä, jälkisanat Tapio Aittola. Tampere: Vastapaino. (Alkuteos: *Der sinnhafte Aufbau der sozialen Welt. Eine Einleitung in die verstehende Soziologie* julk. 1932).
- Sommerville, I. 2005. Integrated requirements engineering: A Tutorial. *IEEE Software* 22 (1), 16-23.

- Spinoza, B. 1994. *Etiikka. Suomentaja ja selitysten laatija Vesa Oittinen*. Helsinki: Gaudeamus (Alkuteos: *Ethica ordine geometrico demonstrata 1674*).
- Spool, J.M., Scanlon, T. & Snyder, C. Product usability: survival techniques. *Teoksessa Proceedings of CHI '97*. New York: ACM Press, 154-155.
- Stanton, N.A. 2006. Hierarchical task analysis: developments, applications and extensions. *Applied Ergonomics* 37 (1), 55-79.
- Stanton, N.A., Baber, C. 2002. Error by design: methods for predicting device usability. *Design Studies* 23 (4), 363-384.
- Stelmaszweska, H., Fields, B. & Blandford, A. 2004. Conceptualising user hedonic experience. *Teoksessa D.J. Reed, G. Baxter & M. Blythe (toim.) Proceedings of ECCE-12, 12th European Conference on Cognitive Ergonomics, Living and Working with Technology, York, UK, 83-89*.
- Stone, D., Jarrett, C., Woodroffe, M., & Minocha, S. 2005. *User interface design and evaluation*. San Francisco: Elsevier/Morgan Kaufmann.
- Sutcliffe, A. 1998. Scenario-based requirements analysis. *Requirements Engineering Journal* 3 (1), 48-65.
- Sutcliffe, A. 2002. *User-centered requirements engineering: theory and practice*. New York: Springer-Verlag.
- Sutcliffe, A., Maiden, N., Minocha, S. & Manuel, D. 1998. Supporting scenario-based requirements engineering. *IEEE Transactions on Software Engineering* 24 (12), 1072-1088.
- Swanson, E. B. 1994. Information systems innovations among organizations. *Management Science* 40 (9), 1069-1092.
- Syrjäläinen, E. 1994. *Etnografinen opetuksen tutkimus: kouluetnografia*. Teoksessa L., Syrjälä, S., Ahonen, E., Syrjäläinen, & S. Saari (toim.) *Laadullisen tutkimuksen työpajoja*. Rauma: Kirjayhtymä Oy, 68-112.
- Szajna, B. 1996. Empirical evaluation of the revised technology acceptance model. *Management Science* 42 (1), 1996, 85-92.
- Tang, H.K. 1998. An integrative model of innovation in organizations. *Technovation* 18 (5), 297-309.
- The Agile Manifesto. 2001. *Manifesto for agile software development*. <http://www.agilemanifesto.org>.
- Tolman, E. C. 1932. *Purposive behavior in animals and men*. New York: Appleton-Century-Crofts.
- Tsalgatiadou, A., Veijalainen, J., Markkula, J., Katasonov, A., & Hadjiefthymiades, S. 2003. Mobile e-commerce and location-based services: technology and requirements. *Teoksessa Proceedings of the 9th Scandinavian Research Conference on Geographical Information Services, Espoo, 1-14*.
- Tulving, E. & Craik, F.I.M. (toim.) 2000. *The Oxford Handbook of Memory*. Oxford: Oxford University Press.
- Tuunanen, T., Myers, M.D. & Cassab, H. 2008. Challenges of consumer information systems development: the case of interactive television services. *Teoksessa D. Avison, G.M. Kasper, B. Pernici, I. Ramos & R.*

- Dewald (toim.) *Advances in Information Systems Research, Education and Practice*. Boston: Springer, 89-100.
- Tractinsky, N. 1997. Aesthetics and apparent usability: empirically assessing cultural and methodological issues. *Teoksessa Proceedings of the Conference on Human Factors in Computing systems (CHI)*, Atlanta. New York: ACM, 115-122.
- Tractinsky, N., Katz, A.S. & Ikar, D. 2000. What is beautiful is usable. *Interacting with Computers* 13, 127-145.
- Travis, D. 2011. 13407 is dead. Long live 9241-210! Web document. Available from <<http://www.userfocus.co.uk/articles/iso-13407-is-dead.html>> November 11, 2012.
- Tyldesley, D. A. 1988. Employing usability engineering in the development of office products. *The Computer Journal* 31 (5), 431-436.
- Uusitalo, H. 1999. *Tiede, tutkimus ja tutkielma. Johdatus tutkielman maailmaan*. Helsinki: WSOY.
- van Lamsweerde A. 2000. Requirements engineering in the year 00: a research perspective. Invited Keynote Paper, Proc. ICSE'2000: 22nd International Conference on Software Engineering. ACM Press, 5-19.
- van Lamsweerde, A. 2009. *Requirements engineering – from system goals to UML models to software specifications*. London: Wiley.
- van Lamsweerde, A., Darimont, R. & Ph. Massonet. 1995. Goal-directed elaboration of requirements for a meeting scheduler: problems and lessons learnt. *Teoksessa Proceedings of RE'95 - 2nd International IEEE Symposium on Requirements Engineering*, 194-203.
- Venkatesh, V., Morris, M. G., Davis, G. B. & Davis, F. D. User acceptance of information technology: Toward a Unified View. *MIS Quarterly* 27 (3), 425-478.
- Veryzer, R. W. 2003. Marketing and the development of innovative new products. *Teoksessa L.V. Shavinina (toim.) International Handbook on Innovation*, New York: Pergamon/Elsevier Science Ltd., 845-855.
- Viukari, L. 2010. *Tieto- ja viestintätekniikkavälitteisen palvelun kehittämisen kolme diskurssia. Väitöskirja*. Jyväskylä: Jyväskylän yliopisto.
- von Goethe, J.W. (1808/1832/1965). *Faust I-II*. Suomentaja Otto Manninen. Helsinki: Otava.
- Von Hippel, E. & Von Krogh, G. 2003. Open source software and the private-collective innovation model: issues for organization science. *Organization Science* 14 (2), 209-233.
- Von Krogh, G., Sabeth, S. & Lakhani, K.R. 2003. Community, joining, and specialization in open source software innovation: a case study. *Research Policy* 32, 1217-1241.
- Vredenburg, K., Isensee, S. & Righi, C. 2002. *User centered design: an integrated approach*. Upper Saddle River, NJ: Prentice Hall PTR.
- Vroom, V. H. 1964. *Work and motivation*. New York: Wiley.

- Vujovic, S. & Ulhøi, J.P. 2008. Online innovation: the case of open source software development. *European Journal of Innovation Management* 11 (1), 142-156.
- Vygotsky, L.S. 1978. *Mind in society: the development of higher psychological processes*. Cambridge: Harvard University Press.
- Väänänen-Vainio-Mattila, K. 2011. Käytettävyys ja käyttäjäkeskeinen suunnittelu. Teoksessa A. Oulasvirta (toim.) *Ihmisen ja tietokoneen vuorovaikutus*. Helsinki: Gaudeamus Helsinki University Press, 102-126.
- Väänänen-Vainio-Mattila, K., Roto, V. & Hassenzahl, M. 2008. Now let's do it in practice: user experience evaluation methods in product development. Teoksessa *CHI 2008 Proceedings - Workshops*, Florence, Italy, 3961-3964.
- Warshaw, P. R. & Davis, F. D. 1985. Disentangling behavioral intention and behavioral expectation. *Journal of Experimental Social Psychology* 21 (3), 213-228.
- Weber, M. 1922. *Wirtschaft und Gesellschaft (Economy and Society)*. Mohr: Tübingen.
- Wei, J. & Salvendy, G. 2004. The cognitive task analysis methods for job and task design: review and reappraisal. *Behaviour & Information Technology* 23 (4), 273-299.
- Winograd, T. & Flores, F. 1987. *Understanding computers and cognition: A new foundation for design*. Reading, MA: Addison-Wesley.
- Wixon, D. and Wilson, C. 1997. The usability engineering framework for product design and evaluation. Teoksessa M.G. Helander, T.K. Landauer & P.V. Prabhu (toim.) *Handbook of Human-Computer Interaction*. 3.painos. Amsterdam: Elsevier Science, 653 - 688.
- Woods, D.D. 1998. Commentary: designs are hypotheses about how artefacts shape cognition and collaboration. *Ergonomics* 41 (2), 169-173.
- Woods, D.D. & Dekker, S. 2000. Anticipating the effects of technological change: A new era of dynamics for human factors. *Theoretical Issues in Ergonomics Science* 1 (3), 272-282.
- Wright, P.C., & Blythe, M. 2007. User experience research as an inter-discipline: towards a UX Manifesto. *Proceedings of the Workshop on Towards a UX Manifesto, in conjunction with HCI Conference 2007*. Lancaster, UK, 65-70.
- Xie, H. 2000. Shifts of interactive intentions and information-seeking strategies in interactive information retrieval. *Journal of the American Society for Information Science* 51 (9), 841-857.
- Yi, M.Y. & Hwang, Y. 2003. Predicting the use of web-based information systems: self-efficacy, enjoyment, learning goal orientation, and the technology acceptance model. *International Journal of Human-Computer Studies* 59 (4), 431-449.
- Yu, E. 1997. Towards modeling and reasoning support for early-phase requirements engineering. Teoksessa *Proceedings of the 3rd IEEE International Symposium on Requirements Engineering*. Los Alamitos, CA: IEEE Computer Society Press, 226-235.

Yu, E. 1999. Strategic modeling for enterprise integration. Teoksessa Proc. of the 16th World Congress of the International Federation of Automatic Control. Beijing, China, Pergamon: Elsevier Sciences, 127-132.

ORIGINAL PAPERS

I

A FRAMEWORK FOR INTENTION-DRIVEN REQUIREMENTS ENGINEERING OF INNOVATIVE SOFTWARE PRODUCTS

by

Leppänen, M., Lamminen, J. & Saariluoma, P. 2011

In J. Pokorny, V. Repa, K. Richta, W. Wojtkowski, H. Linger, C. Barry & M. Lang (Eds) Information Systems Development. New York: Springer, 417-428

Reproduced with kind permission by Springer.

A Framework for Intention-Driven Requirements Engineering of Innovative Software Products

Mauri Leppänen and Juha Lamminen and Pertti Saariluoma

Abstract Requirements engineering is highly challenging particularly when designing innovative software products. This is so because there are no corresponding products, ultimate needs of actors are difficult to capture, the products may have unforeseeable impacts on the actors' behavior, and it is hard to find out how value-added and competitive the product actually is. In this paper, we propose a novel framework for intention-driven requirements engineering of innovative software products, which combines technological, social and business viewpoints. We illustrate its use with a short example related to the domain of web mapping services and augmented reality.

1 Introduction

Requirements engineering (RE) is the most challenging discipline in the systems development lifecycle. Requirements are often ambiguous, incomplete, redundant and contradictory, due to stakeholders' divergent perspectives, terminology, and interests [14]. They are frequently changing because stakeholders are not able to say what they really need. Requirements engineering also involves reluctant participation, misperception and disagreement. Many RE languages and frameworks have been proposed to help making requirements more precise, complete, and consistent [20, 3]). These techniques are mainly targeted to late-phase requirements engineering, and less attention has been given to consider how the desired software product would meet personal or business goals, and why the product is needed.

Mauri Leppänen
Department of Computer Science and Information Systems, P.O. Box 35 (Agora), FI-40014
University of Jyväskylä, Finland, e-mail: mauri@cs.jyu.fi

Juha Lamminen
Department of Computer Science and Information Systems, P.O. Box 35 (Agora), FI-40014
University of Jyväskylä, Finland, e-mail: juha.e.lamminen@jyu.fi

Pertti Saariluoma
Department of Computer Science and Information Systems, P.O. Box 35 (Agora), FI-40014
University of Jyväskylä, Finland, e-mail: pertti.saariluoma@jyu.fi

The emergence of novel IC technologies has created new possibilities to add value through innovative products and services. Realizing this potential requires creativity and innovative acts across the whole product design lifecycle. *Innovation* means a multi-stage process whereby organizations transform ideas into new/improved products, service or processes, in order to advance, compete and differentiate themselves successfully in their marketplace [1].

Requirements engineering is particularly challenging when it concerns the design of an innovative software product. First, there is no corresponding product from which to learn. Second, a new product may affect, in an unforeseen manner, ways of how actors behave and business is run. Third, it is hard to discover how value-added and competitive the product-to-be would actually be, and for whom, among the networked companies. These challenges have been tried to be met by moving the focus onto early phase RE [31,32], by developing goal-oriented approaches [4, 6, 15], and by crafting creativity techniques [9, 22] for requirements engineering. Nevertheless, the situation is still unsatisfying.

We argue that in RE of innovative products creativity should be a built-in, all-bracing property, the intentions of human and business actors have to be taken as the basis for all the RE activities, and the way of requirements engineering should be tailored based on the situation at hand (cf. customer-driven vs. market-driven vs. technology-driven). We suggest a novel framework for RE of innovative software products, which aims to satisfy the abovementioned demands. The framework combines the technological, social and business viewpoints. It is aimed to be used for analyzing and comparing existing RE methods in terms of how they address, emphasize, and integrate creativeness and the three viewpoints. It also provides a basis for considerations of how to enhance existing RE methods.

The paper is organized as follows. In Section 2 we describe the framework, and in Section 3 we illustrate it with a small example. In Section 4, a short literature review of relevant literature is presented. Section 5 concludes with the summary.

2 Framework for Requirements Engineering

We define *requirements engineering* (RE) as a creative process in which stakeholders and designers work together to create and concretize ideas for a new product [18, 19]. We build our RE framework on four concepts: innovation, user-centredness, goal-orientedness, and multi-viewpoint. First, the overall structure of the framework is designed to boost the capturing and elicitation of new ideas for software products. Second, the RE process starts with finding out the intentions and needs of users, and users are expected to have an active role in the RE process. Third, goals provide rationale for requirements that operationalize them and help detect and manage conflicts among the requirements [15]. Fourth, requirements engineering is considered from the business, social, and technology viewpoints. The focus of the first viewpoint is on value creation, distribution and consumption (cf. [8]). Social viewpoint concerns members of communities and their

social relationships. Technology viewpoint is applied to bring forward and discuss new technological innovations.

The RE framework is composed of five interrelated RE activities: envisioning, user profiling, business context analysis, social context analysis, and goal setting (see Figure 1). In the following, the activities are described in more detail.

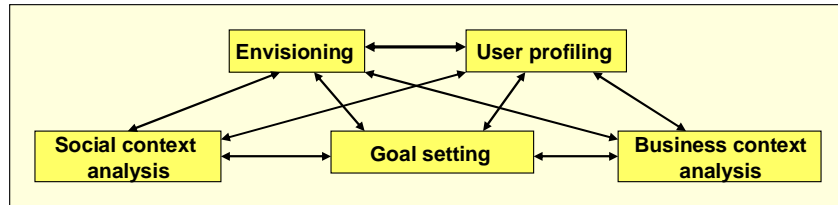


Fig. 1. Framework for Requirements Engineering

2.1 Envisioning

Envisioning means innovative and creative action aiming at evoking and eliciting ideas for new products and/or new ways of using the products. Product design often starts with a vague desire, concept, or image of something new. Work is done, through innovative techniques [9, 20], to elaborate it towards a more concrete vision. To ensure its feasibility, the vision should be shared within the community and be translatable into reality. Creativity is a key factor to successful envisioning. It is the interplay between the ability and process by which an individual or group produces a product that is both novel and useful within some social or business context [25]. *Technical innovations* improve existing features or facilitate the introduction of completely new features. *Need innovations* happen when a hidden need is found or an existing need is abstracted into a more profound need (cf. [12]). *Business innovations* help companies increase their profits through the use of a new product and by re-engineering their business processes.

Envisioning can be driven by technology, business, or human needs. In the first case, a new IT intensive product is first invented and then an attempt is made to find out contexts in which its features could be utilized [16]. In the second case, the focus is on searching “enablers” for innovative business solutions [32]. In the third case, a social context is analyzed to understand what profound needs the human actors have, and to figure out how the needs could be satisfied through a new product. Our RE framework support all of these approaches. There is a large range of creative models and techniques available for envisioning [9, 22].

2.2 *User profiling*

User profiling aims at distinguishing user categories and characterizing them in terms of relevant features. Typically, the features include psychological characteristics (e.g., attitude, motivation), knowledge and experience (e.g. typing skill, task experience), job and task characteristics (e.g. frequency of use), and physical characteristics (e.g. color blindness) [21]. Characteristics can be concretized and combined by attaching them to personas, or fictional people [5]. Early personas are sketches that are later elaborated toward more detailed characters. Personas should not replace active user participation.

Data for user profiling is collected through interviews and/or user profile questionnaires, and indirectly from marketing personnel [20]. As new products built upon emerging technologies seldom have existing counterparts, it is important also to gather information about the users' subjective opinions and preconceptions regarding their future (cf. cultural probes [7]).

2.3 *Business context analysis*

If the product is to act as an “enabler” for innovative business solutions, not just as a means of automating well-established business processes, one has to obtain a deep understanding about the domain. This means learning about the interests, priorities and abilities of various business players. Also, to ensure that the idea of a new product will really add value, it is necessary to analyze its profitability. The purpose of *business context analysis* is to model the context, objectives, and processes of the business entity for which a new software product is to be designed, in order to better understand its ICT needs and potentials. A business context means a web of networked companies and/or organizations that are established to provide goods and/or services to consumers.

There are several approaches and models for business context analysis (e.g., business process models, workflow models, cultural models). We do not propose any new approach, but utilize two of them that are particularly suitable for early-phase requirements engineering. They are the e^3 value approach [8] and the i^* framework [31, 32, 33].

In the e^3 value approach [8], a business context is viewed from a commercial perspective. It is seen as a multi-actor network in which economic value is created, distributed and consumed through a software product. An innovative idea means to find new economic value propositions that are yet unknown to the market and significantly change the way a company does business [8]. The e^3 value approach, as it is integrated into our RE framework, starts from an innovative idea resulting from envisioning. Based on this, a baseline model is constructed. The model shows business actors, value activities and value objects, as well as concepts related to value exchange through the distribution of value objects among

the business actors [8]. Construction of the model may yield more new ideas which can be iteratively elaborated through envisioning and perhaps through social context analysis. Based on the improved understanding of the business context and a clearer conception of the economic value of a new innovative idea, it is possible to elicit goal statements for the product and its use.

In the i^* framework [33], early requirements are assumed to involve actors who depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished. The goals are analyzed and elaborated into functional and non-functional requirements of the product-to-be. The key concept is actor. Organizational actors are viewed as having intentional properties such as goals, beliefs, abilities, and commitments. Actors are strategic in the sense that they are concerned about opportunities and vulnerabilities, and seek rearrangements of their environments that would better serve their interests [33]. The i^* framework includes the strategic dependency (SD) model and the strategic rational (SR) model. The former is used for describing the dependency relationship among various actors in the business context. The latter is used to describe stakeholder interests and concerns, and how they might be addressed by various configurations of products and environments [32].

Both of these approaches emphasize the importance of understanding the business context, motivations, and rationales (the “Whys”). The way of executing business context analysis depends on the situation at hand. For instance, if economic value is important, an e^3 value model is first constructed. The approaches can also be used to analyze the “eco-systems” of competitors and combine the results with information got from market analysis, in order to contemplate the competitiveness of the product.

2.4 Social context analysis

The purpose of *social context analysis* is to make sense of the motives and actions of the members in a given social community. A *social community* means a group of people who share common characteristics or interest and is perceived or perceiving itself as distinct in some respects from the larger society with which it exists [29]. Communities can be established based on family or friendship relationships, ideological views, hobbies, work, etc. Novel IC technologies (e.g. Facebook, Twitter, LinkedIn, Bebo, and MySpace) have substantially helped establishing new communities and networking within and between them.

Resulted from the first iterations in envisioning and user profiling, there exist preliminary conceptions about the relevant contexts and actors. Here, these conceptions are elaborated and analyzed. Within a family, for instance, the actors are the father, the mother, and children, and possibly grand parents and other relatives [10]. The next step is to recognize the intentions of the human actors. The purpose

is to analyze the underlying rationale and purpose of what people are doing: what are they trying to achieve, and why are they trying to achieve it?

During social context analysis, problems are uncovered and analyzed to find out whether ICT in some form could support social actions. The analysis may result in more elaborated ideas on a desired product, or it may lead to the conclusion that no product is needed. Another approach is to start with considering how an existing technology, perhaps in a new form and/or in a novel manner, could be utilized by the community. In parallel to the work in goal setting, work here continues with describing actions the actors are doing to reach their intentions and goals. Features of the product are outlined, and user tasks are modeled and analyzed to help discover main functionalities and qualities of the product. There are various models and techniques that can be used to describe user tasks in the social context: e.g., scenarios, (essential) use cases, use case templates, task decomposition trees, and work flow models.

2.5 Goal setting

Goal setting means the activity by which human and business actors' intentions are captured and refined into goals and ultimately specific, preferably measurable, requirements for a new product. An *intention* is a mental state of the actor, which motivates and regulates actions [3]. During this process, the goals and requirements are discussed, negotiated, formalized and prioritized. At its best, the requirements become concise, feasible, precise, complete, consistent and verifiable. As our focus is on early requirements engineering, we do not discuss formalization.

A *goal* is the reason for which something is done, made, used etc. Goals can be formulated at different levels of abstraction, ranging from high-level, motivational and strategic concerns to low-level, technical concerns [14]. From the business viewpoint, goals are objectives of the business organization, which guide decisions at various levels. From the human actor's viewpoint, goals are conditions or states of affairs that the actor would like to achieve. The lower-level goals concern objectives related to the concrete use of a product. A *requirement* specifies properties (functional, structural, physical, etc.) of the product-to-be [17]. A *feature* is a property of the product. It is first a design feature, then an implementation feature, and finally a usage feature (see Figure 2).

Goals are elicited, elaborated and analyzed in parallel to social context analysis and business context analysis. If the product is aimed for a social community, intentions and actions of the members give the starting point for elicitation. If the product is to be designed for business, business goals of and relationships between the business actors provide a basis for goal elicitation. In case the product has some features analogical to existing products, problems and deficiencies encountered in their use can be negated and used as anti-goals for the product-to-be.

Goals can also be got from marketing groups, technical support groups, and derived from competitive analysis.

Once goals have been identified, they are refined progressively into lower-level goals until they involve the use of the product. This process is done by asking the HOW questions. Another approach is to derive more abstract goals from those already identified by asking WHY these goals exist. This way it may be possible to refine them and find their sub-goals that were originally left undetected [15]. The goals and requirements can be described in AND/OR trees or networks. As all the goals and requirements cannot be satisfied, some prioritization is necessary (for prioritization techniques see [11]).

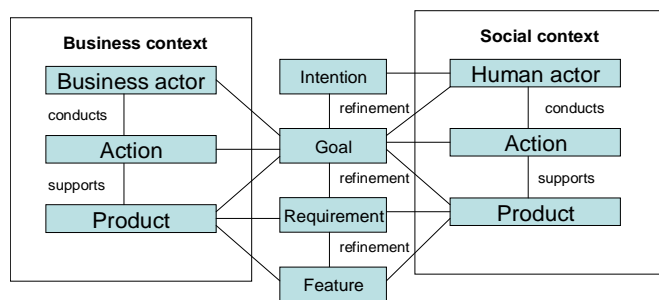


Fig. 2. Main concepts of the framework

3 Example

In this section, we illustrate the RE framework with a hypothetical example. Let us suppose that we are designing a new product to help tourists and start with the following vague ideas: planning trips should be fun, experiences during trips should be enriched, and networking with and communication between people with the same interests should be encouraging. The product is to be built upon three novel technologies: location-based services (LBS), distributed geographic information systems (GIS), and augmented reality (AR).

Most LBS offer basic navigation and routing services (cf. GoogleLatitude). We are especially interested in web map services (cf. Google Maps, Yahoo! Maps, Microsoft Virtual Earth, MapQuest, ArcWeb) in the form of distributed GIS that enhance the accessibility and dissemination of geographical capabilities and knowledge to internet users [28]. AR facilitates overlying labels or other virtual information over the real world objects (e.g. scene, building) observed through a camera attached to a computer, thus helping information contextualization and localization [30].

In the following, we describe how the activities of our framework can be applied to innovate the novel software product. Our case mainly applies technology-driven approach to RE. Due to the space limit, the description is very indicative. Intentions and goals are not discussed separately (cf. Goal setting) but mentioned in connection with the other activities.

Envisioning. We seek innovative solutions from three perspectives: technology, social and business. For finding technology innovations, we first recognize that there already exists a wide range of web mapping products. Our task is to invent a product with new features which make it desirable and competitive. We decide to build upon three promising concepts: geoportal, online annotation editing, and collaboration support. *Geoportals* are websites providing entries to geographical content on the Web [27]. Through them it is possible to provide tourists with a large assortment of existing web map functionalities and services [28]. Today, only a few augmented reality (AR) applications allow the *online creation and editing of annotations* [30]. When AR becomes popular on mainstream devices, there will be a large group of users who will be able to add content. This potential we want to exploit. We believe that with online annotation techniques an explosion of user created content, similar to the increase of online context with Web 2.0 applications, will be experienced [30]. Thereupon, the emergence of new forms of *social networking and cooperation* related to traveling can be also expected.

User profiling. The World Tourism Organization defines tourists as persons who "travel to and stay in places outside their usual environment for more than twenty-four (24) hours [...] for leisure, business and other purposes [...]" A tourist wants to obtain new experiences (e.g. roads with hairpin bends), learn exotic cultures or new things, or raise his/her social status (cf. visits in glamorous places). Tourists differ from each other in pre-knowledge, motivational, psychological, educational and physical terms. These differences should be taken into account in the functionality and interaction design of the product-to-be.

Social context analysis. We consider the context of traveling in three stages: trip planning, travelling and retrospective recalling and discussions of the experiences. The way a trip is planned depends on the motives of a human actor. Here, we assume that tourists are mainly interested in attractions. They want to know, among others, which kinds of places and attractions there are, how to reach them, where to have lunch and accommodation. Trip planning should also be fun, not stressful as it is usually. In addition to searching information related to geographical objects through common web service functionalities, tourists want to make their personalized maps and/or geotags. Because travelers plan less and less ahead of a trip and engage in more on-route and in-destination planning, there is a need to support "ad-hoc decisions" on restaurants or attractions. Tourists want also to be active members of social communities. Therefore, they should be facilitated to search for other travelers with similar profiles, interests and travel experiences, and their personalized maps (social bookmarking) [27].

During the trip, a tourist needs many kinds of guidance, not only that provided by GPS navigators. To offer a tourist richer experiences from places and attractions, the product should provide virtual information, attached with objects seen in

the real world. This way, they can “see”, for instance, how the building that is now in ruins looked like in the past. Tourists also want alternative viewpoints from which to look at objects around them. For example, it should be possible to select “eyeglasses” through which (s)he is provided with virtual information related to architecture, history, sport, or culinary art in the certain place. Tourists want also to include location related items (hotels, attractions, monuments, restaurants) on their personal maps and enrich them with feedback and experience of the places.

To refresh memories and share experiences with people, the product should allow to create new networks based on geographical routes and location-items (map networking [27]), and to use them for communication on interesting topics.

Business context analysis. The product connects a wide range of stakeholders, including tourists and their social networks, tourist bureaus, restaurants, pizzerias, hotels, museums, art galleries, etc. In this multi-actor network, economic value is created, distributed and consumed through the software product [8]. There is a large variety of ways of how tourism companies can exploit a product like ours in their business [27]: localized exploitation, internal integration, business process redesign, business network redesign, and business scope redefinition. Companies can generate greater business benefits when they increase their level of exploitation. The product-to-be should provide a large range of options for exploitation. Next, we only give some examples of them.

Map networking transforms a trip planning process to a more collaborative and social decision making process where social networks with other travelers are incorporated into value chains and become co-creators and co-producers of travel services [27]. A hotel’s website does not only allow customers to search for, contribute and read user-generated content, but the hotel can also use this content for developing its new services [27]. Virtual information attached to places and attractions in web maps enables a tourism business update their information even on an hourly basis and provide, for example, a special deal for the day if the restaurant is short of customers or a hotel is having a low occupancy rate [24].

To recognize the involved stakeholders and to examine their complex value-added relationships, we should build an e^3 model [8]. The model shows, among others, key tourism business actors, value activities and value objects. In addition, to elaborate goals, beliefs, abilities and commitments of the business actors and strategic dependencies between them, we may find the strategic dependency (SD) model and the strategic rational (SR) model [32] useful.

Concluding from business context analysis, we can state that the product-to-be should provide tourist companies with new opportunities to redesign their operations, internally and/or externally, and to develop new collaborative business models by involving new partners and/or users-customers into their value chains and systems.

4 Related work

There are a large number of studies on innovation and creativity in requirements engineering (RE). Nguen and Shanks [23] present a theoretical framework for understanding creativity in RE. Kauppinen et al. [12] identify three main opportunities for innovations: discovering hidden user and customer needs, inventing new product features, and supporting feature development. Maiden et al. [19] present a scenario-driven RE process, called RESCUE, that integrates human activity modeling, system goal modeling, and creativity techniques. Maiden et al. [18] apply theories from cognitive science to build creative models and working methods in air traffic management domain. Grube et al. [9] propose a framework to select creativity techniques for requirements elicitation. Gordijn et al. [8] consider RE from business perspective and distinguish three stakeholder-type related viewpoints: value, business process and information system viewpoints.

There is also a wide range of studies on goal-oriented approach to RE. Chung et al. [4] present a goal-based framework for clarifying and prioritizing non-functional requirements. The KAOS methodology [6] contains a rich set of formal analysis techniques and three types of models: goal model, object model, and operation model. *i** [31, 32] is an agent-oriented modeling framework that supports the modeling activities before the system requirements are formulated. Shibaoka et al. [26] proposes a method called GOORE which deploys a domain ontology to support goal decomposition. There are also some goal-oriented requirements engineering methods, such as AWARE, AGORA, and Tropos.

We have strongly exploited earlier research on innovation to make creativity an all-bracing property of the framework. Our approach borrows ideas from goal-oriented approaches but goes further by emphasizing the importance of the human actors' intentions as the starting point for requirements engineering. Business context analysis has been built on [8] and [31, 32]. The RE research has also yielded multiple generic conceptual and functional frameworks for requirements engineering, such as those by IEEE, ISO and [13]. These are, however, mainly lifecycle-based, whereas our framework is focus-based. It shows which you should particularly focus on in requirements engineering.

In summary, our framework integrates creativity and the derivation of requirements from the intentions of human actors (cf. social context) and the goals of business actors (cf. business context).

5 Summary and Conclusions

We are experiencing challenging times in today's software product design. Real business potential is only gained from radically novel products, not from making improvements in existing products. A growing portion of products is directed to so-called consumer markets (cf. social media software, game industry), for which

it is typical that nobody knows how a new product should be like. In this situation, it is not enough to ask what is needed but why it is needed. This can only be done by deriving requirements from the intentions of human and business actors.

We have suggested a new intention-driven framework which combines the technological, social and business viewpoints for requirements engineering of innovative software products. It is composed of five activities: envisioning, user profiling, social context analysis, business context analysis, and goal setting. The framework can be used to analyze and compare existing RE methods in terms of how they address and integrate creativeness, user-centredness, business view, and goal-orientness. It also provides a basis for considerations of how to enhance existing RE methods, in order to make them better meet today's challenges. The framework equally applies to software product design for consumer markets as well as for company use. In the former, social context analysis is emphasized whereas in the latter business context and social context (within a company) are evenly important.

Our next step is to elaborate the framework to address, in more detail, interaction requirements engineering, especially regarding user experience, which is important to social media and other novel application domains. It is also important to pay attention to the so-called micro-innovation processes, i.e., thought processes, concepts and design thinking paradigms, which actually create new ideas of the products, and develop innovation techniques based on them. Thirdly, future research is needed to tailor the activities of the framework to be part of agile methods whose popularity is growing fast in practice.

References

1. Baregheh A., Rowley J., Sambrook S. 2009. Towards a multidisciplinary definition of innovation. *Management Decision* 47(8), 1323-1339.
2. Bickerton M., Siddiqi J. 1993. The classification of requirements engineering methods. In *Proc. of IEEE Symp. on Requirements Engineering*, 182-185.
3. Bratman M. 1987. *Intentions, plans, and practical reasons*. Harvard University Press, Cambridge, MA.
4. Chung L., Nixon B., Yu E., Mylopoulos J. 2000. *Non-functional requirements in software engineering*, Kluwer Academic, Boston.
5. Cooper A. 1999. *The inmates are running the asylum*. Macmillan Publ. Co.
6. Dardenne A., van Lamsweerde A., Fickas S. 1993. Goal-directed requirements acquisition. *Science of Computing Program* 20(1-2), 3-50.
7. Gaver B., Dunne T., Pacenti E. 1999. Design: cultural probes. *Interactions* 6(1), 21-29
8. Gordijn J., Akkermans J. 2003. Value-based requirements engineering: exploring innovative e-commerce ideas. *Req. Engineering* 8(2), 114-134.
9. Grube P., Schmid K. 2008. Selecting creativity techniques for innovative requirements engineering. In *Proc. of Workshop on Multimedia and Enjoyable Requirements Engineering (MERE'08)*, 32-36.
10. Hughes R. Jr., Hans J. 2001. Computers, the Internet, and families: a review of the role new technology plays in family life. *Journal of Family Issues* 22(6), 778-792.

- 12 Mauri Leppänen and Juha Lamminen and Pertti Saariluoma
11. Karlsson L., Höst M., Regnell B. 2006. Evaluating the practical use of different measurement scales in requirements prioritisation. In *Int. Symp. on Empirical Software Engineering (ISESE 2006)*, 326-335.
 12. Kauppinen M., Savolainen J., Männistö T. 2007. Requirements engineering as a driver for innovations. In *IEEE Requirements Engineering Conf.*, 15-20.
 13. Kotonya G. and Sommerville I. 1998. *Requirements Engineering, Processes and Techniques*. Chichester: John Wiley & Sons, England.
 14. van Lamsweerde A. 2009. *Requirements engineering – From system goals to UML models to software specifications*. John Wiley & Sons, England.
 15. Lapouchnian A. 2005. *Goal-oriented requirements engineering: an overview of the current research*, Depth Report, University of Toronto.
 16. Leffingwell D. 2007. *Scaling software agility – Best practices for large enterprises*. Addison-Wesley, Boston, USA
 17. Lin J., Fox M., Bilgic T. 1996. A requirements ontology for engineering design, *Concurrent Engineering*, 4, 279-291
 18. Maiden N., Gizikis A. Robertson S. 2004. Provoking creativity: imagine what your requirements could be like. *IEEE Software* 21(5), 68-75.
 19. Maiden N., Manning S., Robertson S., Greenwood J. 2004. Integrating creativity workshops into structured requirements processes. In: *Designing Interactive Systems (DIS2004)*, 113-122.
 20. Maiden N., Rugg G. 1996. ACRE: Selecting Methods for Requirements Acquisition. *Software Engineering Journal* 11(3), 183-192.
 21. Mayhew D. 1999. *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, Morgan Kaufmann.
 22. McFadzean E. 1998. The creativity continuum: towards classification of creative problem solving techniques. *Journal of Creativity and Innovation Management* 7(3), 131-139.
 23. Nguyen L., Shanks G. 2009. A framework for understanding creativity in requirements engineering. *Information and Soft. Technology* 51(3), 655-662
 24. Pan B., Crotts J., Muller B. 2007. Developing web-based tourist information tools using google map. In Sigala M., Mich L., Murphy J. (eds.), *Information and Communication Technologies in Tourism*, 503-512.
 25. Plucker J. 2003. *Creativity*. College of Arts and Sciences at Indiana Univ.
 26. Shibaoka M., Kaiya H., Saeki M. 2007. GOORE: goal-oriented and ontology driven requirements elicitation method, In Hainaut J.-L. et al. (eds.), *ER Workshops 2007*, Springer LNCS 4802, 225-234.
 27. Sigala M., Marinidis D. 2009. Exploring the transformation of tourism firms' operations and business models through the use of Web map services. *European and Mediterranean Conf. on Information Systems (EMCIS2009)*, Izmir.
 28. Tait M. 2005. Implementing geoportals: applications of distributed GIS. *Computers, Environment and Urban Systems*, 29, 33-47.
 29. Webster 1989. *Webster's Encyclopedic Unabridged Dictionary of the English Language*. New York: Cramercy Books
 30. Wither J., diVerdi S., Höllerer T. 2009. Annotation in outdoor augmented reality. *Computers & Graphics* 33(6), 679-689.
 31. Yu E. 1997. Towards modeling and reasoning support for early-phase requirements engineering. In *Proc. of IEEE Symposium on Requirements Engineering*, 226-235
 32. Yu E. 1999. Strategic modeling for enterprise integration. In *Proc. of the 14th World Congress of the Int. Federation of Automatic Control*, Beijing, China.
 33. Yu E., Mylopoulos J. 1996. AI models for business process reengineering. *IEEE Expert* August, 16-23.

II

USER PSYCHOLOGY IN INTERACTION DESIGN - THE ROLE OF DESIGN ONTOLOGIES

by

Saariluoma, P., Parkkola, H., Honkaranta, A., Leppänen, M., & Lamminen, J.
2009

In P. Saariluoma & H. Isomäki (Eds) Future Interaction Design II, Springer,
Berlin, 69-86

Reproduced with kind permission by Springer.

User Psychology in Interaction Design: The Role of Design Ontologies

Pertti Saariluoma, Hanna Parkkola, Anne Honkaranta, Mauri Leppänen,
and Juha Lamminen

Abstract In the various forms of interaction design, it is essential to analyze, understand, and predict human behavior. This is equally true with devices such as information systems that are meant to interact with people. The importance of these problems has inspired scientists to develop numerous approaches to investigate and explicate human actions. However, they have mainly been characterized by intuitive and folk psychological approaches to the human mentality in interaction. To improve the scientific foundations of design, we present here a psychology-based approach to collecting user knowledge, as well as a related design practice. The former can be called user psychology and the latter the action-oriented design. User psychology is an approach that applies psychological knowledge and methods to analyzing and solving interaction design problems. It works to develop explanatory design practices so that it is possible to say on which psychological grounds one design alternative is better than another or why a solution is ineffective. One step toward improving explanatory design practices is to develop effective design ontologies to manage the design processes. Here, we discuss the nature of user psychological knowledge and analyze the process of developing respective ontological solutions.

1 Introduction

Understanding human–technology interaction (HTI) has become an important scientific and practical problem for several reasons. Firstly, the continuing development of information and communication technology (ICT) has opened new possibilities for human living, but the development makes interaction problems more complex. Secondly, it has been noticed that it is essential to make interactions easier with devices such as information systems if one wishes people to fully adopt new ICT products. Finally, emerging technologies, such as

P. Saariluoma (✉)
Department of Computer Science and Information Systems, University of Jyväskylä,
Jyväskylä, Finland
e-mail: ps@ju.fi

agents, ubiquitous, pervasive, and embedded systems, shall be very challenging for people to use because the direct keyboard and WYSIWYG-type interaction mode is often replaced by much less concrete forms of interaction. Improvement in human technology interaction is thus one important factor regulating the development of the rising ICT society.

If interactions are not well organized, people simply reject the new possibilities. The comparison between short message service (SMS) and wireless application protocol (WAP) demonstrates a typical example of the difficulties and provides interaction researchers with lessons for future. While SMS has become one of the biggest successes in mobile services, WAP, despite its sound basic idea as a mobile parallel to the Internet, did not work. It presupposed more complex interaction patterns than SMS does, and manufacturers failed to teach people the required skills (Parkkola, 2003).

New technologies make it possible to produce a multitude of new ICT services, but people know very little about them and are not necessarily interested in learning what is needed to adopt them (Kämäräinen & Saariluoma, 2007). This means that the vital cycle between invention and innovation may become unnecessarily slow, which slows down the development of information society, as well as substantially increases the development costs. The business logic is very simple here: Companies will have more money for development if more users adopt a new service quickly.

Designers of the future must abandon outdated thought models and think in new ways. One of the major challenges shall be finding a new way of interaction thinking. The immediate interaction with devices through traditional HCI forms is no longer the only vision (Carroll, 1997; Helander, Landauer, & Prabhu, 1997; Olson & Olson, 2003; Rosson & Carroll, 2002). Such thinking is, in many cases, too narrow because it does not cover the whole spectrum of human action and mentality. Devices and information systems are not central to human life and actions; they are tools for people. Moreover, it is not necessary to focus on finding uses for existing technologies, but rather to define and design the actions of people, and then find technical solutions to support these. This is why it is essential to build design practices on wider analyses of human nature, activities, and mentality. How such a holistic analysis can be realized is one of the major challenges for modern interaction-oriented cognitive and information systems science.

2 Development of Interaction Research

When designers begin to design a new product, they should have a solid idea about the users and their actions. For example, to improve family communication in the near future, they have to consider this problem area from several different human points of view, instead of limiting themselves to immediate device interaction or usability problems (Parkkola, 2006a; Parkkola, Saariluoma, & Berki, in press). This applies to all of the modern services. The development of the product

must be based on people's actions. The change in interaction design practice from the immediate interaction to a wider understanding of the human mentality makes it necessary to know if and how the traditional interaction analysis really can solve such problems.

The first attempts to understand users, such as the early psychology of programming or the early analysis of human roles, human action, and mentality, were recognized as problems, even though no scientific psychology was applied (cf. Royce, 1987). Experienced system designers considered how their own minds would react in these interaction situations and generalized from their own experiences to form conceptions of human thinking and reacting. One cannot claim these approaches had not been successful. Classic programming inventions, such as the early 3G programming languages (e.g., Fortran and BASIC) and programming paradigms (e.g., structured programming), had psychological motives as the foundations for learnability and memorability of code (Dahl, Dijkstra, & Hoare, 1972; Nielsen, 1993). Nevertheless, these early programming innovations did not necessarily lead to improved user understanding. Progress was based on standard intuitive folk psychology.

The second approach to answering the challenges of interaction, which also is a form of folk psychology, was to adopt usability experiments and iterative testing (e.g., prototyping models by Floyd, 1984; the spiral model by Boehm, 1988; or the analyses by Nielsen, 1993). Typically, these ideas led to systematic testing that no longer focused only on technical issues but also called attention to human performance (e.g., Knittle, Ruth, & Patton Gardner, 1986; Nielsen, 1993). However, the tests were not intended to systematically improve a psychological understanding of users' mental processes. The goal was simply to make improvements to the technologies under development at that particular moment. Of course, this approach has greatly improved the knowledge regarding various use problems.

The next method for obtaining information about users' actions were deployed via contextual task analysis (Beyer & Holtzblatt, 1998; Mayhew, 1999) and related traditions such as scenario-based analysis (Jarke, 1999; Tollmar & Persson, 2002; Torgny, 1998; Whiteside, Bennett, & Holtzblatt, 1988). In contextual task analysis, designers systematically observe what people do in organizations and base the systems design on this collected information. However, no psychological or social scientific knowledge is required or applied in analyzing the human dimension. This means that only rather practical questions can be asked and solved because any scientific elaboration of the problems would be difficult in the absence of psychological understanding of human behavior.

Though it would be unjust to claim that folk psychological approaches have not been useful, it would be equally incorrect to avoid pointing out the difficulties. Folk psychological thinking has its inherent problems (Stich, 1983). Its progress is relatively slow in absence of systematic theory testing and development. In interaction research, progress has taken place mostly in technical systems, but not in understanding the human mentality. Therefore,

interaction problems are addressed one at a time, often repeating the same process, and rarely advancing the field.

Folk psychology causes problems in developing design practices because the level of conceptualization is very low. While systems should have good learnability and memorability, these are no more than norms (Nielsen, 1993). Extensive analysis of the phenomena of learning and memory, however, would make it possible to say how learnability and memorability can be systematically improved (see Baddeley, 1997, for an introduction to the psychological tradition in these issues).

Good testing practices in ICT interaction analyses should be founded on psychologically reliable processes. This means that a sufficient number of experimental subjects are used and the experiments are controlled in a proper way. “Quick and dirty” methods may be practical in some situations, but one cannot reliably build usability testing on them. This is why it is important to take seriously the principles of modern experimental and empirical psychology in interaction testing, instead of relying on folk psychological intuitions.

Intuitive and unsystematic design thinking also causes difficulties in finding roles for different types of expertise. For example, the software tester’s role and the usability specialist’s role can be seen differently, depending on the ICT design models used. In the early ICT design models (e.g., the waterfall model by Royce, 1987) the software tester’s responsibility area was both technical testing and usability testing. The technical testing was based strictly on scientific principles while usability testing was fully intuitive, based on folk psychology without knowledge of scientific psychology.

Over the decades, the ICT design models have advanced, but the role of human specialists is still quite vague within these models. In modern ICT design models, the software tester’s main responsibility area can be, for example, to localize technical problems by using modern testing tools and documenting the problems for the developers and designers. The usability specialist’s responsibility area is larger and includes, for example, the usability requirements, usability design, usability tests, and usability development in general. Additionally, the usability specialist may have a technical background without any understanding of scientific psychology, in which case the usability knowledge might be fully grounded in folk psychology.

The critical question today is how far we can rely on folk psychological approaches when thinking about the properties of the human mind. These practices are based on the personal and private experiences of the researchers. While this type of psychological investigation prevailed for thousands of years, it was totally dismantled by the growth of scientific psychology between 1872 and 1912. Thus, folk psychology was rendered notoriously unreliable because, for example, several conditions of the human psyche, such as subconscious phenomena, could be investigated (Boring, 1950). Indeed, these problems remain equally unsolvable in folk psychological interaction design.

This critique is not meant to imply that everything done in folk psychological paradigms is incorrect. An analysis of human work contexts, for example, is a

necessity. The point made here is different and important: Folk psychology is not the most reliable, productive, or accurate way of conceptualizing and operationalizing the human mind and/or the elements of human activity within interaction research. The development of new possibilities in the ICT world creates parallel demands on the quality of human action analysis, and demands as well more scientifically rigorous procedures. The concepts, methods, and theoretical approaches of modern psychological thinking in design, as well as in testing, are needed to replace folk psychology.

3 Modern User Psychology

Certainly scientific psychological analysis had begun to develop even in the early stages of interaction research. Shneiderman (1976) applied basic experimental analysis to understand skills when analyzing programming and illustrated that human working memory is an essential theoretical construct when considering the behavior of programmers. Around the same time, a number of important psychological papers were published on the psychology of computer programming, which essentially improved understanding of user psychology (Adelson, 1981; Anderson, Farrell, & Sauers, 1984; Anderson & Jeffries, 1985; Carroll, Thomas, & Malhotra 1980; McKeithen, Reitman, Rueter, & Hirtle, 1981; Pennington, 1987). These papers among others helped introduce the paradigm of scientific psychological analysis of computer use. Much of the theoretical psychological knowledge of users' behavior and mentality has been developed under this paradigm.

The first extensive theoretical synthesis was made by Card, Moran, & Newell (1983) in the form of the GOMS (goals, operations, methods and selection rules) architecture to describe computer users in practical contexts, such as text editing. Later, this cognitive modeling paradigm was greatly extended and became fundamental in the scientific consideration of human mentality in use contexts (Anderson, Matessa, & Lebiere, 1997; Hoves & Young, 1997; Kieras & Meyer, 1997).

These early developments have led to technologically oriented and motivated research. The key challenges have been finding new uses for existing technologies, and testing iteratively existing solutions and prototypes of new systems (Rosson & Carroll 2002). The dependency of developers on the advancement of new technologies makes the usability research too slow to be of real use in designing technical interaction solutions. User analysis should direct the development of technologies. Consequently, one should begin with users as the point of departure for interaction analyses, rather than beginning with existing technologies.

User psychology research begins with the user and users' psychological properties. Moran (1981) suggested this term, and the construction of cognitive models is user psychology in practice (Card et al., 1983). This means the psychologist begins by analyzing what users are before any actual application

development and design work is begun. Certainly the concept of user psychology is every bit as valid as commonly used practices such as traffic psychology or industrial psychology.

The user psychological approach has always been overshadowed by the difficulties in communicating with industry (Carroll, 1997). Industrial people often view themselves from a practical point of view, and thus they at times may perceive the psychological user approach as too theoretical and distant from the actual design work. Indeed, this makes sense because relatively little energy has so far been invested in developing user psychological design methods.

The task of usability research is to develop uses for existing and emerging technologies and to test these to assure that they work (Rosson & Carroll, 2002). This work can have either scientific psychological or folk psychological foundations (Nielsen, 1993; Olson & Olson, 2003; Rosson & Carroll, 2002). Nevertheless, this kind of work is always bound by or related to a number of tacit user requirements that are built into the technology long before the aspects or facets of the actual user have been considered.

User psychology, however, has somewhat different goals. It investigates the psychological preconditions for use (Oulasvirta & Saariluoma, 2004, 2006; Saariluoma, 2004, 2005a,b). This means that user psychology must be able to explicate usability problems within psychological concepts and investigate them with psychological methods. The main goal is to replace the traditional intuitive interaction design with scientifically justifiable and reliable methods.

As a result, user psychology provides an approach for interaction designers. It allows them to elaborate the analysis of human behavior in several ways. Research outcomes are more reliable because the psychological methods have been well tested through practice. In addition user psychology allows designers to rely on diverse psychological knowledge bases in searching for usability solutions. While traditional usability engineering employs concepts such as memorability and learnability as criteria for smoothness of interaction, that is still quite different from using specific psychological knowledge about the human memory to improve these two aspects of interaction (e.g., Baddeley, 1997). Learnability and memorability are external measures, not the internal principles that would be applied to improve interaction (for the criteria, see Nielsen, 1993). More extensive application of user psychological knowledge makes it possible to base the design concepts on psychological findings and theories.

4 Towards Explanatory Design

When considering how to use user psychological knowledge in design, it is important to make distinctions between folk psychology and scientific psychology, and between science and intuition. The crucial difference can be found in explanatory ability. Science is capable of explaining its solutions, providing a

scientific answer—not an educated guess—to the “Why?” question (Hempel, 1965; Saariluoma, 1997). Modern user psychology, as we see it, must be explanatory and not intuitive or merely modeling. It answers questions with known and tested scientific laws, principles, or facts.

Take, for example, the question regarding why pop-ups are so disturbing to Internet users. Plausible explanatory answers could be because pop-ups cause an orientation reaction or because they interrupt the main task (Oulasvirta, 2006, for interruption; Posner, 1980, for orientation). Thus, an orientation reaction is a simple explanation, but it also provides the solid empirical understanding, analysis, and testing for what happens to users when they encounter pop-ups.

That user psychological design should be explanatory (Saariluoma, 2005a,b) is not a dramatic demand because design, in modern engineering practice, is always explanatory. Engineering design science begins quite straightforwardly with the idea that engineering must be based on the natural sciences and experimental knowledge (Pahl, Beitz, Feldhusen, & Groete, 2005, p. 1).

When an engineer designs a machine, he or she must take into account a number of natural laws. It is necessary, for example, to factor in the strength of the materials and the reliability of the construction (Pahl et al., 2005). Such rules of thumb as “if the designer is uncertain, let’s put twice the amount of steel to make the construction secure” do not belong to the today’s engineering. Nevertheless, such explanatory design practices in interaction design are seldom apparent. The argumentation is mostly folk psychological and excludes any deeper understanding of the human mentality. An exception is computational modeling, which seeks to apply basic cognitive properties in design processes (Card et al., 1983).

However, modeling and cognitive psychology offer only a partial solution to the problems of explanatory design. The demands of user understanding have gone beyond the limits of cognitive concepts: Cognitive concepts and theories cannot provide complete answers to all type of interaction problems. It is necessary to develop explanatory practices or frameworks from what is known about other dimensions of human mentality. Therefore, human dynamic and sociocultural properties are becoming increasingly more important (Saariluoma, 2005a,b).

The only way to bring user psychological design to the level of normal engineering design is to move to explanatory design practices and put aside folk psychological design paradigms. This does not necessarily mean that all intuitive elements should be rejected in interaction design: They have not been eliminated from engineering design practices either. There is always room for creative thinking; but even within creativity, interaction designers should be capable of explaining their key solutions based on scientific grounds and their standard design solutions should rest upon scientific grounds.

The development of modern technology sets new demands for the psychological analysis of users. For example, with the numerous variety of technologies and services available in the market, prospective users must be introduced

to and then learn to like the technology or service. In the Web service world, or perhaps the area of home electronics, the dynamic dimensions of the human mind shall undoubtedly play a bigger role. It is not sufficient that people understand how to use a new device or service, but they also have to like it and be motivated to use it (Parkkola, 2006b).

The versatility of user psychological problems requires that the variety of problems be resolved through an array of concepts, theories, and methods. Within scientific psychology, the main areas provide different perspectives on interaction. There is no unified psychological point of view on interaction; one may investigate it through either cognitive or socioemotional terms. The “right” way of looking at a problem depends on the nature of the problem.

Very few psychologically relevant issues would be unimportant in modern user psychological analyses (Saariluoma, 2004). On par with the traditional cognitive analyses of attention, memory, learning, and thinking are emotions and motives (Norman, 2004; Saariluoma, 2004, 2005a,b). Even apparently unrelated areas, such as culture (Calhoun, Teng, & Cheon, 2002), personality (Caprara, Barbaranelli, & Guido, 2001), or family communication (Latvala, 2006; Parkkola, 2006a; Parkkola et al., in press; Pulkkinen, 2000), should not be overlooked.

Today, the most advanced user understanding has been reached in the field of cognitive processes. Perception, attention, memory, as well as motor process, have been extensively studied for the last 25 years (e.g., Baddeley, 1997; Styles, 2005). That is why this knowledge can be extensively applied in interaction research (see e.g., Helander et al., 1997). However, we still know relatively little about some very important cognitive structures in use contexts, such as models and thinking (see, however, Payne, 2003).

Cognitive psychology has been under intensive research, but many other fields are largely neglected. Emotions, for instance, are the internal system that defines the importance of a matter to a person and plays a vital role in issues such as the pleasantness of use, acceptance, and purchasing decisions (Norman, 2004; Power & Dalglish, 1997). Paradoxically, while it is very common to speak about user needs in folk psychology, what is actually presented often has little to do with human needs in a psychological sense. It would be more accurate, then, to analyze user motives when investigating, for example, why some people use a specific technology and others do not. This knowledge would elaborate important questions such as how people attach themselves to brands and products, and how to segment user groups.

The concept of group is also important in user psychological thinking. Designers may address how to make mobile social software and other group tools efficiently, based on what is known about group processes or how various cultures differ in adopting ICT. A typical example of a group-based field would be family communication services, which are under development and design (e.g., Latvala, 2006; Parkkola, 2006a; Pulkkinen, 2000).

User psychology can thus be organized around different “language games,” depending on the nature of the design problem. Problems are bound to

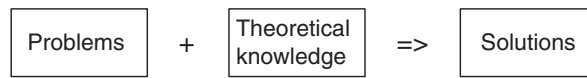


Fig. 1 The explanatory framework has three major components: the usability problem, relevant theoretical knowledge, and the derived interaction solutions

respective fields of psychology. The combination of a problem, a respective psychological theory basis, and a solution model can be called the *explanatory framework* (Fig. 1). One of the leading ideas in user psychology is explanatory design, and this is based on explanatory frameworks (Saariluoma, 2004).

User psychology has its specific perspective on human–technology interaction. It focuses on users, and on the psychological preconditions of their behaviors and actions. This means that user psychology may question the role of some psychological construct in interaction, rather than studying the uses of the devices. Typical examples of user psychology are the analysis of the role of long-term working memory during interruptions (Oulasvirta & Saariluoma, 2004, 2006) or the investigation of visual information chunking in spreadsheet interaction (Saariluoma & Sajaniemi 1989, 1991, 1994). In these examples, the focus is not on any specific device but on the possible roles of psychological mechanisms in the person’s interaction with a device.

5 Ontology as a Tool

Though explanatory frameworks provide a rational basis for solving user psychological problems in design, an important gap in linking it with technological development still exists. This reflects the difficulty in mediating psychological knowledge to technical designers. Psychologists seldom are specialists in information systems and designers are normally not deeply involved with the theories of modern psychology and the empirical evidence backing them. The differences between technical and human research traditions have commonly made it difficult to mediate knowledge from one group to another (Snow, 1959). This serious gap in the shared knowledge, concepts systems, and vocabulary of psychologists and designers, and the difficulty in finding common ground for communication, may explain why the application of psychological analysis has been so rare in practical design (cf. Carroll, 1997).

One way of solving the communication difficulties could be the effective use of ontologies in design. Ontologies traditionally have been used to communicate between organizational groups. Design ontologies are information systems referring to objects, actions, and events that are relevant in guiding and controlling the design process. Ontologies can be defined many ways, but the above definition seems to capture much of the essence of these constructs (Borst, Akkermans, & Top, 1997). The main property of ontologies is that they describe the contents of their references (Chandrasekaran, Josephson, &

Bejamins, 1999; see also Leppänen, 2005). Ontologies provide a means for systematizing and transferring organizational information, as well as the tools for collaborative design and the use of everyday knowledge in organizations (Grueninger, Atefi, & Fox, 2000; Leppänen, 2005). This suggests that ontologies might provide a practical tool for mediating knowledge between human-driven and technical design processes.

In order to enable the effective use of ontologies for mediating knowledge from user psychological work to practical design, it is essential to find out what determines the most essential attributes of such ontologies. There are numerous alternatives available with varying conceptual grounds. Chandrasekaran et al. (1999), for example, present four variant ontologies that are based on the notion of *thing*. Two elementary concepts, living and nonliving, can be used. Sowa (1984) speaks of concrete, process, object, and abstract. Borst et al. (1997) present an analysis of mechanism, in which they first give three major attributes: connectivity, effort, and domain. Subsequently, they give different values to these attributes.

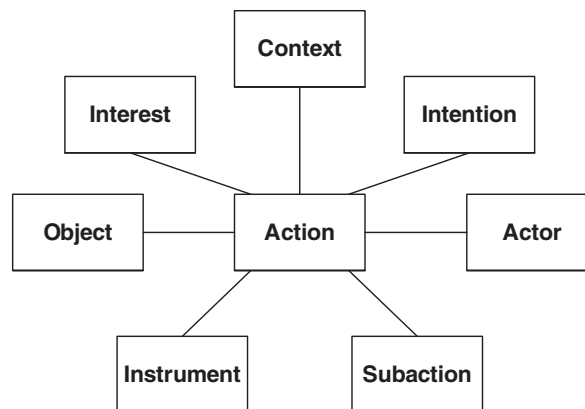
The presented variants are only examples (see Leppänen, 2005, for an extensive overview of existing solutions). The crucial problem is to find effective ground concepts. Chandrasekaran et al. (1999) provide, for example, a good ontology for engineering purposes but, in interaction situations, human dimensions are more important than in engineering design. Therefore, it is necessary to look for a basic concept that is essentially human. Because people use technology to support their actions, action would be a logical basic concept for technology design.

6 Action-Oriented Thinking in Design

In order to understand what designing for human actions means, it is necessary to have a general idea about the structure of human action. This means defining the ontology of action and the attributes in that ontology. In practical design, such a structural analysis of action helps in designing the requirements for the technology under development. A general action ontology may help in defining both what is similar in actions and what is different between them. This is essential for distinguishing between separate actions and in guiding the technological design. There are numerous ways of building ontologies for action. This particular one has been inspired by Parsons (1949), but does not follow his thinking in detail. The main ontological attributes used here are: intention, interest, object, instrument, actor, context, and subaction (Fig. 2).

All actions have goals, or, when mentally represented, the goal is expressed as *intention* and *interest* (Brentano, 1874/1973). Actions are always carried out for some reason. We sing to achieve personal satisfaction, we go to the shop to buy food, we call our parents for support or information, and we fish for fun or as a profession. In any case, what we do is defined by the future expectations

Fig. 2 Action ontology.
This ontology expresses the main attributes of action. They may be all or partly used in analyzing the highest level of actions



underlying our action. A category akin to intention is interest, which defines emotional and motivational aspects of intention. In the end, intentions and interest define why people do what they do.

The analysis of these two ontological attributes of action must be psychologically reliable. Psychological reliability means that it is not sufficient to define only the cognitive goal, but the needs and motives associated with intentions must be described as well (Heckhausen, 1991). A psychological analysis of what the intentions and interest are in a particular case and an explanation why people have the intentions and interest they have must be provided.

Intention thus defines the rational goal of people. Without knowledge about users' goals, researchers are unable to define what users do. However, human actions also have emotional dimensions. Emotions are not necessarily as articulable as the cognitive aspects of human behavior but they are more influential with respect to the lines of action actually chosen. We direct ourselves to pleasurable goals. Of course, these goals may be distant and the path toward them may be full of difficulties, but emotional aspects of the goals remain highly important when explaining what people do (Abele-Brehm & Gendolla, 2000; Lazarus & Lazarus, 1994; Power & Dagleish, 1997). Understanding the emotional nature of an action is essential for describing the motivation behind the intention. Therefore, interests must be analyzed, and not only the intentions.

Actions are targeted at some *object*. It may be a piece of knowledge (e.g., the time when mother comes home) or something more concrete, such as a product or service (e.g., feeding the baby or having clean clothes). In analyzing an action, it is essential to understand the object of that action because the object is crucially important in forming intentions and interests, as well as in understanding the ways people attempt to reach it. Furthermore, mental or spiritual objects are often pursued in a different manner than physical objects. In any case, analyzing the object of an action is a necessary condition for fully understanding an action.

As mentioned earlier, people use technologies as an *instrument* to support their actions and achieve their goals. Technologies are among the physical

instruments. However, an instrument can also be some mental means to reach a goal. Mathematics and language are good examples of mental rather than physical tools. Of course, knowledge and information systems are very important instruments people use in completing their actions.

The next step is to clearly define the actual *actors* or agents. In user contexts, this means understanding the properties of potential users. Thus, in the action-oriented approach, this means defining who the actors are and what their psychological characteristics are. It will be a different task to design communication actions for children, parents, or grandparents.

If the actors are elderly, or from different cultures, the defining should include geropsychological or cultural knowledge in the description of the actors (Charness & Jastrzembski, in this volume; Craik & Salthouse, 2000; Matsumoto, 2000). If they are experts, it is essential to use the knowledge of the psychological characteristics of expertise to describe these actors (Ericsson, 2006; Ericsson & Lehman, 1996). People of different ages also have varying cognitive and physical capabilities, and that must be defined in an action ontology.

Actions always take place in some *context*, with psychological, social, and physical contexts the most significant. The social context involves other people and the way they are organized within the particular action. The social context may be formed by one's friends and family but it also may be some formal organization with precisely defined norms and rules of acting. A valid description of action presupposes an accurate description of the social context. In the family context, for example, it is necessary to consider who the actors of the family are, what kinds of relationship they have, and what other people or institutions influence daily family life.

The physical environment is equally important. Many of the physical contexts may be random, such as the place a mobile phone call is made, while others may be task specific, like the place for laundering. It is important to make models of the technical environment. In practice, these models are often prototypical technologies or interaction models. However, if an action is being designed that is not yet bound to a technology, technical attributes are not needed in the description of the environment. It may be that we are interested in designing a communication service for parents and 6-year-old children who move about in the neighborhood. We then need to have an understanding of the needs of both the parents and the children regarding possible weather conditions or activities in the playground area, but any idea about the technical capabilities of the services are not yet required.

The next attribute of human action that must be included in a cognitive scientifically and psychologically reliable description of action is a description of the *subactions* or action units (Pink & Stone, 2004). To call her child, a mother needs to define to whom she wants to be connected, open the connection, relate the message, and close the connection. This rather simple interaction requires considerable user knowledge of the subactions before attaining a sufficient understanding of the action as a whole. For example, it is necessary

to consider how people designate various recipients of an action and how many potential recipients there might be. The analysis of the subactions is often difficult because many subactions have subconscious elements. None of us knows, for example, how many action recipients there might be in our life and how to differentiate between them if they have the same kind of identifiers used during connections.

The essential aspect of the current approach is that the attributes are explicated in scientific psychological terms. Instead of folk psychologically formulating overall action scenarios, it is essential to explicate and analyze each component of action as scientific psychological concepts and investigate them with psychological methods. This feature separates scientific analysis of actions from traditional intuitive and modern psychological design.

7 From Psychological Analysis to Action-Oriented Design Process

User modeling cannot be an art for the sake of art: It is done to improve design practices. In action-oriented design, an analysis of action is used to design new technologies or redesign actions. Therefore, a designer must define, in a new way, the attributes of the action under design. The list of action attributes posited in Fig. 2 provides a guide for this kind of design activity. However, many additional action attributes have not been discussed here.

Action-oriented design is a constructive activity that is, in some sense, comparable to machine design, and thus the designer is actually a constructor. An action-oriented designer may spend time eliminating unnecessary features in actions, thereby simplifying them. However, the most obvious difference between action design and traditional machine design is in their scientific bases. While machine design relies on the laws of nature, action-oriented design must rely on areas of human research such as cognitive science, psychology or sociology.

When thinking carefully about traditional simulative psychology and its models of the human mind, it is obvious the authors had some engineering ideals in their minds. Card et al. (1983) developed a number of important cognitive concepts that described users' minds in mathematical terms. This was an important kernel of interest. It may indeed be possible to reach stricter design practices to facilitate some elementary means of predicting human behavior in interaction situations. Models provide a strong sign of the reachability of this goal. The main challenge is undoubtedly that the cognitive language of psychology does not have expressive power sufficient for many of the vital issues of humans, such as emotions, personality, and group behaviors.

Saariluoma (2005a,b) called interaction design that relies on the laws and principles of psychology, or any other problem-relevant science, explanatory or argumentative design. This means that design solutions are based on scientific knowledge instead of the intuitive experiences of designers. To obtain such

knowledge for designing actions, one needs to rely on user psychological knowledge and research. The crucial difference between intuitive design and explanatory design is easy to understand: It is hidden in the way design decisions are made. No scientific knowledge or arguments are used in intuitive design.

Today, we are far from the ideal in scientific design through which we could predict how people carry out actions. This is why we must often rely only on testing. Such a predictive and explanatory way of designing, which is the practice in machine engineering, cannot be found currently in designing actions, even though there is no logical obstacle to this. The development of new design practice depends on the level of user psychological understanding within ICT-action environments, and the efficiency of the design activity is dependent on the way user psychological knowledge can be associated with the action analysis and construction.

In the movement toward action-oriented design, it should be no surprise that some level of intuition remains, since highly sophisticated design environments such as machine engineering have their intuitive residuals. Nevertheless, no significant obstacle prevents researchers from using design science effectively.

A brief practical example from the academic world might clarify here. Some years ago, one of the authors had to design a multidisciplinary e-learning network, named Connet, which involved seven universities. Of course, several action design problems exist in such a network. One of the most challenging was to organize teaching in such a way so that students with quite distinct academic backgrounds, from philosophy to engineering and programming, could equally participate in the network. At this stage, the major problem concerned how to organize teaching, not which technology should be used. The solution was generated through problem-based learning because each of the students could, in this manner, contribute to the common discussion about products and other relevant things. Since problem-based learning has a solid theoretical grounding, one might think that the design decision was explanatory. However, the technology for keeping contact and communicating between and among the universities and groups, located geographically several hundred kilometers from each another, was solved later, long after the needs of the people and the groups' issues were explored. This example illustrates in a simple form what action-oriented design means. It means recognizing the actions and their attributes prior to designing the tools (e.g., ICTs) for the environment. Action-oriented design can be an activity completely independent of the technical design; it is not necessary to involve any technology in designing actions.

The actual design process is also organized around the action models. This means that the designers first define the attributes of action needed for describing it. At the same time, they should look for psychological knowledge that can provide a scientific backbone for the designed action, thus making it scientifically grounded and explanatorily argued. In the end, this practice associated with intensive user psychological research should lead to less intuitive action design in ICT interaction design. Such processes are necessary, considering the increasing speed at which potential new technologies are created.

The next step in using the concept of ontologies in technology design should be the comparison of different theories of ontologies and testing their suitability for technology design. In addition, more practical instructions for the use of ontologies in design are needed.

References

- Abele-Brehm, A. E., & Gendolla, G. H. E. (2000). Motivation and emotion. In J. H. Otto, H. A. Euler, & H. Mandl (Eds.), *Emotionspsychologie* (pp. 297–305). Weinheim, Switzerland: Beltz.
- Adelson, B. (1981). Problem-solving and the development of abstract categories in programming languages. *Memory and Cognition*, 9, 422–433.
- Anderson, J. R., Farrell, R., & Sauers, R. (1984). Learning to program LISP. *Cognitive Science*, 8, 87–129.
- Anderson, J. R., & Jeffries, R. (1985). Novice LISP errors: Undetected losses of information from working memory. *Human-Computer Interaction*, 1, 107–131.
- Anderson, J. R., Matessa, M., & Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12, 439–462.
- Baddeley, A. D. (1997). *Human memory*. Hove, UK: Psychology Press.
- Beyer, H., & Holtzblatt, K. (1998). *Contextual design: A customer-centered approach to system design*. San Diego, CA: Academic Press.
- Boehm, B. (1988). A spiral model of software development and enhancement. *IEEE Computer*, 21, 61–72.
- Boring, E. (1950). *A history of experimental psychology*. New York: Appleton-Century-Crofts.
- Borst, P., Akkermans, H., & Top, J. (1997). Engineering ontologies. *International Journal of Human-Computer Studies*, 46, 365–406.
- Brentano, F. (1973). *Psychologie vom empirischen Standpunkt* [Psychology from an empirical point of view]. Hamburg, Germany: Felix Meiner. (Original work published in 1874)
- Calhoun, K. J., Teng, J. T., & Cheon, M. (2002). Impact of national culture on information technology usage behaviour: An exploratory study of decision making in Korea and the USA. *Behaviour & Information Technology*, 21, 293–302.
- Caprara, G. V., Barbaranelli, C., & Guido, G. (2001). Brand personality: How to make the metaphor fit? *Journal of Economic Psychology*, 3, 377–395.
- Card, S., Moran, T., & Newell, A. (1983). *The psychology of human computer interaction*. Hillsdale, NJ: Erlbaum.
- Carroll, J. (1997). Human computer interaction: Psychology as science of design. *Annual Review of Psychology*, 48, 61–83.
- Carroll, J. M., Thomas, J. C., & Malhotra, A. (1980). Presentation and representation in design problem solving. *British Journal of Psychology*, 71, 143–153.
- Chandrasekaran, B., Josephson, J., & Bejamins, V. R. (1999). What are ontologies, and why do we need them? *IEEE Intelligent Systems*, 14, 20–26
- Craick, F., & Salthouse, T. (2000). *The handbook of aging and cognition*. London: Erlbaum.
- Dahl, O. J., Dijkstra, E. W., & Hoare, C. A. (1972). *Structured programming*. New York: Academic Press.
- Ericsson, K. A. (2006). *The Cambridge handbook of expertise and expert performance*. Cambridge, UK: University Press.
- Ericsson, K. A., & Lehman, A. (1996). Experts' performance. *Annual Review of Psychology*, 47, 273–305.
- Floyd, C. (1984). A systematic look at prototyping. In R. Budde, L. Kuhlenkam, L. Mathiassen, & H. Zullighoven (Eds.), *Approaches to prototyping* (pp. 1–18). Berlin, Germany: Springer-Verlag.

- Grueninger, M., Atefi K., & Fox, M. S. (2000). Ontologies to support process integration in enterprise engineering. *Computational and Mathematical Organization Theory*, 6(4), 381–394.
- Heckhausen, H. (1991). *Motivation and action*. Berlin, Germany: Springer.
- Helander, M., Landauer, T., & Prabhu, P. (Eds.). (1997). *Handbook of human-computer interaction*. Amsterdam: North-Holland.
- Hempel, C. G. (1965). *Aspects of scientific explanation and other essays in the philosophy of science*. New York: Free Press.
- Hoves, A., & Young, R. (1997). The role of cognitive architecture in modelling the user: Soar's learning mechanisms. *Human-Computer Interaction*, 4, 311–343.
- Jarke, M. (1999). Scenarios for modeling. *Communications of the ACM*, 42, 47–48.
- Kämäräinen, A., & Saariluoma, P. (2007). Under-use of mobile services: How advertising space is used. In V. Evers, C. Sturm, M. A. Moreno Rocha, E. C. Martínez, & T. Mandl (Eds.), *Designing for Global Markets 8* [Proceedings of the Eighth International Workshop on Internationalization of Products and Systems; pp. 19–29]. Rochester, NY: Product & Systems Internationalization, Inc.
- Kieras, D., & Meyer, D. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, 12, 391–438.
- Knittle, D., Ruth, S., & Patton Gardner, E. (1986). Establishing user-centered criteria for information systems: A software ergonomics perspective. *Information & Management*, 11, 163–172.
- Latvala, J.-M. (2006). *Digitaalisen kommunikaatiosovelluksen kehittäminen kodin ja koulun vuorovaikutuksen edistämiseksi* [Development of a digital communication system to facilitate interaction between home and school]. Doctoral dissertation [Jyväskylä Studies in Education, Psychology and Social Research, No. 292]. Jyväskylä, Finland: University of Jyväskylä.
- Lazarus, R. S., & Lazarus, B. N. (1994). *Passion & reason: Making sense of our emotions*. Oxford, UK: Oxford University Press.
- Leppänen, M. (2005). *An ontological framework and a methodical skeleton for method engineering: A contextual approach*. Jyväskylä, Finland: University of Jyväskylä Press.
- Matsumoto, D. (2000). *Cultural psychology*. Stanford, CA: Wadsworth.
- Mayhew, D. (1999). *The usability engineering lifecycle: A practitioner's handbook for user interface design*. San Francisco: Morgan Kaufmann Publishers.
- McKeithen, K. B., Reitman, J. S., Reuter, H. H., & Hirtle, S. (1981). Knowledge organization and skill differences in computer programmers. *Cognitive Psychology*, 13, 307–325.
- Moran, T. (1981). An applied psychology of the user. *Computing Surveys*, 13, 1–11.
- Nielsen, J. (1993). *Usability engineering*. Boston: Academic Press.
- Norman, D. (2004). *Emotional design: Why we love (or hate) everyday things*. New York: Basic Books.
- Olson, G. M., & Olson, J. S. (2003). Human-computer interaction: Psychological aspects of the human use of computing. *Annual Review of Psychology*, 54, 491–516.
- Oulasvirta, A. (2006). *Studies of working memory in interrupted human-computer interaction*. Doctoral dissertation (Department of Psychology Rep. No. 38). University of Helsinki, Finland.
- Oulasvirta, A., & Saariluoma, P. (2004). Long-term working memory and interrupting messages in human-computer interaction. *Behaviour & Information Technology*, 23, 53–64.
- Oulasvirta, A., & Saariluoma, P. (2006). Surviving task interruptions: Investigating the implications of long term working memory theory. *International Journal of Human Computer Studies*, 64, 53–64.
- Pahl, G., Beitz, W., Feldhusen, J., & Groete, K. (2005). *Konstruktionslehre* [Engineering design]. Berlin, Germany: Springer.

- Parkkola, H. (2003, November). *Observations about the use of technologies in family communication*. Paper presented at the Social and Cultural Dimensions of Technological Development Symposium, Jyväskylä, Finland. Available at <http://www.cc.jyu.fi/~hanpark/publications/Obsuse.pdf>
- Parkkola, H. (2006a). *Designing ICT for mothers*. Unpublished doctoral dissertation, University of Jyväskylä, Finland.
- Parkkola, H. (2006b). What do mothers demand from information and communication technologies? In J. Multisilta & H. Haaparanta (Eds.), *Proceedings of the Workshop on Human Centered Technology* (Porin Yksikkö, Julkaisu 6, pp. 143–151). Pori, Finland: Tampereen teknillinen yliopisto.
- Parkkola, H., Saariluoma, P., & Berki, E. (in press). Action oriented classification of families' information and communication actions: Exploring mothers' viewpoints. *Behaviour & Information Technology*.
- Parsons, T. (1949). *The structure of social action*. New York: Free Press.
- Payne, S. (2003). Mental models: The very ideas. In J. Carroll (Ed.), *Models, theories and frameworks* (pp. 135–154). San Francisco: Morgan Kaufman.
- Pennington, N. (1987). Stimulus structures and mental representations in expert comprehension of computer programs. *Cognitive Psychology*, 19, 295–341.
- Pink, T., & Stone, M. (2004). *The will and human action*. London: Routledge.
- Posner, M. (1980). Orienting of attention. *Quarterly Journal of Psychology*, 32, 3–25.
- Power, M., & Dalgleish, T. (1997). *Cognition and emotion: From order to disorder*. Hove, UK: Psychology Press.
- Pulkkinen, L. (2000, May). *Life-span perspective on human-centered technology*. Presentation made at the Thematic Seminar and Demonstration on Human Centered Technology organized by the Jyväskylä Region, Finland, Brussels, Belgium.
- Rosson, M. B., & Carroll, J. M. (2002). *Usability engineering: Scenario-based development of human-computer interaction*. San Francisco: Morgan-Kaufmann.
- Royce, W. W. (1987). Managing the development of large software systems: Concepts and techniques. In *Proceedings of the 9th International Conference on Software Engineering* (pp. 328–338). Los Alamitos, CA: IEEE Computer Society Press.
- Saariluoma, P. (1997). *Foundational analysis: Presuppositions in experimental psychology*. London: Routledge.
- Saariluoma, P. (2004). *Käyttäjäpsykologia* [User psychology]. Porvoo, Finland: WSOY.
- Saariluoma, P. (2005a). Explanatory frameworks for interaction design. In A. Pirhonen, H. Isomäki, C. Roast, & P. Saariluoma (Eds.), *Future interaction design* (pp. 67–82). London, UK: Springer-Verlag.
- Saariluoma, P. (2005b). Mitä on käyttäjäpsykologia? [What is user psychology?]. *Psykologia*, 40, 181–186.
- Saariluoma, P., & Sajaniemi, J. (1989). Visual information chunking in spreadsheet calculation. *International Journal of Man-Machine Studies*, 30(5), 475–488.
- Saariluoma, P., & Sajaniemi, J. (1991). Extracting implicit tree structures in spreadsheet calculation. *Ergonomics*, 34, 1027–1046.
- Saariluoma, P., & Sajaniemi, J. (1994). Transforming verbal descriptions into mathematical formulas in spreadsheet calculation. *International Journal of Human-Computer Studies*, 41, 915–948.
- Shneiderman, B. (1976). Exploratory experiments in programmer behavior. *International Journal of Computer and Information Sciences*, 5, 123–143.
- Snow, C. P. (1959). *Two cultures and the scientific revolution*. Cambridge, UK: Cambridge University Press.
- Sowa, J. (1984). *Conceptual structures*. Boston: Addison-Wesley.
- Stich, S. (1983). *From folk psychology to cognitive science: A case against belief*. Cambridge, MA: MIT Press.
- Styles, E. (2005). *Attention, perception and memory*. Hove, UK: Psychology Press.

- Tollmar, K., & Persson, J. (2002). Understanding remote presence. In *Proceedings of the Second Nordic Conference on Human-Computer Interaction* (pp. 41–50). New York: ACM Press.
- Torgny, O. (1998). Future home environments and media forms. Centre for User Oriented IT Design, Royal Institute of Technology (Report TRITA-NA-D9808, CID-35). Stockholm, Sweden: CID. Retrieved October 12, 2006, from http://cid.nada.kth.se/pdf/cid_35.pdf
- Whiteside, J., Bennett, J., & Holtzblatt, K. (1988). Usability engineering: Our experience and evolution. In M. Helander (Ed.), *Handbook of human computer interaction* (pp. 791–817). Amsterdam: Elsevier.

III

ONTOLOGIES FOR HUMAN-TECHNOLOGY INTERACTION DESIGN

by

Saariluoma, P., Lamminen, J., & Leppänen, M. 2009

In Proceedings of the 16th International Product Development Management
Conference, Twente, The Netherlands: European institute for advanced studies
in management, EIASM

Reproduced with kind permission by EIASM.

ONTOLOGIES FOR HUMAN-TECHNOLOGY INTERACTION DESIGN

Pertti Saariluoma¹

Juha Lamminen²

Mauri Leppänen³

^{1,3}Department of Computer Science and Information Systems

²Agora Center

University of Jyväskylä

P.O.Box 35, FI-40014 University of Jyväskylä

psa@jyu.fi, juha.e.lamminen@jyu.fi, mauri@cs.jyu.fi

ABSTRACT

Practical design process in human-technology interaction presupposes knowledge management. It is essential to store earlier experiences and transmit information in industrial design. In this work, ontologies may be efficient tools. Here, we are interested in constructing an ontology for describing human actions for user psychological design. By user psychology we refer to psychological analysis of interaction. It means that we apply methods, concepts and theories of modern psychology to solve interaction design problems. Unlike in psychological usability research, where the interest is focused on the usage of existing technologies, we are interested in designing psychologically motivated interaction activities. Thus, user psychological interaction ontology refers to an ontology that is intended to guide the storage of a collection of psychological design knowledge. A good action ontology can guide psychologically-centered user interaction design processes. Here, we have collected a set of intuitive descriptions of real-life interaction processes and used that data to abstract a general conceptual ontology for describing the highest level human actions during an interaction process.

INTRODUCTION

Human-technology interaction (HCI) is on its way to become a central scientific and technological challenge for developing our modern ICT-society. The development of hardware technology and advancement of services closely related to it make it necessary to improve the level of human-technology interaction. Agent-based, ubiquitous, pervasive, and embedded systems may provide us with important new ways to improve human life, but it is not a foregone conclusion that people can benefit from them the way intended as the use of these systems may be too demanding for many.

The comparison between SMS and WAP provide us with a typical example of the problems, successes and closed opportunities. While SMS has become one of the biggest successes in mobile services, WAP, despite its sound basic idea as a mobile counterpart to the desk-based Internet, has simply disappeared. If interaction problems are not understood deeply, this often leads to situations in which people do not use services, though they could, without doubt, benefit from them (Kämäräinen and Saariluoma, 2007). This means that there is a serious risk of a negative dependence between invention and innovation. When new technical systems are not accepted by the larger audience, they cannot be developed due to drying up of financing sources, though the basic idea behind them might be very sound. The business logic is very simple here: companies will have more money for development if more users adopt a new service quickly. This is

why human-technology interaction has become such a central problem for science as well as for different levels of business.

WAP and many other less expensive unsuccessful developments demonstrate that we have to essentially improve the scientific basis of interaction analysis and the practices we use in it. Designers of the future must find new thought models to effectively use the familiar practices. The traditional forms of HCI thinking must be improved with new types of information and expertise (Carroll, 1997; Helander et al., 1997; Olson and Olson, 2003; Rosson and Carroll, 2002). Both the ways of designing and collecting information from the users and interaction processes require constant improvement. One possibility to improve interaction design practices is offered by interaction ontologies (Saariluoma et al., 2008).

By a human-technology interaction ontology we mean a system of higher level concepts describing human-technology interaction. Of course, ideally any ontology would collect and organize the highest level concepts so that it would have maximal generality. It could thus be applied to guide any human-technology interaction design process.

The advantages of human-technology ontologies can be assessed from several different directions. Firstly, they provide information about the kind of knowledge that concerns interaction and that should be investigated in design. Secondly, they allow knowledge management during the design process. This means that the designers would know about what has been done earlier and what is done at the moment in organizations so that any unnecessary work could be avoided. Finally, ontologies are vital in developing our general understanding of interaction processes. They can be used to collect, store and communicate much of what we know about the core of human-technology interaction processes.

The rest of the paper is organized as follows. The next section describes the origin, needs and progress of research on user psychology. After that, we discuss the role of user psychology as an explanatory science and its potential in human-technology interaction design. Next, we derive our first version of human-technology interaction ontology from interaction examples of ten enterprises participating in the Easy-Interactions project. We also discuss ways in which the ontology can be utilized in human-technology interaction analysis and design. The paper ends with conclusions.

PSYCHOLOGICAL FOUNDATIONS OF INTERACTION DESIGN

It is intuitively evident that the analysis of human-technology interaction must be grounded on in-depth knowledge of human research. This means, for example, knowledge about the philosophy of mind, sociology, and psychology (Saariluoma, 2005). The latter, especially, is in a critical position, because the most valuable knowledge of human mind concerning interaction is psychological. Though not underestimating the importance of the other human research fields and their contributions, in this paper we concentrate on psychology.

Applying psychology in interaction design is not uncommon. It is unavoidable. Designing any interaction processes presupposes some knowledge about the user, because it is not possible to design controls, goals or purposes of interaction without any knowledge of users. This knowledge may be implicit or explicit, but it is present in the design process. This means that the application of psychology in interaction design is not a free choice of the designers but a necessity. It is not possible to design a tool without

making any assumptions about its use or users. In practice, this necessity of making assumptions concerning users has led to a wide use of folk psychological understanding in design. Technically schooled designers apply their common sense understanding of human behavior in making their design solutions.

Folk psychology is not necessarily an incorrect basis for designing human-technology interaction. It is intuitive, i.e., there are no argumentative grounds for its claims (Saariluoma, 1997). Of course, intuitions about the user can be correct. Often in the history of science it has been noticed that some literary ideas have proven to be correct. For example, Flaubert described very neatly a failed communication between Emma Bovary and a priest. This was the moment in the life of this fictive character that might have prevented the final suicide. Indeed, much later, Durkheim (1951) proved that communication in the social networks of catholic communities can prevent suicides. The problem with intuitions is rather that we do not know whether those intuitions are correct or not. They are not grounded on systematic research. In the absence of relevant research, it is, of course, necessary to ground one's thinking on intuitions, but it would be much better to have a systematic scientific basis for analyzing human-technology interaction.

Scientific research to the psychology of human-technology interaction has existed for some time. Shneiderman (1976) applied a basic experimental analysis for understanding skills in his investigation on the psychology of programmers. He showed that human working memory is an essential theoretical construct in this research. At the same time, a number of important psychological papers were published on the psychology of computer programming, which essentially improved psychological understanding of users: (Adelson, 1981; Anderson et al., 1984; Anderson and Jeffries, 1985; Carroll et al., 1980; McKeithen et al., 1981; Pennington, 1987). These papers among others opened the field for scientific psychological analysis of computer use. Much of the theoretical psychological knowledge of users' behavior and mentality has been developed under this paradigm.

The first theoretical synthesis was made by Card, Moran and Newell (1983) in the form of the GOMS (Goals, Operations, Methods and Selection rules) architecture to describe computer users in practical contexts such as text editing. Later, this cognitive modeling paradigm was greatly extended and formed one of the major scientific ways of considering the human mentality in use contexts (Anderson et al., 1997; Hoves and Young, 1997; Kieras and Meyer, 1997).

This early research led to technologically oriented and motivated research. The key challenges were those found in usability problems; i.e., addressing the questions of how to find new uses for existing technologies and iteratively test existing solutions and prototypes of new systems (Rosson and Carroll, 2002). The dependency on the development of new technologies makes the interaction research slow in a developing information society: it cannot define the goals for the development of new technologies, as it just keeps following technical advancements. Nevertheless, there is no logical reason for why the scientific interaction research could not direct the development of technologies and define its goals. Consequently, one may begin a design process by considering the needs, properties and behavior of the user rather than by analyzing and testing the possibilities of already existing technologies. This is the issue in which user psychology differs from usability research (Nielsen, 1993; Olson and Olson, 2003; Oulasvirta and Saariluoma, 2004; Rosson and Carroll, 2002; Saariluoma et al., 2008).

User psychology is a term suggested by Moran (1981). Construction of cognitive models is an example of a partial user psychological practice (Card et al., 1983). In these traditions, the psychologist begins by analyzing what users are before any actual application development and design work is begun. Certainly the validity of user psychology is at least on par with the validity of practices in traffic psychology or industrial psychology, both commonly used. The problem is rather what shape that user psychology should have, to be able to provide an optimal contribution to the interaction design.

User psychology, as we see it, investigates the psychological preconditions for use and human-technology interaction (Oulasvirta and Saariluoma, 2004; Oulasvirta and Saariluoma, 2006; Saariluoma, 2004). This means that user psychology must be able to explicate usability problems within psychological concepts and investigate them with psychological methods. The main goal is to replace the traditional intuitive interaction design with scientifically justifiable and reliable methods.

Modern user psychology allows interaction designers to elaborate the analysis of human behavior in several ways. Research outcomes are more reliable because the psychological methods have been well tested through practice, and user psychology allows designers to rely on diverse psychological knowledge bases in searching for usability solutions. Traditional usability engineering uses concepts such as memorability and learnability as criteria for viability of interaction. However, that approach is still quite distinct from using the psychological knowledge we have about human memory to improve these two aspects of interaction (Baddeley, 1997). Learnability and memorability are external measures, not internal principles which would be applied to improve interaction (Nielsen, 1993). More extensive application of user psychological knowledge would enable basing the design concepts on psychological findings and theories.

FROM INTUITIONS TO EXPLANATIONS

Engineering design has one important property, which is often missing in modern interaction design, i.e., explaining. Science is capable of explaining why something happens. It provides scientific answers—not folk psychological or educated guesses—to the “Why?” question (Hempel, 1965; Saariluoma, 1997). Modern user psychology has ambitions to become an explanatory approach. It should thus provide scientific answers to questions, problems and solutions emerging in interaction design. This means that it should be able to state on the ground of tested scientific laws, principles or facts, why some interaction solution is better than another. Take, for example, the question, why pop-ups are so disturbing. User psychological answer would be: because pop-ups cause an orientation reaction and thus they interrupt the main task (Oulasvirta and Saariluoma, 2006; Posner, 1980). Thus, our knowledge about orientation reaction provides a psychologically grounded explanation, and empirical understanding for what happens within users when they come across pop-ups. The idea that user psychological design should be explanatory (Saariluoma, 2005) is not that dramatic, because design, in modern engineering practices, is always explanatory. Engineering design science begins quite straightforwardly with the idea that engineering must be based on the natural sciences and experimental knowledge (Pahl et al., 2005).

The only way of bringing user psychological design to the level of normal engineering design is to move to explanatory design practices and put aside folk psychological design

paradigms. This does not necessarily mean that all intuitive elements should be rejected in interaction design: they have not been eliminated from engineering design practices either. There is always room for creative thinking, but even within that creativity, interaction designers should be capable of basing their explanations about their key solutions on scientific grounds. Their standard design solutions should also rest upon scientific grounds.

One may justly ask what is the psychological knowledge required in explanatory design. As design is holistic, there are not too many pieces of psychological knowledge which would not be relevant in solving some issues (Saariluoma, 2004). The traditional cognitive analyses of attention, memory, learning, and thinking are as important as information about emotions and motives (Norman, 2004; Saariluoma, 2004; Saariluoma, 2005). Even such apparently distant areas as groups and cultures (Calhoun et al., 2002) or personality (Caprara et al., 2001) can not be overlooked.

User psychology can thus be organized around different psychological knowledge types depending on the nature of the actual design problems. The combination of a problem, the respective psychological theory basis, and a solution model may be jointly called the explanatory framework (Figure 1). One of the leading ideas in user psychology is explanatory design, and this is based on explanatory frameworks (Saariluoma, 2004).

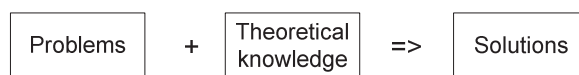


Figure 1. Explanatory framework

User psychology has its specific perspective on human-technology interaction. It focuses on users, and on the psychological preconditions of their behavior and actions. This means that user psychology may question the role of some psychological construct in interaction, rather than studying directly the uses of the devices. Typical examples of user psychology are the analysis of the role of long-term working memory during interruptions (Oulasvirta and Saariluoma, 2004; Oulasvirta and Saariluoma, 2006) the investigation of visual information chunking in spreadsheet interaction (Saariluoma and Sajaniemi, 1989; Saariluoma and Sajaniemi, 1991; Saariluoma and Sajaniemi, 1994) or the functions of mental contents and forms of life (Leikas and Saariluoma, 2008).

In these examples, the focus is not on any specific device but on the possible roles of psychological mechanisms in a person's interaction with a device. However, ontologies are never psychological issues only. They are conceptual structures, which we apply to psychological knowledge. This is why we have to pay attention also to concepts in ontologies and especially the procedures by which we generate the structures.

A HUMAN-TECHNOLOGY INTERACTION ONTOLOGY

In order to organize user psychological knowledge for interaction design, it is important to construct ontologies which make sense psychologically. These ontologies should be able to describe human interaction behavior as well as its internal and external preconditions in detail. This means that we have to have a conceptual structure which can be used to describe user psychological knowledge on a high level of abstraction.

In order to come up with such ontology and to get some realism to it, we collected interaction examples from ten enterprises participating in the Easy-Interactions project. The number of use cases was, at their highest level, over thirty. We extracted the highest level concepts from the resulting documents and organized them into a high level interaction ontology. The idea was that the empirical basis for the ontology would be extracted by means of content-analysis. After that the concepts could be organized into a system of major interaction concepts. Of course, we presupposed that each of the concepts could have a justification both regarding its contents as well as its place in the system.

The system of the concepts of human-technology interaction ontology is outlined in the form of light-weight ontology in Figure 2. An ontology means an explicit specification of a conceptualization of some part of reality (Gruber, 1993). Our ontology is just a tentative conceptualization of highly complex phenomena related to human-technology interaction. For representing the ontology, we have selected a graphical language, and preferred the UML language (Booch et al., 2005) to special ontology representation languages (e.g. CLEO, LINGO, DAML+OIL and OWL) because of its large and rapidly expanding user community, intrinsic mechanism for defining extensions, and largely available computer-support.

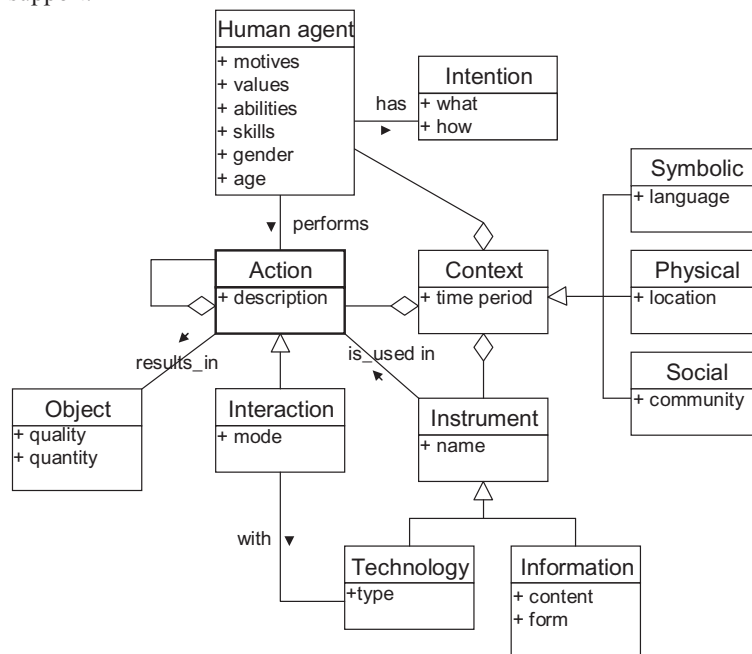


Figure 2. Human-technology interaction ontology

This type of ontology can be characterized as action ontology, because its structure is organized around the concept of action (Saariluoma et. al., 2008). The choice of the focal concept can be justified on the ground that people interact with technologies to reach

their action goals. There is simply no technology which would not be used to aid people in achieving their action goals. Action refers to a behavior with a definite and definable goal. Actions provide us thus with a pragmatic perspective to what people do when they interact with technologies. Understanding actions provides us with explanations why some technology has been constructed and why it has the interaction solutions it has. The action that we aid with a technology explains why this technology makes sense.

Here the concept of technology should be used in a wide sense. This concept can be applied to global issues such as nuclear or aircraft technologies, or to such narrow issues as how to use a camera in invasive surgical operations. Extensive use of technology, to reach sufficient generality, is also an additional reason why interaction ontology should be very general.

Actions, whether small or large, are related to other concepts. Here, we divide the concepts into two categories. The first category contains the concepts which are directly related to action. These are human agent (or user), object, instrument and context. It is not possible to think an action without these concepts. Action and these four concepts have their linguistic counterparts, i.e., verb, noun, object, instrument and context (Parsons, 1949). The concepts in the second category are considered attributes of the concepts of the first category. Next, we will define the concepts and illustrate them with examples extracted from the use cases of the Easy-Interactions project.

Human agent is an essential property of human action, because an agent represents the human being or beings involved in action.¹ There is no action without an acting human being. A mindlessly working machine does not act, it is merely something that happens, a process. The intentions and needs of human mind transform an event into an action. This is why, when we describe action, we need a description of human agent(s).

Instrument is the second necessary property of action in the ontology. We can distinguish between two kinds of instruments, technology-based and information-based. The *technology-based instruments* mean any physical artifacts (e.g., mobile phones, cars, coffee machines, elevators, hammers, knives), which enable a human being to carry out actions or makes the performance more efficient and effective. The *information-based instruments* refer to intellectual artifacts, such as methods, techniques, rules of thumb and databases, which are used to make the execution of actions more straightforward and easier. We focus on the technology-based instruments, because we are specifically interested in action as a human-technology interaction.

Object stands for anything that is changed by the action or should be reached by the action. This means a change in some state of affairs so that the object gets a new form as a consequence of the activity. When a smoke diver, a specialist in the firefighting profession, rescues a man trapped in the burning building, the object is the man and his changed state. If we input a new address to our address file, the object is the new name and respectively the change in address file. Our actions have their objects, which we can understand by analyzing the end states of successful actions.

Context refers to the environment in which the action is carried out. A context can mean a symbolic, physical or social context. A *symbolic context* stands for a linguistic system, composed of icons and commands for example, within and by which the action is carried out. A *physical context* designates an environment composed of tangible objects and

¹ Here, by term agent we always mean a human agent.

states, such as the interface of the phone, sunny beach or frosty forest. By a *social context* we mean a social-communicative network realized, for instance, by means of a service using some commands and presented on the screen with the color and light properties.

Each of the aforementioned concepts has its attributes. Agent has such attributes as intention, which can be divided into what-intentions and how-intentions in the interaction research. Intention defines the goal and the issues relevant to actions. What-intention specifically means what a person wants to do. He may, for example, wish to play a record of Camarón de la Isla or get money (e.g., from a cash machine). How-intention refers to what he has to do with the technology to realize the what-intention. This means, for example having a set of buttons and knowing the order in which one should push them to reach the what-intention. Intention is the property of the user, because it is represented in the mind of the user and is defined in the set of all imaginable actions with the particular technology that is relevant in that particular interaction session. Another example can be given from the domain of fire service. A fireman wants to know if there are people trapped inside the target building. For this purpose, he needs some technology to know the floor plan.

Agent has also other attributes. Some of them are psychological such as age, abilities and use skills. Designing for old people is very different from designing for young people. Many interaction design challenges are posed, for example, by the design of a special mobile user interface for elderly people with impaired hearing or sight. On the other hand, different kinds of design challenges must be faced when designing domestic applications or home entertainment devices for both elderly and young people. For this reason alone we must know what we need to know about the age of the users, because the psychological age-related properties very often provide explanation for interaction decisions. The same is true with abilities and skills. For example, some consumers may have only limited skills for using technical instruments, having limited previous experience about using similar instruments. We can design much more complex interaction processes for specially trained professionals than for consumers. Gender may be relevant in some interaction solutions (Passig and Levin, 2000; Whitley, 1997), and also human values may often explain why people do not accept or like some interaction solutions. Finally, human motives in the end tell whether people will or won't use some technical systems (Friedman, 1996).

Interaction is a complex process to describe, as it has numerous different parts. Using a camera phone can entail, on the highest level, very general actions such as providing information about one's flat to some friends or about a fault to a maintenance service. Nevertheless, we need to define also numerous sub-actions such as moving to camera mode, defining the target of the picture, making adjustments, shooting the picture, moving the picture to the proper gallery or sending it to the intended receiver. This means that an action can be part of a (hierarchical) system of *sub-actions*.

The notion of action can in this kind of system be used repetitively to describe sub-actions. The content of the actions change but many of the features can be inherited from one level to another. The general properties of the user, for example, remain practically unaltered in our mobile camera example. The user is what we have imagined the user to be (e.g., a professional housing agent or a diesel engine engineer). However, each new level brings about new properties. It may be that the engineer does not know how to send a picture from a gallery to the receiver though he knows how to send the picture

immediately after taking it. Thus we have to alter the skills component with respect to this node.

It is also possible that some attributes do not make sense in carrying out some sub-actions. For example, gender is no more an important variable when we investigate how people target the object with a camera phone, though it may be an important factor in designing the physical phone. The size of the buttons is related to the size of the hands used in handling them. Sub-actions themselves must naturally be redefined each time, because they may all be different. Moving cursor is different from taking a picture. In each sub-action point it is essential to define what new this point brings to the analysis of the interaction process. In this way, we can represent a large action by means of a relatively compact ontology.

Of course, we present here only the highest level and the basic idea of a real psychologically motivated user ontology. Industrially, real interaction ontologies are far too large to be presented or discussed in as narrow a space as this single paper. This is also true with most of present day information systems. Scientific analysis and real life information systems do not meet each other but on a principal level.

USER PSYCHOLOGICAL KNOWLEDGE AND INTERACTION ONTOLOGIES

Interaction ontologies are important tools for user psychology, because most of the action and user properties can and must be defined on the ground of psychological knowledge. This means that the attributes of real life ontology must be filled with psychologically grounded information. We have to have good psychologically or in other ways grounded foundations for the ontological structures.

Any interaction process can be expressed in the form of an action hierarchy. In generating this kind of hierarchy, in each action/sub-action node a number of psychologically motivated and psychologically solvable problems can rear their heads. We need to define what agent (user) properties are important in a sub-action node. We need to solve what the relevant issues are concerning instrument, context, and object. We need to find rational and psychologically groundable solutions for these problems.

Let us have an example. In constructing ABS-brakes, their manufacture had left a very strong vibration on the gas pedal. Originally, the brakes had been designed to prevent some steering accidents in winter road surfaces, but in practice the results were not what was expected, because the vibration disturbed the drivers. It caused an orientation reaction and consequently many drivers simply stopped braking. These kinds of problems should become more obvious with the employment of ontological analysis.

The focus is here nevertheless on that the behavior of an instrument must be specified and its psychological consequences should be evaluated. In many cases such as the above, the psychological analysis can be very straightforward, as it is possible to connect interaction situations with well known psychological principles. The design rational is very clear in the above case: when interaction entails orientation activities which are irrelevant to the main task, it is essential to modify the interaction and eliminate the unnecessary disorientation.

Orientation can be a problem in many other cases. People may have too many tasks to take care of. Pop-up effects have already been mentioned. Unnecessarily, disturbing information flow causes many errors and this is why it is essential to analyze the

ontological structure of interaction from that point of view. This type of analysis should really become a routine.

The main issue here is to understand that there is a connection between psychological knowledge and the ontological attributes in a system of concepts which is constructed in the way that makes psychological sense. This is precisely the reason why action ontologies make sense. They enable us to connect psychology of the user with the actual interaction process. This also means that we can think of interaction design as designing human actions.

In practice we have to test the true validity of the concepts and attributes on the grounds of scientific psychological and design testing practices. Firstly, the general ontology described above must be transformed into a task specific ontology. It is one thing to have a general action ontology and another to have a specific instance of it. We can apply general schemas for describing particular services, for example. Let's assume that we have to design a service for senior citizens with impaired vision which would help them in their daily shopping and errands. This means that we have to investigate how, in this kind of action, the psychological preconditions for interaction function, assuming the mentioned type of users.

The use of empirical analysis is only one phase in testing interaction design attributes. We need such testing to check the validity of the given values. Let's assume that we think that voice web-browsing would be a natural way of organizing perceptually impaired senior citizens' shopping and errands. In this service, senior citizens with impaired vision would control the computer by means of voice. Empirical analysis might illustrate us that voice interaction can be very problematic for designers in this kind of case (Hanson et al., 2007). This would mean that our original model would require modification, but it would not indicate that the attribute itself would be poorly figured out. The failure of the test rather suggests that the attribute is making sense.

The testing of the relevance of concepts and attributes should thus be based on whether the required information is essential for the design or not. In this way, testing the relevance of concepts and attributes would depend on whether answering to the questions they imply makes sense in the design process or not. If we would never need what-intention in the design process and it would never generate important design questions, it would make sense to eliminate it from the ontology, but if it has called attention to sense making issues, it is a relevant issue for the design process.

CONCLUSIONS

We have here outlined an ontologically founded way of thinking about human-technology interaction and its design from a user psychological perspective. The core assumption has been that it is possible, by means of action ontologies, to improve the use of scientific psychological knowledge in interaction design. Because the key concept in this design thinking is action, we may call it action-oriented design (Saariluoma, 2004; Saariluoma and Miettinen, 1994).

Action-oriented design is comparable to any machine design activity as it represents constructive thinking. In action-oriented design, the focus is on design action and on the use of action ontologies as the basic tool in representing design plans, interaction solutions and necessary explanatory user psychological information. The difference between action oriented design and traditional machine design is in the scientific

information used in grounding the solutions (Pahl et al., 2005; Saariluoma et al., 2008). While machine design relies on the principles of chemistry, physics, material science or nanoscience, action-oriented design is based on the laws of psychology and sociology.

This means that designing action is essentially different from the traditional design in many senses. As a matter of fact the main difference can be found in Brentano's (1874) famous distinction between intentional and material systems. People have goals and their behaviour must be explained on the basis of those goals, while machines behave as they are intended to behave by their designers. Machines are causal systems with functional specifications, but people set themselves their goals. Since goal driven behaviour is action, it is logical to base the analysis of human behaviour in interaction processes on the notion of action.

There are other dimensions of mind aside from mere action which we must take into account on designing action ontologies. It is not enough to understand human intention and goals, but it is also important to have a clear idea why people do not reach the goals they have set themselves. People make errors and are ignorant of many things. They may have what-intentions but be incapable to form how-intentions. The sad fact is that good what-intentions cannot replace bad how-intentions. This is why we must be able to create interaction processes which do not prevent the realization of what-intentions.

A problem with psychological knowledge in modern design is that it easily disappears. It may be concentrated too much on some specific issues and it thus loses its value when the interaction process changes as the consequence of changing materials and design fashions. This is why we have to be able to cumulatively develop our user understanding. Interaction ontologies provide us with a possible tool for this.

Acknowledgements: This work has been supported by Easy-Interactions and Theseus projects.

REFERENCES

Adelson, B. (1981). Problem-solving and the development of abstract categories in programming languages. *Memory and Cognition*, 9, 422–433.

Anderson, J. R., Farrell, R., and Sauers, R. (1984). Learning to program LISP. *Cognitive Science*, 8, 87–129.

Anderson, J. R., and Jeffries, R. (1985). Novice LISP errors: Undetected losses of information from working memory. *Human-Computer Interaction*, 1, 107–131.

Anderson, J. R., Matessa, M., and Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12, 439–462.

Baddeley, A. D. (1997). *Human memory*. Hove, UK: Psychology Press.

Booch, G., Rumbaugh J., and Jacobson, I. (2005). The unified modeling language user guide, 2nd edition, Addison-Wesley.

Brentano, F. (1874/1955). *Psychologie vom empirischen Standpunkt*. Hamburg: Felix Meiner.

Calhoun, K. J., Teng, J. T., and Cheon, M. (2002). Impact of national culture on information technology usage behaviour: An exploratory study of decision making in Korea and the USA. *Behaviour & Information Technology*, 21, 293–302.

Caprara, G. V., Barbaranelli, C., and Guido, G. (2001). Brand personality: How to make the metaphor fit? *Journal of Economic Psychology*, 3, 377–395.

Card, S., Moran, T., and Newell, A. (1983). *The psychology of human computer interaction*. Hillsdale, N. J. Erlbaum.

Carroll, J. (1997). Human computer interaction: Psychology as science of design. *Annual Review of Psychology*, 48, 61–83.

Carroll, J. M., Thomas, J. C., and Malhotra, A. (1980). Presentation and representation in design problem solving. *British Journal of Psychology*, 71, 143–153.

Durkheim, E. (1951). *Suicide*. New York: Free Press.

Friedman, B. (1996). Value-sensitive design. *Interactions* 3 (6): 16-23

Gruber, T. (1993). A translation approach to portable ontology specification. *Knowledge Acquisition*, Vol. 5, No. 2, 119-220.

Hanson, V. L., Richards, J. T., and Lee, C. C. (2007). Web access for older adults: Voice Browsing?. *Proceedings of HCI International.- Universal Access in HCI*. Beijing, China. July 22 – 27, 2007. Springer.

Helander, M., Landauer, T., and Prabhu, P. (Eds.). (1997). *Handbook of human-computer interaction*. Amsterdam: North-Holland.

Hoves, A., and Young, R. (1997). The role of cognitive architecture in modelling the user: Soar's learning mechanisms. *Human-Computer Interaction*, 4, 311–343.

Hempel, C. G. (1965). *Aspects of scientific explanation and other essays in the philosophy of science*. New York: Free Press.

Kieras, D., and Meyer, D. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, 12, 391–438.

Kämäräinen, A., and Saariluoma, P. (2007). Under-use of mobile services: How advertising space is used. In V. Evers, C. Sturm, M. A. Moreno Rocha, E. C. Martínez, T. Mandl (Eds.): *Designing for Global Markets 8, IWIPS 2007*, Merida, Mexico, Proceedings of the Eighth International Workshop on Internationalisation of Products and Systems. Product & Systems Internationalisation, Inc.

Leikas, J. and Saariluoma, P. (2008). Older People's Representations of ICT-Services: Female Cautiousness Effect. Proceedings of Workshop on Cognition and the Web: Information Processing, Comprehension and Learning, 24-26 April 2008, Granada, Spain Granada: University of Granada.

McKeithen, K. B., Reitman, J. S., Reuter, H. H., and Hirtle, S. (1981). Knowledge organization and skill differences in computer programmers. *Cognitive Psychology*, 13, 307–325.

Moran, T. (1981). An applied psychology of the user. *Computing Surveys*. 13, 1–11.

Nielsen, J. (1993). *Usability engineering*. Boston: Academic Press.

Norman, D. (2004). *Emotional design: Why we love (or hate) everyday things*. New York: Basic Books.

Olson, G. M., and Olson, J. S. (2003). Human-computer interaction: Psychological aspects of the human use of computing. *Annual Review of Psychology*, 54, 491–516.

Oulasvirta, A., and Saariluoma, P. (2004). Long-term working memory and interrupting messages in human computer interaction. *Behaviour & Information Technology*, 23, 53–64.

Oulasvirta, A., and Saariluoma, P. (2006). Surviving task interruptions: Investigating the implications of long term working memory theory. *International Journal of Human Computer Studies*. 64, 53-64.

Pahl, G., Beitz, W., Feldhusen, J., and Groete, K. (2005). *Konstruktionslehre [Engineering design]*. Berlin, Germany: Springer.

Parsons, T. (1949). *The structure of social action*. Clencoe, Country: Free Press.

Passig, D. and Levin, H. (2000). Gender preferences for multimedia interfaces. *J. Comput.-Assisted Learning*, vol. 16, pp. 64–71.

Pennington, N. (1987). Stimulus structures and mental representations in expert comprehension of computer programs. *Cognitive Psychology*, 19, 295–341.

Posner, M. (1980). Orienting of attention. *Quarterly Journal of Psychology*, 32, 3–25.

Rosson, M. B., and Carroll, J. M. (2002). Usability engineering: Scenario-based development of human-computer interaction. San Francisco: Morgan-Kaufmann.

Saariluoma, P. (1997). Foundational analysis: Presuppositions in experimental psychology. London: Routledge.

Saariluoma, P. (2004). Käyttäjäpsykologia (User psychology). Porvoo, Finland: WSOY.

Saariluoma P. (2005). Explanatory frameworks for interaction design. In A. Pirhonen, H. Isomäki, C. Roast, & P. Saariluoma (Eds.), Future interaction design (pp. 67–82). London, UK: Springer-Verlag.

Saariluoma, P. and Miettinen, M. (1994). Problems of adaptive action-oriented user interfaces. In: M. D. Brouwer-Janse and T. L. Harrington (Eds.): Human-machine communications for educational system design. NATO-ASI Series, Series F: Computer and systems sciences, Vol 129. Springer: Berlin. (pp. 119-126.)

Saariluoma, P., Parkkola, H., Honkaranta, A., Leppänen, M., and Lamminen, J. (2008). User Psychology in Interaction Design – The Role of Design Ontologies. In: Saariluoma P., Isomäki H. (eds.) Future Interaction Design II, Springer, Berlin.

Saariluoma, P., and Sajaniemi, J. (1989). Visual information chunking in spreadsheet calculation. International Journal of Man-Machine Studies, 30(5), 475–488.

Saariluoma, P., and Sajaniemi, J. (1991). Extracting implicit tree structures in spreadsheet calculation. Ergonomics, 34, 1027–1046.

Saariluoma, P., and Sajaniemi, J. (1994). Transforming verbal descriptions into mathematical formulas in spreadsheet calculation. International Journal of Human-Computer Studies, 41, 915-948.

Shneiderman, B. (1976). Exploratory experiments in programmer behavior. International Journal of Computer and Information Sciences, 5, 123–143.

Whitley, B.E. (1997). Gender Differences in Computer-Related Attitudes and Behavior: A Meta-Analysis Computers in Human Behavior, Vol. 13, No. 1. (January 1997), pp. 1-22.

IV

TOWARDS ACTION-ORIENTED USER INTERFACE DESIGN

by

Lamminen, J. & Saariluoma, P. 2010

In Proceedings of the 2010 International Conference on Management Science and Engineering, October 17-18, Wuhan, China. Hong Kong: ETP Engineering Technology Press, 58-61

Reproduced with kind permission by ETP Engineering Technology Press.

Towards Action-Oriented User Interface Design

Juha Lamminen

*Department Computer Science and Information Systems
University of Jyväskylä
P.O. Box 35, FI-40014 University of Jyväskylä
juha.e.lamminen@jyu.fi*

Pertti Saariluoma

*Department Computer Science and Information Systems
University of Jyväskylä
P.O.Box 35, FI-40014 University of Jyväskylä
psa@jyu.fi*

Abstract - The ultimate goal of user interface design is to develop a product which people can use to reach their goals. By action we normally refer to our attempts to reach certain goals, and this is why it is natural to think user interface design as an activity we engage in when designing tools, instruments or technologies for people to realize their action goals. This paper suggests a novel action-oriented user interface development method that aims to develop a user interface interaction model. This interaction model is based on an organized tree structure composed of node states with relevant usability-related attributes and operations between the nodes.

Index Terms – *action-oriented design, usability, usability attributes, user experience, user interface design*

I. INTRODUCTION

Human-technology interaction (HTI) research intends to connect technology to human life [1]. This problem is vital because the nature of technologies is changing. The connection between human action and the technology employed has been relatively straightforward in traditional single-function technologies, but today this technology is rapidly becoming multifunctional. This means that the hardware is no longer in a direct relation to the actual use: i.e. in multifunctional technologies one single device can have unlimited number of functions.

In traditional technology such as the one used for steam engines, the function, i.e., what people used the technology for, was well defined. An engine was used to move a train from one station to another, and that was all. Today, one single mobile phone can provide thousands of services, and it can be used for very different goals. It can function as a phone, a camera, or a navigator, and have numerous applications for supporting very different human actions.

This means that we must take a new look at the basic problems of human technology interaction design. We have to begin with the function of the technology and ask what kinds of actions the technology is intended to support and define the interaction processes from the perspective of actions [2, 3].

The difference between two different applications in a computer or a mobile phone is not merely technical. The essential difference is functional, that is, they are supporting human actions that differ from each other. Therefore, it is very logical to think that the contents of the relevant human actions can form the basis for interaction design.

However, human actions have many important parameters. The two most important design questions currently are: 1) how can people best use technology (e.g. traditional

usability thinking)? and 2) do people like to use a given technology (e.g. emotional usability or user experience research)? In this paper, we consider how we can combine analysis of actions with the traditional usability thinking and the investigations in user experience.

II. ASPECT OF USABILITY

The foundations of design thinking form an important platform in developing our investigation into human-technology interaction. One may begin this work with existing technical solutions and adapt the human user to the schema. However, it is also possible to think of these problems from a human point of view as is done in traditional human-centered design paradigms. In this section we present the definitions of traditional usability (Table I) and user experience (Table II) in terms of related attributes.

A. Early views of usability

Shackel's [4] definition of usability is one of the best-known. According to him, the attributes of usability are effectiveness, learnability, flexibility and attitude. Effectiveness refers to performance in the accomplishment of tasks (e.g. speed and errors). Learnability is the degree of learning to accomplish a task (e.g. time to learn and retention). Flexibility means adaptation to variations in tasks and/or environments, and attitude refers to user satisfaction with the system.

Nielsen [5] suggests that usability is associated with five usability attributes: learnability, efficiency, memorability, errors, and satisfaction. By learnability, Nielsen refers to how well a novice user can use the system, while efficiency has to do with efficient use by an expert. Memorability stands for an occasional use of the system. As far as errors are concerned, the system should have a low error rate, and this attribute also includes users recovering from possible errors. Satisfaction refers to the requirement that the system should be pleasant to use. According to Nielsen [5], these attributes are not unambiguous and they can be in conflict with each other to a certain degree.

The definition of usability most often referred to is probably the one in ISO 9241-11 [6] which defines usability as consisting of three attributes: effectiveness, efficiency, and satisfaction. Effectiveness refers to accuracy and completeness in achieving specified goals. Efficiency means resources expended in relation to the accuracy and completeness in achieving goals. Satisfaction refers to freedom from discomfort, and positive attitudes towards using the product.

ISO 9126 [7] identifies usability as one of six different software quality categories that are relevant during product development. The other categories are functionality, reliability, efficiency, maintainability, and portability. Furthermore, the usability category is decomposed into five attributes: understandability, learnability, operability, attractiveness, and usability compliance. Understandability refers to capability of a software product to enable the user to understand whether the software product is suitable or not. Learnability is the capability of a software product to enable the user to learn its application. Operability means the capability of a software product to enable the user to operate and control it. Attractiveness refers to capability of a software product to attract the user. Usability compliance refers to compliance with published style guides or conventions for user interfaces.

TABLE I
EARLY VIEWS OF USABILITY

Usability attributes			
Shackel	Nielsen	ISO 9241-11	ISO 9126
Effectiveness	Learnability	Effectiveness	Learnability
Learnability	Memorability	Efficiency	Operability
Flexibility	Errors	Satisfaction	Understandability
Attitude	Efficiency		Attractiveness
	Satisfaction		

B. Modern views of usability

Preece et al. [8] argues that when designing interactive products, both usability and user experience (UX) are needed. The former refers to the following attributes: efficiency (efficient to use), effectiveness (effective to use), safety (safe to use), utility, learnability (easy to learn), and memorability (easy to remember how to use). The latter refers to fun, emotional fulfilment, rewarding experiences, support to creativity, aesthetically pleasing feelings, motivation, helpful aspects, entertainment, enjoyment, and satisfaction.

TABLE II
MODERN VIEWS OF USABILITY

User experience attributes	
Preece	Quesenbery
Fun	
Emotionally fulfilling	
Rewarding	
Supportive of creativity	Effective
Aesthetically pleasing	Efficient
Motivating	Error tolerant
Helpful	Easy to learn
Entertaining	Engaging
Enjoyable	
Satisfying	

Quesenbery [9] suggests that the user dimensions of usability are associated with five attributes (five Es) described by the adjectives effective, efficient, engaging, error-tolerant, and easy-to-learn. It can be seen that one of the main motivation behind this classification is that computer systems have moved beyond being just work-based [10]. Although the five Es are mainly built on ISO 9241-11 [6], one different kind

of attribute, called engaging, is suggested. Engaging refers to the degree to which the tone and style of the interface make the product pleasant or satisfying to use.

1) *Fun*: Read, MacFarlane & Casey [11] suggest that the relationship between fun and usability is associated with three dimensions: expectations, engagement, and endurance (Especially in the context of measuring children's fun). Expectations refer to expectation of use and subsequent perception. This means that fun is composed of two components, (a) the fun that is attached to the current event, and (b) the fun that is attached to the prior expectations of the user. As an example, if we have high expectations and an event is then perceived to be dull, we feel disappointed, and vice versa. Engagement refers to the positive and negative instantiations (e.g. smiling, laughing, concentration etc.). Endurance is composed of two facets: remembrance and accessibility. The former is based on the Pollyanna principle: the likelihood to remember things that we have enjoyed. The latter refers to the desire to perform a funny activity again.

2) *Enjoyable*: Lin and Gregor [12] have reviewed relevant studies of enjoyment (especially as it is experienced by users of the Web), and they suggest that the concept of enjoyment is necessarily characterized by engagement in an activity, the resultant positive affect, and the fulfilment of some need or desire. Engagement in an activity refers to a situation in which attention is deeply focused on some activity. The resultant of positive affect refers to positive emotions (e.g. pleasure, happiness, contentment etc.) which result from use experience. The fulfilment of some need or desire refers to the fulfilment of user needs, including unexpected needs.

III. ACTION-ORIENTED USER INTERFACE DESIGN, USABILITY AND USER EXPERIENCE

The main goal in action-oriented user interface design is to develop a user interface interaction model which would allow users to use technology in the best possible manner. In other words, the users would be able to effectively reach their action goals. The action-oriented user interface development method is composed of four steps: defining the set of organized actions, constructing an organized tree structure, setting usability-related attributes for node states, and constructing user interface interaction model (see Figure 1).

A. Defining the set of organized actions

The first step in action-oriented interaction analysis is to define the relevant actions. One can use here the life-based analysis [13], the goal of which is to study the situation of human life and find the separate actions in it. There are also several other suitable methods and techniques for defining the relevant actions (see e.g. [14]).

B. Constructing an organized tree structure

In step 2, user interaction is decomposed into a form of a hierarchical tree composed of node states and operations. Clearly, whether the user interface is a graphical user interface (GUI), a Web user interface (WUI) or a touch user interface, the user controls the states of the interaction with the product by making selections (e.g. pressing buttons). There are always node states and operations which move the system from one node to another. The task of the user is to make the decisions which give meaning or sense to the transitions from one state to another.

As the nodes are contact points in interaction, we have to concentrate our analysis to these points. The given design criteria should thus be investigated in the context of the interaction in the decision nodes. These nodes do not have such a clear role when the actual operation is carried out. As an example, when the user navigates from one Web page (e.g. the main page of an online magazine) to another (e.g. a sports page of an online magazine), the navigation does not cause that many problems. However, if a Web advertisement in a form of a pop-up banner is added between these nodes, the nature of interaction will be quite different.

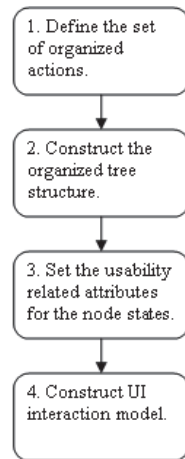


Fig. 1 The steps in action-oriented UI design.

C. Setting the usability-related attributes for the node states

Step 3 means that the nodes of tree structure are equipped with appropriate traditional usability attributes and user experience attributes. The traditional usability attributes (e.g. effectiveness, efficiency, memorability, and learnability) refer to the factors which foster the capability of the user to make a correct decision in the node. They help people pick up relevant information for making the right decision. The UX attributes (e.g. enjoyable, fun) give information about how user interface

designers can implement interfaces which the users like to use. Both usability and user experience aspects are important when the user interface interaction model is built in action-oriented user interface design.

As an example, a radio button can be associated with a single action. The designer must understand how to differentiate the critical button from other buttons and how to communicate that this button is important in reaching the defined action state. On the other hand, the designer must also think what the design should be like so that the users would find it cool or prestigious. This means the UX concepts look at the interface from a different point of view. While the usability concepts help in making a device easy to use, the UX concepts tell what looks fine and is enjoyable. In this way, the same node should be analysed from different points of view.

D. Constructing a UI interaction model

Step 4 refers to the organized tree structure description of the relevant interaction tasks. The hierarchical tree structure description is composed of node states with relevant usability and UX attributes and operations between the nodes.

IV. SUMMARY AND CONCLUSIONS

Usability-related attributes play a major role in the action-oriented user interface development method. In this method it is essential to consider usability and UX attributes not only at the level of a single interaction task but also at the level of single nodes. In this way, it is also possible to define usability-related requirements at the level of single state nodes. Consequently, this method is particularly suitable for UI-intensive systems with high demands to reduce risks.

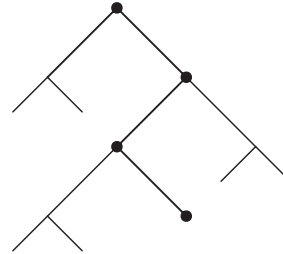


Fig. 2 An organized tree structure with task-specific usability and UX attributes. The relevant usability-related attributes are determined for each state node.

Usability and user experience concepts are interrelated. If the users are not able to reach their goals with technologies in an easy manner, they become irritated. They do not like slow or incomprehensible systems. It is also possible that a beautiful and enjoyable interface design makes it difficult to find and

comprehend crucial information. In this way, the two sets of attributes are often intimately connected to each other. Our action-oriented tree structure analysis gives roles for both types of usability-related attributes.

Our goal was to outline a design approach which is based on the idea of beginning interface design with the analysis of human actions and investigating separately the issues of how users can use interfaces and how they like to use them. The main motivation is that we have to unify “can” and “like” – perspectives in practical interaction design. The action-oriented user interface design and development scheme is composed of four steps: defining a set of organized actions, constructing an organized tree structure while keeping in mind how users can reach their action goals, setting usability-related attributes for node states, and constructing a user interface interaction model. Action-oriented user interface design defines the relations between usability and user experience attributes. It enables us to understand why the users really use the technology, how they could best deploy it and finally, what makes them feel good when they are reaching their action goals through the user interface.

The method of action-oriented user interface design can be integrated into early phases of human technology interaction design methods. For example, within the Life-Based design [13], the action-oriented user interface method can be deployed to build a user interface interaction model based on a set of organized actions. Furthermore, in the later phases of user interface design, early action-oriented UI prototypes can be evaluated by applying action-oriented user interface evaluation methods such as the D-TEO method [15].

REFERENCES

- [1] SaariLuoma, P., & Leikas, J. (2010). Life-Based Design - An Approach to Design for Life. *Global Journal of Management and Business Research*, 10(5), 27 – 33.
- [2] SaariLuoma, P. (2004). *Käyttäjäpsykologia [User psychology]*. Porvoo, Finland: WSOY.
- [3] SaariLuoma P., Parkkola H., Honkaranta A., Leppänen M., & Lamminen J. 2009. User psychology in interaction design – The role of design ontologies. In: SaariLuoma P., Isomäki H. (Eds.) *Future Interaction Design II* (69-86), Springer, Berlin.
- [4] Shackel, B. (1991). Usability: Context, framework, definition, design and evaluation. In B. Shackel & S. Richardson (Eds.), *Human factors for informatics usability* (pp. 21–38). Cambridge, MA, USA: Cambridge University Press.
- [5] Nielsen, J. (1993). *Usability engineering*. Boston: Academic Press.
- [6] International Organization for Standardization [ISO]. (1998, March). *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*. (Standard No. 9241-11). Geneva, Switzerland: ISO.
- [7] International Organization for Standardization [ISO] and International Electrotechnical Commission [IEC]. (2001, June). *Software engineering: Product quality–Quality model*. (Standards No. 9126-1). Geneva, Switzerland: ISO.
- [8] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-computer interaction*. Wokingham, UK: Addison-Wesley.
- [9] Quesenbery, W. (2003). The five dimensions of usability. In M.J. Albers, & B. Mazur (Eds.) *Content and complexity: Information design in technical communication*. (pp. 81–102). Mahwah, NJ: Lawrence Erlbaum Associates.
- [10] Stone, D., Jarrett, C., Woodroffe, M., & Minocha, S. (2005). *User interface design and evaluation*. San Francisco: Morgan Kaufmann Publishers.
- [11] Read, J. C., MacFarlane, S. J., & Casey, C. (2002). Endurability, engagement and expectations: Measuring children’s fun. In *Proceedings of interaction design and children, International workshop*. Eindhoven: Shaker Publishing.
- [12] Lin, C. H., & Gregor, S. (2001). Understanding and assessing the extent of enjoyment of Web experiences. In *Proceedings of the 17th Australasian Conference on Information Systems (ACIS 2006)*. Adelaide, Australia.
- [13] Leikas, J. (2009). *Life-based design. Form of life as a foundation for ICT design for older adults*. Jyväskylä studies in computing 105. Jyväskylä: Jyväskylä University Printing House.
- [14] Hoffman, R., & Millitello, L. (2009). *Perspectives on cognitive task analysis: Historical Origins and Modern Communities of Practice*. Boca Raton, FL: CRC Press/Taylor and Francis.
- [15] Lamminen, J., Leppänen, M., Heikkinen, R., Kämäräinen, A., & Jokisuu, E. (2009). A quantitative method for localizing user interface problems: The D-TEO method. *Human Technology, An Interdisciplinary Journal of Humans in ICT Environments*, 5(2), 121-145.

V

**INTERACTION DECONSTRUCTION METHOD FOR
USABILITY REQUIREMENTS ENGINEERING**

by

Lamminen, J., Rousi, R. & Saariluoma, P. 2011

In K. Blashki (Eds) Proceedings of the IADIS International Conference Interfac-
es and Human Computer Interaction 2011, Rome, Italy, July 24-26, 2011.
Portugali: IADIS Press, 161-168

Reproduced with kind permission by IADIS Press.

INTERACTION DECONSTRUCTION METHOD FOR USABILITY REQUIREMENTS ENGINEERING

Lamminen Juha

*Department Computer Science and Information Systems
University of Jyväskylä
P.O. Box 35, FI-40014 University of Jyväskylä
juha.e.lamminen@jyu.fi*

Rousi Rebekah

*Department Computer Science and Information Systems
University of Jyväskylä
P.O. Box 35, FI-40014 University of Jyväskylä
rebekah.rousii@jyu.fi*

Saariluoma Pertti

*Department Computer Science and Information Systems
University of Jyväskylä
P.O. Box 35, FI-40014 University of Jyväskylä
psa@jyu.fi*

ABSTRACT

This paper presents a novel usability requirements engineering method called Interaction Deconstruction, which supports user interface design in the interaction design process. The basic idea of the Interaction Deconstruction is to break down user actions into sub-actions and define, for each sub-action, the relevant usability-related attributes. This method is composed of three steps: defining the set of organized actions, constructing an organized tree structure, and setting the usability- and user experience related attributes for node states. The Interaction Deconstruction method differs favorably from existing usability requirements engineering methods. Firstly, it supports defining both usability and user experience requirements in parallel. Secondly, it utilizes psychological knowledge to engineer scientifically justifiable requirements. Thirdly, our method defines the relevant user-related attributes at the level of sub-actions. This is particularly beneficial in situations when interaction tasks are easy to separate into several sub-actions.

KEYWORDS

Non-functional requirements, usability attributes, usability requirements, user experience requirements, requirements engineering

1. INTRODUCTION

Today it is essential to develop requirements engineering (RE) methods and techniques for human-technology usability and UX design. This research is needed for industry to increase the speed of design and safeguard reliable outcomes and to improve communication between human and technical interaction usability design processes. Currently it is common to apply intuitive usability design processes in industry. Designers, who rely on scientific procedures when designing the electronics or mechanics of machines, tools and devices, may totally forget this attitude when they begin to construct interaction (Pahl, Beitz, Groete and Feldhuisen 2007). Nevertheless, there is no reason to assume that users are less complex than weather conditions or electronic circuits.

By usability and user experience design we refer to a design process that aims at designing interactive software products to support people in their everyday life. The process of interaction design typically involves four main activities (Preece et al. 1994; Rosson and Carroll 2002): identifying needs and

establishing requirements, developing alternative design, building interactive concepts and solutions, as well as evaluating the prototypes and solutions. In this paper we are interested in the process of establishing usability requirements for interactive software products.

The usability requirements definition process has mainly been discussed in usability engineering (UE) and user-centered-design (UCD) (e.g. ISO 9241-11; Mayhew 1999; Nielsen 1993; Preece et al. 1994; Quesenbery 2003; Wixon and Wilson 1997). However, the existing methods and techniques for usability requirements do not include the supports for facilitating the following issues:

- defining user experience requirements
- decomposing usability requirements into sub-requirements
- analyzing the usability-related attributes at the level of sub-actions
- clearly distinguishing usability and user experience components or cognitive and emotional usability
- utilizing scientific psychological knowledge

To solve the above problems and strengthen the definition of the usability requirements process, we propose a novel usability requirement analysis method called Interaction Deconstruction. This method is composed of three steps: defining a set of organized actions, which enable users to develop the technology (i.e., machine or program) towards the intended state, constructing an organized tree structure while keeping in mind the way in which users can reach their action goals, and setting usability-related attributes for node states.

We have strongly exploited earlier research on usability and user experience attributes to involve two of the most current and important interaction design aspects in our method: 1) how can people best use technology (e.g. traditional usability thinking)? And, 2) do people like to use a given technology (e.g. emotional usability or user experience research)? Our method borrows ideas from RE methods such as hierarchical task analysis in the experimental situation and analysis of user actions. Moreover, it goes further to emphasize the importance of usability and user experience attributes as the focus point of usability requirements engineering. Our method suggests analyzing user actions on the level of sub-actions (similar to the hierarchical task analysis) but our method also suggests adding the relevant usability-related attributes for each sub-action. Finally, our method supports the usability-related RE process of new and innovative user interfaces (UIs), overcoming many of the problems noted by literature when attempting to apply more traditional RE methods to similar tasks.

This paper is organized as follows: In section II a review of existing methods and techniques for usability requirements is presented; in section III, the definitions of traditional usability and user experience (in terms of related attributes) are presented; section IV presents the Interaction Deconstruction method; finally, in the last section, we summarize the results.

2. USABILITY REQUIREMENTS

The requirements can traditionally be divided into functional and non-functional requirements. The former can be seen as a system service or function (the intended actions of the system) and the latter as a constraint on the system or development process (how the system should perform these) (Kotonya and Sommerville 1998). While the interaction design process concerns both functional and non-functional requirements (Preece et al. 1994), in this paper we are interested in usability requirements and user experience requirements which are non-functional.

Usability requirements are qualitative or quantitative usability goals for a computer system. Qualitative requirements can be subjective goals, which are not easy to measure (e.g. user satisfaction). Quantitative requirements are objective and measurable goals (e.g. the time taken to complete the task) (Stone et al. 2005). Qualitative usability goals can be seen as general goals that guide design, whereas quantitative usability goals can serve as acceptance criteria during usability evaluation (Mayhew 1999). User experience requirements refer to the requirements concerning emotional aspects (e.g. using the intended system should be fun, enjoyable, and motivating). According to Preece (Preece et al. 1994) user experience requirements are typically subjective and difficult to quantify.

Usability attributes play a vital role in the usability requirements definition process. As an example, in the usability goal setting process by Wixon and Wilson (1997), the relevant usability attributes are determined based on specifying and categorizing the users and conducting task analysis. Also, Nielsen (1993) for example, has suggested that usability is associated with five usability attributes (learnability, efficiency, memorability, errors and satisfaction) and these attributes must be prioritized based on user analysis and tasks analysis when setting usability requirements. Nevertheless, there are still many open issues in developing a rational design process for usability.

Attributes turn the gaze towards evaluation rather than design. They are difficult to measure objectively and the outcome does not give us any idea about what should be done, when some objective outcome has been achieved. Knowledge of deficient learnability does not yet give any idea about what would be an ideal usability solution in that respect. Secondly, these traditional methods for usability requirements engineering (e.g. Mayhew 1999; Wixon and Wilson 1997) do not cover emotional and user experience requirements at all. Thirdly, few guidelines exist, or perhaps none at all, on how to define user experience requirements together with usability requirements and how to incorporate these two aspects of usability design into a rational design process. Here, we suggest that usability and user experience attributes and requirements be considered together in the usability-related requirements engineering process. Fourthly, traditional usability requirements engineering methods do not give clear (if any) instructions on how to incorporate psychological knowledge (i.e., memorability, learnability etc.) into this process. This, in turn leads to intuitive (folk psychological understandings) adoption and application of psychological concepts.

Current usability requirements engineering literature indicates that there is an obvious lack of empirical research reports on quantitative usability requirements in practice (Kantola and Jokela 2007), and that better methods for managing, organizing, and defining quantitative usability requirements should be developed (Jokela et al. 2006). Instead of quantitative usability requirements, literature presents many empirical research reports concerning qualitative usability requirements (e.g. Hermann and Heidmann 2002; Tsalgatidou et al. 2003). However, these empirical research reports typically refer to requirements that focus on technological limitations such as strong bandwidth variability, unpredictable disconnections, limited computing power and small memory size. This approach leads to usability requirements stemmed from technology (e.g., possibility to offline operation), in which concrete user interface design details are considered at a very general level (e.g., "User interface should be very simple and user friendly and the amount of presented information content limited and well specified"). Of course, technology related aspects need to be noticed when usability requirements are defined, but we suggest that much more scientific psychological knowledge about humans as users is needed as well.

The goal of user interaction is to develop the technology towards an ideal end state. This is a state in which people can realize their action goals. A person willing to pay bills on the net wants to find a webpage which makes it possible. The task of usability engineering is to make reaching the end state via the technology easy and pleasant. However, to reach its goal, usability engineering should have clear processes. One important part of this work is to define the precise process of reaching the end state of technology, which often presupposes proceeding through a number of steps. Therefore, we suggest that the first step towards utilizing scientific knowledge in usability-related requirements engineering is to analyze interaction tasks on the level of sub-tasks. This is where the user is in the process of working towards the hoped technologically assisted state. Once these steps have been achieved, we can define the relevant usability and user experience attributes on the level of the defined sub-actions.

3. FROM USABILITY TO USER EXPERIENCE

In this section we present the definitions of traditional usability and user experience in terms of related attributes. This chapter is relevant for understanding the method of interaction deconstruction. This chapter is especially relevant in terms of understanding sub-section 4.3, where we explain how to set the usability-related attributes of node states. Further, this chapter provides a background of usability traditions, against which we apply our psychological approach to understanding the content of usability requirements. The chapter begins with describing the earlier views towards usability and then progresses into discussion of more contemporary approaches leading to user experience. We use the concepts of fun and enjoyable in sub-sections 3.1 and 3.2 to illustrate what we mean by psychological content of user experience.

Shackel's (1991) definition of usability is one of the best-known. According to him, the attributes of usability are effectiveness, learnability, flexibility and attitude. Effectiveness refers to performance in the accomplishment of tasks (e.g. speed and errors). Learnability is the degree of learning to accomplish a task (e.g. time to learn and retention.). Flexibility means adaptation to variations in tasks and/or environments, and attitude refers to user satisfaction with the system.

Nielsen (1993) suggests that usability is associated with five usability attributes: learnability, efficiency, memorability, errors, and satisfaction. By learnability, Nielsen refers to how well a novice user can use the system, while efficiency has to do with efficient use by an expert. Memorability stands for the ability to use the system occasionally, without the need to re-learn it every time it is used. As far as errors are concerned, the system should have a low error rate, and this attribute also includes users recovering from possible errors. Satisfaction refers to the requirement that the system should be pleasant to use. According to Nielsen (1993), these attributes are not unambiguous and they can, to a certain degree, conflict with one another. Here, Nielsen mentions satisfaction as being a key component of usability, however, as with much user experience scholarship that we will shortly mention, this does not give a clear definition of what pleasant to use actually entails.

The definition of usability most often referred to is probably the one in ISO 9241-11 (1998) which defines usability as consisting of three attributes: effectiveness, efficiency, and satisfaction. Effectiveness refers to accuracy and completeness in achieving specified goals. Efficiency means resources expended in relation to the accuracy and completeness in achieving goals. Satisfaction refers to freedom from discomfort, and positive attitudes towards using the product. Again, this definition which is often repeated when referring to usability and user experience points towards feelings experienced during satisfaction. However, the way that they exist within this definition is still on an extremely general level. We are still left wondering what the core components of discomfort and positive attitudes actually are.

ISO 9126 (2001) identifies usability as one of six different software quality categories that are relevant during product development. The other categories are functionality, reliability, efficiency, maintainability, and portability. Furthermore, the usability category is divided into five attributes: understandability, learnability, operability, attractiveness, and usability compliance. Understandability refers to the capability of a software product to enable the user to understand whether the software product is suitable or not. Learnability is the capability of a software product to enable the user to learn its application. Operability means the capability of a software product to enable the user to operate and control it. Attractiveness refers to the capability of a software product to attract the user. Usability compliance refers to compliance with published style guides or conventions for user interfaces. Here, the aspect of attractiveness is raised, but it seems to be more vague than the aspect of satisfaction, due to the abstractness of what is actually attractive within the product – to whom is it attractive, and for what reasons?

We will return back to the 1990s in a moment, but jumping forward a decade, Quesenbery (2003) suggests that the user dimensions of usability are associated with five attributes (five Es) described by the adjectives effective, efficient, engaging, error-tolerant, and easy-to-learn. It can be seen that part of the main motivation behind this classification is that computer systems have moved beyond being just work-based (Stone et al. 2005). Although the five Es are mainly based upon the ISO 9241-11 (1998), one different kind of attribute, called engaging, is suggested. Engaging refers to the degree to which the tone and style of the interface make the product pleasant or satisfying to use.

Moving back into the 1990s, and somewhat addressing this notion of engaging, Preece et al. (1994) argue that when designing interactive products, both usability and user experience are needed. The former refers to the following attributes: efficiency (efficient to use), effectiveness (effective to use), safety (safe to use), utility, learnability (easy to learn), and memorability (easy to remember how to use). The latter refers to fun, emotional fulfillment, rewarding experiences, support for creativity, aesthetically pleasing feelings, motivation, helpful aspects, entertainment, enjoyment, and satisfaction. From a psychological perspective, the following sub-sections elaborate on what is fun and enjoyable in relation to user experience. These are not of course the only components of user experience, others such as motivation exist, but we use these to demonstrate the psychological dimension.

3.1 Fun

Read et al. (2002) suggest that the relationship between fun and usability is associated with three dimensions: expectations, engagement, and endurability (especially in the context of measuring children's fun). Expectations refer to the expectation of use and subsequent perception. This means that fun is composed of two components, (a) the fun that is attached to the current event, and (b) the fun that is attached to the prior expectations of the user. As an example, if we have high expectations and an event is then perceived to be dull, we feel disappointed, and vice versa. Engagement refers to the positive and negative instantiations (e.g. smiling, laughing, concentration etc.). Endurability is composed of two facets: remembrance and accessibility. The former is based on the Pollyanna principle: the likelihood to remember things that we have enjoyed. The latter refers to the desire to perform a fun activity again. Here, it is important to remember that numerous factors affect what is perceived as fun. Most notably, people's notions of what is fun changes with age and for example experiences. Also, perceptions of fun alter depending on culture, social conditions, profession, education and so forth. Thus, careful attention needs to be placed during research towards these influential factors.

3.2 Enjoyable

Lin and Gregor (2006) have reviewed relevant studies of enjoyment (especially as it is experienced by users of the Web), and they suggest that the concept of enjoyment is necessarily characterized by engagement in an activity, the resultant positive affect, and the fulfillment of some need or desire. Engagement in an activity refers to a situation in which attention is deeply focused on some activity. The resultant of positive affect refers to positive emotions (e.g. pleasure, happiness, contentment etc.) which result from use experience. The fulfillment of some need or desire refers to the fulfillment of user needs, including unexpected needs.

4. INTERACTION DECONSTRUCTION METHOD

The main goal of the Interaction Deconstruction method is to develop a human-technology interaction model of relevant user interface actions which would allow users to use technology in the best possible manner. In other words, the users would be able to effectively reach their action goals through a user interface. The Interaction Deconstruction method is composed of three steps: defining the set of organized actions, constructing an organized tree structure, and setting usability-related attributes for node states (see Figure 1). In step 1, we define the relevant actions on the level of high-level goals. In this case we use the example of internet banking, and within this first step the problem of reaching this high-level goal is difficulty in paying a bill. In step 2, we move towards examining the low-level goals, by looking at the sub-tasks of the operational actions or interactive intentions. In step 3, we articulate the usability-related attributes. This method can be used for example to develop totally new user interfaces and replace/update existing user interfaces.

4.1 Defining the set of organized actions

The first step of the Interaction Deconstruction method is to define the set of organized actions. In case a totally new and innovative user interface is developed, we suggest utilizing an intention-driven approach (Leppänen et al. 2010). This approach combines technological, social, and business viewpoints aiming to define the intention driven "use of the product" goals and requirements. In the Interaction Deconstruction method, these goals and requirements refer to the step defining the set of organized actions.

In the instance of replacing or updating the existing user interface, the set of organized actions refers to the certain user interface actions which are proposed to be repaired or replaced. For example, if a netbank system is needed to be renewed, the set of organized actions might refer to actions such as paying a bill, transferring money, watching stock markets, and buying pension insurance. There are several suitable methods and techniques for gathering and analyzing the relevant interactions from existing products (see e.g. Hoffman and Millitello 2008).

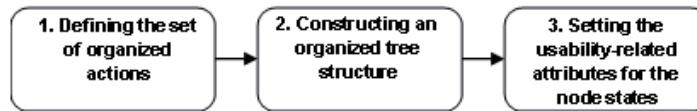


Figure 1. The steps in Interaction Deconstruction method

4.2 Constructing an organized tree structure

In step 2, user interaction is divided into a form of hierarchical tree, composed of node states and operations. At this point, one can use hierarchical task analysis (Annett 2003), the goal of which is to break a task into subtasks and then into sub-subtasks if needed. As an example, if the set of organized actions refers to user interaction with a netbank system (the previous example), the certain action “paying a bill by using a netbank system” can be broken down into subtasks: 1.1 log in; 1.2 select new payment; 1.3 enter the required payment information (e.g. account number, beneficiary’s name, amount, reference number, and message); 1.4 confirm payment by using confirmation code; and 1.5 log out. This refers to the interactive intention or low-level goals of the user.

4.3 Setting the usability-related attributes for the node states

Clearly, whether the user interface is a graphical user interface (GUI), a Web user interface (WUI) or a touch user interface, the user controls the states of interaction with the product by making selections (e.g. pressing the log in button). There are always node states and operations which move the system from one node to another. The task of the user is to make the decisions which give meaning or sense to the transitions from one state to another. As the nodes (e.g. 1.3 enter the required payment information) are contact points in interaction, we have to consider usability and user experience attributes at these points. As an example, when the user is entering the relevant payment information, preventing errors (usability attribute) is a more important attribute than fun or entertainment (user experience attribute). Furthermore, when the user is logging out, the feeling of trust with other user experience attributes are more important than usability attributes such as efficiency and effectiveness. It is important to find a balance between usability and user experience attributes on the level of sub-actions. This is the process which differs from previous usability requirement engineering processes.

Step 3 means that the nodes of the tree structure are equipped with appropriate traditional usability and user experience attributes. The traditional usability attributes (e.g. effectiveness, efficiency, memorability, and learnability) refer to factors which foster the capability of the user to make the correct decision in the node. They help people pick up relevant information for making the right decision. The user experience attributes (e.g. enjoyable, fun) give information about how user interface designers can implement interfaces which the users like to use. Both usability and user experience aspects are important when the user interface interaction model is built in action-oriented user interface design.

As an example, a radio button can be associated with a single action. The designer must understand how to differentiate the critical button from other buttons and how to communicate that this button is important in reaching the defined action state. On the other hand, the designer must also think about what the design should be like, in order to draw the appropriate experiential reaction. For example, should the users find the UI cool or prestigious? What is the intended experiential message that the owners of the program hope to transfer to the user? Thus, when considering user experience, the UI is considered from a different point of view. While the usability concepts help in making a device easy to use, the user experience concepts convey notions of what looks fine and is enjoyable. In this way, the same node should be analyzed from different perspectives. This leads into the construction of an organized tree structure description of the relevant interaction tasks. The hierarchical tree structure description is composed of node states (or sub-actions) with relevant usability and user experience attributes, plus operations between the nodes.

In addition to the obvious usability factors, we can analyze the above chain of sub-actions from the user experience perspective, starting from the netbank welcome and log in page. To ensure that a user will feel

comfortable with entering their account number and password on the log in page, the user should be able to not only identify that the current and subsequent pages are secure via symbols such as the padlock in the bottom right-hand corner of the page, or various trusted page symbols, but also authenticity within the layout design itself. Visually, the initial page needs to conform to the schematic themes seen throughout the organization. It needs to express order (Lynch 2010; Tractinsky 2004; Tractinsky et al. 2006).

Once the user has logged in, as indicated by the nodes chart above, steps to be taken in order to reach the intended state need to be clear, concise and direct. Coinciding with the usability factor is the UX of feeling that one understands the terminology of the steps, thus, simple language is necessary. The sense of control is then heightened by the interactive, personalizing attribute of the message function. When the steps have been completed and the user logs out, as mentioned before, it is necessary for maintaining the experience of trust, to include a message that the user is safely logged out. Thus, as within a narrative, there is closure to the set of actions.

This example illustrates how our understanding of UX differs and expands upon the previous UX models mentioned earlier in the article. Both Preece's and Quesenbery's models combine elements of usability (Preece – helpful; Quesenbery – effective, efficient, error tolerant and easy to learn), with elements encouraging active positive emotions (the rest of Preece's list and 'engaging' from Quesenbery's list). The problem with these elements is that they are not related to the experiential nature of banking. Perhaps in earlier times when people visited bank branches in person, we were attracted by the friendly, helpful service which covered a number of Preece's listed elements. But when dealing with an online UI responsible for the management of one's financial well-being, the experience the user is hoping to gain covers another side of the cognitive-emotional spectrum: the user's sense of trust and security is heightened by the feeling of control induced by a clear, understandable and straight-forward user interface operating on both visual/aesthetic and cognitive levels; and trust in the UI itself also comes through consistency (authenticity) of layout and symbols, operating on an aesthetic level. Rousi et al. (2011) discuss this in terms of the dimensions of user experience, which divides mental contents into four different (yet somewhat overlapping) categories: emotional, aesthetic, cognitive and practical.

5. CONCLUSION

Usability-related attributes play a major role in the Interaction Deconstruction method. In this method it is essential to consider usability and user experience attributes not only on the level of a single interaction task but also on the level of single nodes. In this way, it is also possible to define usability-related requirements on the level of single state nodes. Consequently, this method is particularly suitable for UI-intensive systems with high demands to reduce risks.

Usability and user experience concepts are interrelated. If the users are not able to reach their goals with technologies in an easy manner, they become irritated. They do not like slow or incomprehensible systems. It is also possible that a beautiful and enjoyable interface design makes it difficult to find and comprehend crucial information. In this way, the two sets of attributes are often intimately connected to each other. Our action-oriented tree structure analysis gives roles for both types of usability-related attributes.

Our goal was to outline a design approach which is based on the idea of beginning interface design with the analysis of human actions and separately investigating the issues of how users can use interfaces and how they like to use them. The main motivation is that we have to unify "can" and "like" –perspectives in practical interaction design. The action-oriented user interface design and development scheme is composed of three steps: defining a set of organized actions, constructing an organized tree structure, and setting usability-related attributes for node states. Interaction Deconstruction defines the relations between usability and user experience attributes. It enables us to understand why the users really use the technology, how they could best deploy it and finally, what makes them feel good when they reach their action goals through the user interface.

In this model, step 3, setting the usability-related attributes for the node states, is considered central as it differentiates the Interaction Deconstruction method from other usability requirements engineering methods. Step 3 defines the user experience requirements in relation to usability. This is achieved by clearly distinguishing usability and user experience components through charting nodes of interaction. Through

doing this, scientific psychological knowledge is applied to generate a more in-depth understanding of user experience.

REFERENCES

- Annett, J., 2003. *Hierarchical Task Analysis*. In D. Diaper & N. Stanton (Eds.), *The Handbook of Task Analysis for Human-Computer Interaction* (pp. 67–82). Lawrence Erlbaum Associates, Mahwah, NJ, USA .
- Hermann, F. and Heidmann, F., 2002. User requirement analysis and interface conception for a mobile, Location- Based Fair Guide. *Proceedings of the 4th International Symposium on Human-Computer Interaction*. pp. 388-392.
- Hoffman, R. and Millitello, L., 2008. *Perspectives on cognitive task analysis: Historical Origins and Modern Communities of Practice*. Taylor and Francis, New York.
- Jokela, T. et al, 2006. Quantitative Usability Requirements in the Development of the User Interface of a Mobile Phone. A Case Study. *In Personal and Ubiquitous Computing* Vol. 10, No. 6, pp. 345-355.
- International Organization for Standardization [ISO], 1998, March. *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*. (Standard No. 9241-11). Geneva, Switzerland: ISO.
- International Organization for Standardization [ISO] and International Electrotechnical Commission [IEC], 2001, June. *Software engineering: Product quality–Quality model*. (Standards No. 9126-1). Geneva, Switzerland: ISO.
- Kantola, N. and Jokela, T., 2007. Determining high level quantitative usability requirements: a case study. *Proceedings of the 12th international conference on Human-computer interaction: interaction design and usability (HCI'07)*, Julie A. Jacko (Ed.). Springer-Verlag, Berlin, Heidelberg, pp. 536-543.
- Kotonya, G. and Sommerville, I., 1998. *Requirements Engineering, Processes and Techniques*. John Wiley & Sons, Chichester, England.
- Leppänen, M. et al, 2010. A framework for intention-driven requirements engineering of innovative software products. *In: 19th International Conference on Information Systems Development (ISD 2010)*, Prague, Czech Republic, August 25 - 27 (in print).
- Lin, C.H. and Gregor, S., 2006. Understanding and assessing the extent of enjoyment of Web experiences. *Proceedings of the 17th Australasian Conference on Information Systems (ACIS 2006)*, Adelaide, Australia.
- Lynch, P., 2010. Aesthetics and Trust: Visual Decisions About Web Pages. *Proceedings of AVI'10*, Rome, Italy.
- Mayhew, D. J., 1999. *The usability engineering lifecycle: A practitioner's handbook for user interface design*. Morgan Kaufmann Publishers, San Francisco.
- Nielsen, J., 1993. *Usability engineering*. Academic Press, Boston.
- Pahl, G. et al. 2005. *Engineering design*. Springer, Berlin.
- Preece, J. et al, 1994. *Human-computer interaction*. Addison-Wesley, Wokingham, England.
- Quesenbery, W., 2003. *The five dimensions of usability*. In M.J. Albers, and B. Mazur (Eds.) *Content and Complexity: Information Design in Technical Communication* (pp. 81–102). Lawrence Erlbaum Associates, Mahwah, NJ, USA.
- Read, J.C. et al, 2002. Endurability, engagement and expectations: Measuring children's fun. *Proceedings of Interaction Design and Children*, International workshop, Shaker Publishing, Eindhoven.
- Rosson, M. B., and Carroll, J. M., 2002. *Usability engineering: Scenario-based development of human-computer interaction*. Morgan-Kaufmann, San Francisco.
- Rousi, R. et al, 2011. Unpacking the Contents: a conceptual model for understanding user experience in user psychology. *Proceedings of the Fourth International Conference on Advances in Computer-Human Interactions February 23-28, 2011*, Guadeloupe, France.
- Shackel, B., 1991. *Usability: Context, framework, definition, design and evaluation*. In B. Shackel & S. Stone, D. et al, 2005. *User interface design and evaluation*. Morgan Kaufmann Publishers, San Francisco.
- Tractinsky, N., 2004. Using Ratings and Response Latencies to Evaluate the Consistency of Immediate Aesthetic Perceptions of Web Pages. *Proceedings of the Third Annual Workshop on HCI Research in MIS*, Washington, D.C.
- Tractinsky, N. et al, 2006. Evaluating the Consistency of Immediate Aesthetic Perceptions of Web Pages. *International Journal of Human- Computer Studies*, Vol. 64, No.11, pp 1071–1083.
- Tsalgatiidou, A. et al, 2003. Mobile E-Commerce and Location-Based Services: Technology and Requirements. *Proceedings of the 9th Scandinavian Research Conf. on Geographical Information Science*, Espoo, FIN, pp. 1-14.
- Wixon, D. and Wilson, C., 1997. *The usability engineering framework for product design and evaluation*. In M. Helander, T. Landauer, & P. Prabhu (Eds.) *Handbook of human-computer interaction* (pp. 653–688). Elsevier, Amsterdam.

VI

A QUANTITATIVE METHOD FOR LOCALIZING USER INTERFACE PROBLEMS: D-TEO METHOD

by

Lamminen, J., Leppänen, M., Heikkinen, R., Kämäräinen, A. & Jokisuu, E. 2009

In Human Technology Journal, 5 (2), 121-145

Reproduced with kind permission by Human Technology Journal.

A QUANTITATIVE METHOD FOR LOCALIZING USER INTERFACE PROBLEMS: THE D-TEO METHOD

Juha Lamminen
*Agora Center
University of Jyväskylä
Finland*

Mauri Leppänen
*Department of Computer Science and
Information Systems
University of Jyväskylä, Finland*

Risto Heikkinen
*Department of Mathematics and Statistics
University of Jyväskylä
Finland*

Anna Kämäräinen
*Agora Center
University of Jyväskylä
Finland*

Elina Jokisuu
*Agora Center
University of Jyväskylä
Finland*

Abstract: *A large array of evaluation methods have been proposed to identify Website usability problems. In log-based evaluation, information about the performance of users is collected and stored into log files, and used to find problems and deficiencies in Web page designs. Most methods require the programming and modeling of large task models, which are cumbersome processes for evaluators. Also, because much statistical data is collected onto log files, recognizing which Web pages require deeper usability analysis is difficult. This paper suggests a novel quantitative method, called the D-TEO, for locating problematic Web pages. This semiautomated method explores the decomposition of interaction tasks of directed information search into elementary operations, deploying two quantitative usability criteria, search success and search time, to reveal how a user navigates within a web of hypertext.*

Keywords: *D-TEO method, usability, quantitative method, usability testing, log-based evaluation.*

INTRODUCTION

In the last two decades, the World Wide Web (Web) has become one of the most important means of disseminating and searching for information. Companies, government agencies,

municipalities, communities, and individual persons maintain a plethora of Web sites on the Internet, Intranets and Extranets, and the number of sites is increasing explosively (see Netcraft, 2009). Examples of drivers fueling this progress are eGovernment initiatives and programs that foster more efficient and effective provision of government services through the Internet (Cordella, 2007; Wolf & Krcmar, 2008). Web sites are often so large and lacking integration that finding a desired piece of information appears to be quite difficult and time consuming. It is not atypical that users become disoriented and “lost” in this hypertext world (Dillon, McNight, & Richardson 1990). The primary reason for these kinds of problems stem from poor design of Web sites (Nielsen, 1993).

Various principles (e.g., Nielsen, 1993; Schneiderman, 1998; Tidwell, 2005), techniques (e.g., Goldberg, Stimson, Lewenstein, Scott, & Wichansky, 2002), and methods (e.g., Beyer & Holtzblatt, 1998; Mayhew, 1999) have been developed for designing Web sites to satisfy usability criteria. *Usability* means “the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use” (International Organization of Standards [ISO], 1998). In our case, the *product* is a Web site composed of Web pages. Beyond the criteria indicated in the definition by ISO 9241-11 (1998), usability is seen to embrace other criteria, such as ease of learning, error rates, memorability, reliability in use, retention over time, throughput, and so on (cf. Constantine & Lockwood 1999; Nielsen, 1993; Preece et al., 1994; Schneiderman, 1992; Seffah, Donyaee, Kline, & Padda, 2006; Shackel, 1991; Wixon & Wilson 1997).

There is also a wide array of techniques (e.g., Chi et al., 2003; Paganelli & Paternò, 2002) and methods (e.g., Blackmon, Polson, Kitajima, & Lewis, 2002; Card et al., 2001) for evaluating and testing Web sites. The objective of carrying out an evaluation can be to test whether a design is appropriate, to compare alternative designs, or to check conformance to a standard (Lecerof & Paternò, 1998). Commonly applied methods are heuristic evaluation, usability testing, and log-based evaluation (cf. Matera, Rizzo, & Carughi, 2006). In a *heuristic evaluation* (Nielsen & Mack, 1994; Nielsen & Molich, 1990), the usability problems are identified in a heuristic fashion by a usability expert. The main concerns about a heuristic evaluation are that it does not include the actual end users (Nielsen & Mack, 1994) and the number of expert evaluators is often too low (Cockton & Woolrych, 2002). In *usability testing* (Dumas & Redish, 1993), the participants represent real end users and everything that they do and say during the usability test is observed and recorded. After the usability test itself, the data are analyzed and suggestions to eliminate the problems are proposed. The concerns regarding usability testing are that this process is based only on observational data and that user interface experience is needed to be able to interpret the data (Lecerof & Paternò, 1998). There is also the problem of cost and the time of the users and the observers (Lecerof & Paternò, 1998). In *log-based evaluation*, information about the performance of the users is collected automatically and stored in log files (e.g., Lecerof & Paternò, 1998). A benefit of this method is that large amounts of data can be collected in an exact form and with reduced work and cost (for more benefits, see Ivory & Hearst, 2001). The weak points of the method are that some handwork (e.g., adding code to the target system) is needed and the use environment is typically restricted to certain applications (Scholtz & Laskowski, 1998).

Usability evaluation and testing apply both qualitative (e.g., user satisfaction, easy to use) and quantitative measures (Mayhew, 1999; Stone, Jarret, Woodroffe & Minocha, 2005; Wixon & Wilson, 1997). The most common quantitative measures are task completion time, the

number of errors, and the success or failure in executing the tasks (e.g., Martin & Weiss, 2006; Masemola & De Villiers, 2006; Nielsen, Overgaard, Pedersen, Stage, & Stenild, 2006). Typically, values derived from the evaluations are compared to the predefined target values. The number of failed and successful attempts and the total number of attempts in each task are used to find out how difficult the task is. Masemola and De Villiers (2006) also use log files to record the number of mouse clicks. Others have combined quantitative measurements with qualitative evaluation to identify usability problems and evaluate the number and severity of the problems (e.g., De Angeli, Sutcliffe, & Hartmann, 2006; Duh, Tan, & Chen, 2006; White, Wright, & Chawner, 2006). Freeman, Norris, and Hyland (2006) have evaluated the navigation processes with the aim of getting a more accurate picture of a product's usability, particularly its efficiency (see more about evaluation methods in Ivory & Hearst, 2001).

Making a careful and in-depth usability evaluation of a large Web site requires significant time and resources (Dumas & Redish, 1993; Mayhew, 1999; Nielsen, 1993). Unfortunately, these often are not available in most situations. Therefore, there should be some means to first distinguish those parts of a Web site that seem to be more problematic, so that scant resources can be applied directly to a deeper evaluation of these areas only.

We propose a novel usability testing method, called D-TEO (Decomposition of Tasks into Elementary Operations), that aims to locate usability problems in the information search process in Web sites. The basic idea in D-TEO is to decompose a user task into elementary operations and define, for each task, an optimal navigation path composed of operations. In order to satisfy usability requirements, the structure and contents of a Web site should guide the users to find the optimal paths and to follow them efficiently. D-TEO helps identify Web pages that cause problems for the users and, based on this information, usability designers can focus their attention on these pages specifically.

This paper is organized as follows: In the next section, we define basic concepts related to Web sites, user tasks, information search, and search metrics. In the following section, we describe the proposed method. Later, we provide an example of the method in use, and then offer a short comparative review of related works. The final section presents a summary and conclusions.

BASIC CONCEPTS

Web Sites and Web Pages

A *Web site* is a collection of Web pages that is hosted on one or more Web servers. A *Web page* is a hypertext document, typically written in HTML or XHTML format. *Hypertext* involves data that are stored in a network of nodes connected by links. The interconnecting nodes form an interdependent web of information that is nonlinear. The nonlinearity enables great flexibility in the selection of information, but at the same time increases risks of disorientation.

There are two primary hypertext topologies (Batra, Bishu, & Donohue, 1993; Bernard, 2002). In the *strict hierarchical structure*, nodes are grouped in a hierarchical arrangement, allowing movement either up or down, but only one level at a time. In the *network topology* it is possible, in the most extreme case, to move through so-called referential hyperlinks from each node to every other node. Between these two types of topologies, there are mixed hierarchies, which allow limited movements from nodes to some other nodes at different levels within the structure.

A Web page consists of *user interface components*, such as titles, text boxes, data fields, tables, check boxes, radio and control buttons, menus, text links and image links, icons, forms, frames, and scroll bars. A user is allowed to make selections through menus or buttons, thus triggering the transmission of requests to the Web server to return the desired information in a new Web page. In the traditional Web application, communication between a client and a Web server is asynchronous, and the whole Web page is returned. In rich Internet applications, communication is synchronous and only part of the Web page can be substituted by a new one (Paulson, 2005; Preciado, Lanaje, Sanchez, & Comai, 2005).

User Actions, Tasks, and Operations

A user deploys an application as an instrument in order to improve his/her abilities to carry out some action (Saariluoma, Parkkola, Honkaranta, Leppänen, & Lamminen, 2009). *Actions* are composed of four kinds of tasks (Lecerof & Paternò, 1998). A *user task* is an action that is exclusively performed by a user, that is to say, without any interaction with the application. An *application task* is completely executed by the application. An *interaction task* is performed by the user interacting with the application. An *abstract task* requires complex actions whose performance allocation has not yet been decided. From the perspective of this paper, we are interested in interaction tasks. They can be further divided into categories, depending on the types of tasks the application makes: information search, on one hand, and information insert, update, and delete, on the other hand. Here, we only consider information search.

The tasks can be at different abstraction levels, ranging from high-level tasks to very low-level tasks. An execution of a task necessitates that all of its subtasks are carried out in a predefined manner. Decomposing a task into subtasks establishes a hierarchical tree in which subtasks on the lowest level are called *operations*; these elementary tasks focus on a single user interface component (e.g., the OK button). The execution of an operation triggers the transmission of a request to the Web server to search for the desired information and return it in a new Web page. An operation can also return a previous page (i.e., back-page button).

Information Search

Web sites show up as webs of hypertext that contain information of interest to the user. Information can be searched for in two ways (Bernard, 2002). The first type is a *directed search*, also called explicit search (Norman & Chin, 1988). The purpose of this type of search is to acquire specific information about a target item (e.g., find the title of the 1953 film that starred Audrey Hepburn and Gregory Peck). The second type is an *exploratory search* that involves the broader goals of finding and integrating information from various nodes in a web of hypertext. This is also called browsing (Canter, Rivers, & Storrs, 1985). A user explores the hypertext by continually refining his/her search until the information goal is satisfied. An example of this kind of search is “Compare the movies *Independence Day* and *Sleepers* by using the information the MovieGuide can give you” (Bernard, 2002). An exploratory search takes more time and causes disorientation more often, partly because it poses more cognitive burden (Kim & Hirtle, 1995; Norman & Chin, 1988; Smith, 1996). We focus on directed searches in this study.

A page containing the target item is called a *terminal node*. Information search proceeds from an entry node to the terminal node through hyperlinks. The shortest route to the specific

terminal node that satisfies a search task is called an *optimal path* (Bernard, 2002; Gwizdka & Spence, 2007; Norman & Chin, 1988). The length of the path depends on how many nodes (Web pages) have to be visited during the search. In a Web site following a mixed or network topology, there may be several optimal paths for one information search.

Metrics

How effectively and efficiently a desired piece of information can be found is influenced by several factors, including size of the web of hypertext, the breadth and depth of hypertext topology and its compliance with the users' mental models, the visualization of Web pages, the understandability of terms used in Web pages, and so on. Generally speaking, the effectiveness and efficiency of the search task depend upon the usability of a Web site. The literature presents a large variety of definitions and taxonomies for usability (e.g., Constantine & Lockwood, 1999; ISO, 1998; Nielsen, 1993; Preece et al., 1994; Schneiderman, 1992; Shackel, 1991). It goes beyond the scope of this paper to discuss them in more detail (see the analysis by Seffah et al., 2006). Therefore, we quote ISO 9241-11 (1998), which distinguishes between three main usability attributes: effectiveness, efficiency, and satisfaction. We are particularly interested in search efficiency.

Search efficiency is commonly measured in terms of search time, navigation accuracy, lack of disorientation, and success in finding the desired page. For instance, Bernard (2002) defined timed accuracy as the number of times a user fails to find the correct terminal node. Search efficiency is measured by examining the number of deviations from the optimal path and by the number of total back-page presses used in reaching the targeted node. Search time means the time taken to correctly complete the given task.

Our metrics of search efficiency is composed of two criteria: search success and search time. *Success* measures the extent to which a user follows the optimal path when he/she is carrying out the task. Deviations from the optimal path, or back and forth movement in the path (e.g., through the back-page button), decreases the measure of success. Usage of the back-page button suggests uncertainty in the navigation paths taken (cf. Norman & Chin, 1988). *Search time* represents the time that it takes the user to complete the task from start to finish. Since a task is decomposed into elementary operations, it is also possible to measure the time required to carry out an operation, that is, how long from the end of one operation to the end of the next operation. For each of these two criteria, a set of measures were defined and used.

METHOD

This section describes the proposed method for locating problems in Web pages. We first describe the objectives and application domain of the method. Then, we detail the steps of the method.

Application Domain

D-TEO is a usability testing method for revealing problems in user navigation in Web sites. The Web sites can be either in the prototype phase or in production. The basic idea underlying the method is to examine how closely a user follows the optimal paths and how fast he/she performs the given interaction tasks. Deviations from the optimal path and/or

delays in executions indicate problems that should be examined more closely with some other usability evaluation methods (e.g., heuristic methods).

The D-TEO can be integrated into a Web application development method or a hypermedia development method. The literature provides a large variety of these kinds of methods (e.g., OOHDM, Rossi & Schwabe, 2006; RMM, Isakowitz, Stohr, & Balasubramanian, 1995; IDM, Lee, Lee, & Yoo, 1998; W2000, Baresi, Garcotto, & Paolini, 2001; UWE, Hennicker & Koch, 2001; WHDM, Lee & Suh, 2001). For example, within the WHDM method, the D-TEO technique can be deployed to test prototypes produced by the design activities of navigation design and interface design.

Steps

The D-TEO method is composed of four steps: (a) define the goals of the test and the user profiles, (b) devise the test tasks and identify the optimal paths, (c) organize the test and collect data, and (d) analyze the data and make the conclusions (Figure1). In the following subsections, the steps are described in more detail.

Define the Goals of the Test and the User Profiles

The use of the method starts with defining the goals for the test at hand. The goal statements should describe which subset of the Web pages should be tested, which pages in this subset are particularly important, and how the results from the test are to be utilized. The goal setting is affected by whether the Web site is in the prototype phase or already in use, and the reasons triggering the test. In order to define the user profiles, it is important to identify the audience at whom the Web site is targeted. Conducting a survey or interviews among the current and potential users helps to define the typical characteristics (user profiles) of the primary user categories in terms of their skills, motivations, experience, and so on (e.g., Mayhew, 1999).

Decisions regarding which user groups, and to what extent, are included in the test are based on the goals of the test and the resources available.

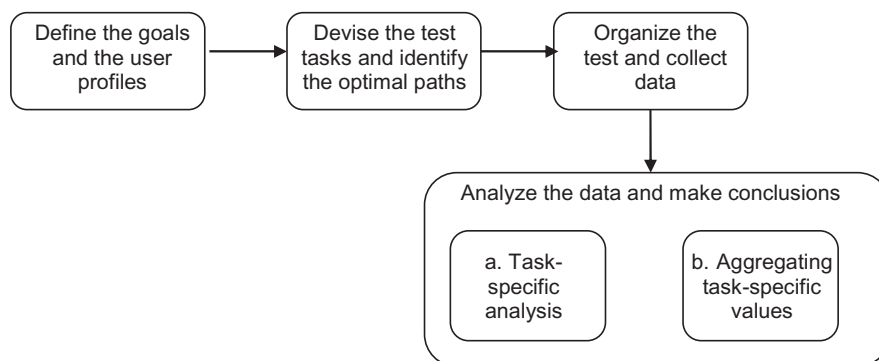


Figure 1. The steps of the D-TEO method.

Devise the Test Tasks and Identify the Optimal Paths

A *test task* is a typical interaction process carried out by a person representing an appropriate user profile. In order to devise a set of relevant test tasks, the overall structure of the Web site has to be outlined and typical interaction tasks should be recognized through a task analysis. If there is a site map describing the Web site, it can be used to ascertain that the test tasks cover a sufficient number of Web pages. Whether the coverage is sufficient or not is determined based on the goals of the test. As an example, let us assume that one of the test tasks is as follows: “There is one ringette team in the Jyväskylä region. What is the name of this team?”

After specifying the test tasks, they are decomposed into operations. As defined above, an operation is an elementary task that focuses on a single user interface component. To establish decomposition hierarchies of test tasks requires that those responsible for testing have good knowledge about the topological structure of the Web site and details of page visualization. For each test task, it is determined which Web pages should be visited and what operations should be performed, in order to reach the terminal page containing the desired piece of information. The shortest path from the entry page to the terminal page is an optimal path. Because the method is intended for testing directed searches, typically only one, or just a few, optimal paths exist for each test task. If there are several paths with the same number of operations, these paths are analyzed as equals.

As an example of the optimal path, let us consider the test task introduced above. For purposes of analysis, each Web page involved by the test tasks is coded with a number reflecting its position in the hierarchical structure of the Web site. By doing so, we have found the optimal path for this test task is as follows:

0 → 4 → 4.11 → 4.11.4 → 4.11.4.3.

In this coding, 0 means the Entry page, 4 refers to the page Services, 4.11 represents the page Sport (under the Services page), 4.11.4 means the Sports Clubs page (under the Sport page), and finally 4.11.4.3 refers to the page containing the information about the ringette team. The optimal path can be described as an ordered set of numbered Web pages visited. It also shows the operations a user must follow in order to complete the test task. The marking $p_i \rightarrow p_j$ denotes the operation by which a user navigates from one Web page (p_i) to another (p_j).

Organize the Test and Collect Data

The test participants are selected to meet the stated goals of the test and the defined user profiles. The number of participants can vary, depending on the goals of the test. Nielsen and Molich (1990) state that 50% of the most important usability problems can be identified with three users. Other authors claim that five users facilitate the discovery of 90% of the usability problems (e.g., Virzi, 1992).

The test tasks are given to the participants on a sheet of paper. No discussion between a participant and the test facilitator is needed during the test. The test equipment should record, with time stamps, all the actions the participant makes and all the Web pages he/she visits. In addition, the test facilitator can make notes on the behavior of a participant, which can be used later in the analysis of time stamped data. This is, however, optional.

To illustrate the data that is collected, let us continue with our previous example. The optimal path was defined as: 0 → 4 → 4.11 → 4.11.4 → 4.11.4.3. Table 1 depicts the unprocessed data collected about operations by three test participants (P1-P3).

We can see in Table 1 that participant P1 followed the optimal path and successfully completed the test task. Participant P3, on the other hand, carried out the operation 0 → 4 successfully but failed to execute the operation 4 → 4.11 (see 00 in Table 1). He/she also failed to complete the operation 4.11 → 4.11.4 and interrupted at 10:23.

Analyze the Data and Draw Conclusions

The collected data is processed and analyzed in two phases. First, the measure values for single tasks and single participants are derived and analyzed. After that, the operation-specific values are aggregated to concern all the tasks and participants. Based on these analyses, conclusions are drawn.

a. Task-Specific Values

The D-TEO method deploys metrics derived from two evaluation criteria, success and search time (see above). The metrics for task-specific analysis comprise two measures, success value and duration time. These are elaborated here.

For each task, and for each participant, the next questions are considered:

- *How successfully did the participant navigate from one Web page to another along the optimal path?*

This is measured by *Success Values (SV)* that are derived by the following rule: If an operation in the optimal path was carried out in the first attempt, then $SV = 1$ for that operation; by the second attempt, $SV = 0.5$; by the third attempt, $SV = 0.33$, and so on. If the participant deviated from the optimal path, without returning to it, the

Table 1. Example of Unprocessed Data.

P1		P2		P3	
ID	T	ID	T	ID	T
0	10:00	0	14:29	0	9:41
4	10:03	4	14:33	4	9:47
4.11	10:10	4.11	15:04	00	9:55
4.11.4	10:15	00	15:16	00	10:04
4.11.4.3	11:27	4.11	15:20	000	10:23
		4.11.4	15:34		
		4.11.4.3	15:55		

Note: ID means the numeric identifier of the Web page. T stands for the clock time (in minutes and seconds) when a participant arrived at a certain Web page. We use the symbol 00 to refer to a Web page that is not on the optimal path. The symbol 000 means that the participant has interrupted the execution of the task.

operation is coded with the number 0. In the event the participant did not find the Web page that is a part of the optimal path, the operation is coded by NA.

▪ *What was the duration of each operation?*

It is important to study the time that the participant spent on each Web page (i.e., performing each operation). *Duration* (D) is the difference between time the participant arrived at the page and the time when he/she left the page (by executing the operation). If the participant realized that he/she made a mistake (i.e., deviated from the optimal path) and returned to the previous page, the duration time is the sum of the duration times he/she spent on the page in each visit.

To continue with the example data, consider Table 2. It contains the durations (D) and success values (SV) for the operations of the task by three participants P1, P2, and P3. The duration values (in seconds) have been derived from the clock times in Table 1. The success values have been calculated based on the aforementioned rules. We can see in Table 2 that the participant P2 spent a relatively long time (31 seconds) in performing the operation 4 → 4.11, although he/she finally completed the task. P2 also had problems with the operation 4.11 → 4.11.4 because he/she could not find the page 4.11.4 until the second attempt (SV = 0.5). The participant P3 managed to carry out only the first operation of the task.

b. Aggregating Task-Specific Values

Here, we consider the two evaluation criteria, success and duration, through the following aggregated measures:

- *Average Success Value* (ASV) for an operation. This is obtained by calculating the average success value for the operation in the task across all the participants. The smaller the ASV, the more probable it is that the concerned Web page contains problems.
- *Average Duration* (AD) for an operation. This is derived by calculating the average duration for the operation in the task across all the participants. A large AD value indicates problems in the concerned Web page.

Table 2. Example Data Expressed in Success Values and Durations.

Operation	P1		P2		P3	
	SV	D	SV	D	SV	D
0 → 4	1	3	1	4	1	6
4 → 4.11	1	7	1	31	0	8
4.11 → 4.11.4	1	5	0.5	26	NA	NA
4.11.4 → 4.11.4.3	1	72	1	21	NA	NA

Note: SV represents how successfully the participant moved along the optimal path. Reaching the correct page on the first try results in 1, by the second try, 0.5, by the third attempt, 0.33, and so forth. D represents the time duration for the participant to successfully move to the correct next page, and includes any time spent recovering from poor choices.

- *Standard Deviation of Durations (SD)*. This is calculated from the duration of the operation across all the participants. A large SD indicates problems.

The three aggregated measures calculated for the operations of one task, performed by three participants (see Table 2), are presented in Table 3. We can see that the operation 0 → 4 is the only one that is performed successfully by all the participants. In all the other operations, there have been some deviations from the optimal path or additional attempts.

The critical question to determine is when a certain aggregated value for some operation is so large (for AD and SD) or so small (for ASV) that the concerned Web page should be investigated more closely for usability problems. We approach this question by aggregating the values of the operations and examining the deviating values in the statistical distributions of these three measures. We calculate fractiles to specify the limits that are then used as the criteria for identifying the problematic Web pages.

The next issue is to determine the suitable fractile for each task. Selecting too large a fractile increases the risk of ignoring some problematic Web pages. Conversely, if too small a fractile is chosen, it may lead to selecting too large a set of problematic Web pages, thus increasing the need of resources for a closer examination. The suitable fractile depends on the situation. We recommend the use of probability theory to determine a suitable fractile. The probability that at least one of the three measures recognizes a Web page as problematic is $1-p^3$ if all of the measures are independent of one another.

In the formula above, p stands for a fractile (decimal number) and 3 is the number of the measures (i.e., ASV, AD, SD). The assumption of independent measures is not exactly true, but we still use this formula as an approximation.

Table 4 presents the probabilities for four different fractiles. We can see that with the 75% fractile about 58% of the Web pages are regarded as problematic. Correspondingly, with the 95 % fractile about 14 % of the Web pages should be selected for further examination. In actuality, the probabilities are a bit smaller than indicated by the formula because the very problematic operations often are identified through more than one measure, due to some correlations between the measures.

Table 3. Aggregate Measures of the Example Data.

Operation	ASV	AD	SD
0 → 4	1.00	9.17	9.37
4 → 4.11	0.67	18.67	15.83
4.11 → 4.11.4	0.63	12.25	9.39
4.11.4 → 4.11.4.3	0.88	28.75	29.32

Note: ASV means average success value, AD means average duration, and SD means standard deviation of durations.

Table 4. Probabilities of Four Fractiles.

Fractile	75%	80%	90%	95%
Probability	$1-0.75^3 = 0.578$	$1-0.80^3 = 0.488$	$1-0.90^3 = 0.271$	$1-0.95^3 = 0.143$

The D-TEO method does not prescribe the use of any specific fractile because it depends on the situation and available resources. Instead, we offer some guidelines for selecting a fractile. A large fractile can be selected if

- the number of operations in the test tasks is large
- it can be assumed that there are only a few problems
- there is a limited amount of resources available for further examination

Conversely, a small fractile can be selected if

- the number of the operations is small
- if many problematic operations are expected to appear
- if there are sufficient resources for closer examination of the problem pages.

The final decision on whether to include a particular Web page in a set of problematic pages should be discussed with the user interface designer to avoid misinterpretations. Often, if the measures are calculated based on small samples, exceptional deviations from the standard values may appear. Thus, we emphasize that the values as such do not directly indicate which pages are problematic. The test results are best used to localize those areas in the structure of the Web site that should be analyzed more carefully.

After having determined the set of problematic Web pages, a variety of methods can be applied to identify the reasons for usability problems within specific Web pages. We suggest the use of the interaction design patterns of Tidwell (2005) and van Duyne, Landay, and Hong (2006). If inconsistencies or deficiencies are recurrent in the Web pages, stemming possibly from the applied screen design standards, changes should be extended to involve all the Web pages with similar structures. The screen design standards then should be updated correspondingly. After having made the changes, the improved Web pages can be heuristically inspected, if time and resources are allowed.

AN EXAMPLE OF THE D-TEO METHOD IN USE

In this section, we describe how the D-TEO method was used in testing the Web site of the Jyväskylä, Finland, region.¹ This is not a case study in a strict sense but rather an example for illustrating the application of the method. The description proceeds in a step-by-step manner.

Define the Goals of the Test and the User Profiles

In this first step, we determined who the stakeholders involved with the Web site were, how the results of the test were going to be used, what the stage of development (e.g., completed product, prototype, etc.) of the Web site was, and which parts of the Web site should be tested. It was concluded that the Web site was a finished prototype and that the results of the test would be used to finalize it prior to implementing as the final version. The Web site was to be tested in its entirety, a feasible task because the Web site was relatively small scale (approximately 1,300 pages) and hierarchically compact. When considering the user groups, it was thought that the Web site could be useful, for instance, for tourists planning trips to the Jyväskylä region. Their primary need would be, for example, to find accommodations in the region. We did not define any explicit user profiles.

Devise the Test Tasks and Identify the Optimal Paths

The Web site was aimed at providing information about living, working, studying, and traveling in the Jyväskylä region. The main menu covers living, municipalities, travel, services, recruitment, and events. It was decided that each of the main menu items should be selected for at least one task. We coded the Web pages corresponding to the main menu items with numerical codes instead of the URL addresses in order to make the analysis easier. Therefore, 1 = Living, 2 = Municipalities, 3 = Travel, 4 = Services, 5 = Recruitment, and 6 = Events.

There was no site map available, and hence we had to go through the paths to form a sufficient overview of the hypertext topology. Based on the structure of the Web site, we constructed a test story to include eight test tasks (see the Appendix for a description of the test tasks). We ensured the validity of the test tasks by checking that each of the tasks could be carried out and optimal paths could be specified. In this phase, some of the tasks had to be changed or made more detailed in order to fulfill the objectives above.

Organize the Test and Collect Data

When the aims of the testing were discussed with the client, it became apparent that no specific user group could be identified. Because no specific user group could be identified, the participants were randomly selected from a group of volunteer university students. Eleven native-Finnish-speaking participants participated in the study conducted in Finnish, one of whom took part in the pilot test to elaborate the test tasks. Thus, the results of 10 participants were included in the statistical analysis.

The tests were conducted in a usability laboratory at the university. The test data was collected using Windows Media Encoder, and the results were analyzed with the statistical software environment R. The time stamping was made manually.

Analyze the Data and Draw Conclusions

The data were analyzed in the manner of the instructions given in the Methods section. Problems in the user interface were localized by calculating the ASV, AD, and SD for every task and every operation. Table 5 presents the values for each of the 25 operations within the eight test tasks.

Figures 2, 3, and 4 represent the distributions of the values of ASV, AD and SD, respectively. Frequency in the histograms means the number of operations. In Figure 2, for instance, there are two operations with ASVs less than or equal to 0.4. Problems can be located in those Web pages that are involved by the operations situated at the extreme ends of the distributions (cf. the two operations in Figure 2), indicated by the circled areas.

To decide which Web pages should be selected for closer examination, a suitable fractile had to be determined. In order to avoid rounding problems in defining the critical values, the possible fractiles were 72% (18 of 25 Web pages), 76% (19/25), 80% (20/25) and 86% (21/25). Because we did not want to select too many Web pages, we used the 76% fractile, implying that the probability of recognizing a random Web page as problematic is 0.56 ($1-0.76^3$). Critical values for the three measures were determined according to the selected fractile. The critical limit of ASV is 0.60. The Web pages involved by the operations with smaller values were considered to be problematic. Correspondingly, the critical limits for AD and SD are 23.38 and 17.8.

Table 5. Average Success Values (ASV), Average Durations (AD) and Standard Deviations of Durations (SD) for Each Task and Each Operation.

Task	Operation	ASV	AD	SD
1	0 → 4	0.70	11.80	12.00
	4 → 4.3	0.60	34.90	23.30
	4.3 → 4.3.1	0.75	24.50	3.40
	4.3.1 → 4.3.1.1	1.00	3.12	0.60
2	0 → 6	0.20	19.60	39.00
	6 → 6.66	1.00	42.38	10.60
3	0 → 3	0.80	7.50	2.80
	3 → 3.2	0.80	16.20	17.80
	3.2 → 3.2.3	0.45	10.70	6.80
	3.2.3 → 3.2.3.7	0.70	5.00	2.00
4	0 → 4	0.50	7.70	4.00
	4 → 4.11	0.68	44.10	32.70
	4.11 → 4.11.4	0.50	13.25	5.60
	4.11.4 → 4.11.4.1	0.56	41.63	41.20
5	0 → 4	0.90	8.00	7.30
	4 → 4.11	0.75	17.30	13.90
	4.11 → 4.11.4	0.69	9.50	7.20
	4.11.4 → 4.11.4.3	0.81	23.38	24.20
6	0 → 1	0.40	5.20	4.00
	1 → 1.1	0.80	3.80	1.30
	1.1 → 1.25.6.1	1.00	7.13	3.30
7	0 → 4	0.80	13.30	7.10
	4 → 4.19	1.00	28.00	16.60
	4.19 → 4.19.5	1.00	22.50	16.70
8	0 → 0.100	0.70	21.40	18.40

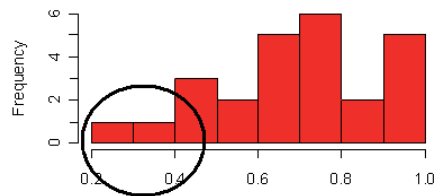


Figure 2. Histogram of average succeed values (ASV).

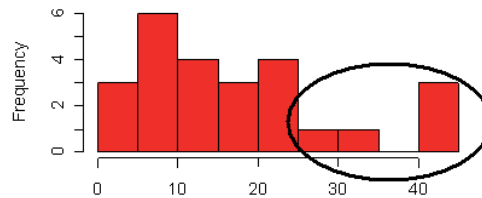


Figure 3. Histogram of average durations (AD).

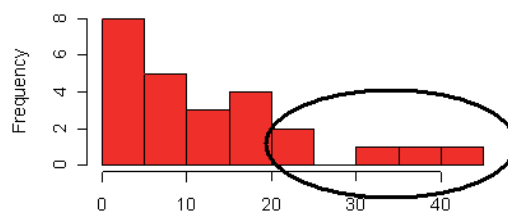


Figure 4. Histogram of standard deviations (SD).

Table 6 shows the numbers exceeding the critical limits in bold (cf. Table 5). The Column # denotes how many of the three aggregate measures suggest that the operation (Web page) is problematic. If more than one of the measures exceeds the critical limits, it is even a stronger indication of problems. The operation 4.11.4 → 4.11.4.1 appears to be a problem candidate based on all three measures. Three operations (4 → 4.3; 0 → 6; 4 → 4.11) appeared to be problem candidates based on two measures. In total, 13 out of 25 operations were selected as problem candidates.

As noted in the Methods section, the aggregated measures are correlated with one another. We calculated Spearman’s rank correlation coefficient (Spearman’s rho):

$$\text{cor}(\text{ASV}, \text{AD}) = -0.01, p = 0.96$$

$$\text{cor}(\text{ASV}, \text{SD}) = -0.187, p = 0.372$$

$$\text{cor}(\text{SD}, \text{AD}) = 0.804, p < 0.001$$

There is no correlation between ASD and AD, and the negative correlation between ASV and SD is not statistically significant. There is a significant correlation between SD and AD, meaning that the operations with high AD values tend to have high SD values. Because of this correlation, probability calculations are only approximations.

Hence, we distinguished 13 problem candidates for more careful consideration. Each problem candidate was mapped to a specific Web page. One of those Web pages was the Sport page (coded with 4.11.4; see Figure 5), which is the target of the operation 4.11.4 → 4.11.4.1 (cf. Table 6). When the Sport page and its UI components were analyzed more carefully interaction design patterns (see Methods section), several usability problems were found. For example, some text fields, labels, and links were not arranged in a systematic manner, the Search button was difficult to notice, and it was difficult to distinguish the labels from the text. The problems could be solved applying UI design patterns (Tidwell, 2005; van Duyne et al., 2006).

Table 6. Critical Values (Boldface) for Each Operation.

Task	Operation	ASV	AD	SD	#
1	0 → 4	0.70	11.80	12.00	
	4 → 4.3	0.60	34.90	23.30	2
	4.3 → 4.3.1	0.75	24.50	3.40	1
	4.3.1 → 4.3.1.1	1.00	3.12	0.60	
2	0 → 6	0.20	19.60	39.00	2
	6 → 6.66	1.00	42.38	10.60	1
3	0 → 3	0.80	7.50	2.80	
	3 → 3.2	0.80	16.20	17.80	
	3.2 → 3.2.3	0.45	10.70	6.80	1
	3.2.3 → 3.2.3.7	0.70	5.00	2.00	
4	0 → 4	0.50	7.70	4.00	1
	4 → 4.11	0.68	44.10	32.70	2
	4.11 → 4.11.4	0.50	13.25	5.60	1
	4.11.4 → 4.11.4.1	0.56	41.63	41.20	3
5	0 → 4	0.90	8.00	7.30	
	4 → 4.11	0.75	17.30	13.90	
	4.11 → 4.11.4	0.69	9.50	7.20	
	4.11.4 → 4.11.4.3	0.81	23.38	24.20	1
6	0 → 1	0.40	5.20	4.00	1
	1 → 1.1	0.80	3.80	1.30	
	1.1 → 1.25.6.1	1.00	7.13	3.30	
7	0 → 4	0.80	13.30	7.10	
	4 → 4.19	1.00	28.00	16.60	1
	4.19 → 4.19.5	1.00	22.50	16.70	
8	0 → 0.100	0.70	21.40	18.40	1

RELATED WORK

In this section, we make a short review of related work and discuss how our method differs from and performs among the existing methods. Our taxonomy for the review is composed of five general dimensions and three specific dimensions. The general dimensions, borrowed from Ivory & Hearst (2001), are UI, method class, method type, automation type, and effort level. UI distinguishes between WIMP (windows, icons, pointer, and mouse) interfaces and Web interfaces. Method classes are testing, inspection, inquiry, analytical modeling, and simulation. Method types include, for example, thinking aloud, log file analysis, guideline review, feature inspection and the like. Automation type is used to specify which aspects of a method are automated (i.e., capture, analysis, critique). Effort level indicates the human effort



Figure 5. The Sports page.

required by a method in use. The options are (a) minimal effort, (b) model development (M), and (c) informal (I) and formal (F). (See more about the options in Ivory & Hearst, 2001). The first specific dimension distinguishes basic concepts and constructs used to conceptualize user behavior and Web sites (e.g., user task, operation, navigation path). The second specific dimension differentiates criteria used to evaluate user interaction usability. The third specific dimension shows how the evaluators interpret the results of the evaluation.

The UI literature suggests a large array of evaluation methods (cf. Ivory & Hearst, 2001, distinguish 75 WIMP user interface evaluation methods and 57 Web user interface evaluation methods). We selected only those methods that are most relevant to our comparative review. The reviewed methods are UsAGE (Uehling & Wolf, 1995), QUIP (Helfrich & Landay, 1999), USINE (Lecerof & Paternò, 1998), RemUSINE (Paternò & Ballardin, 1999, 2000) and WebRemUSINE (Paganelli & Paternò, 2002). The results are summarized in Tables 7 and 8. The D-TEO method is included in the tables to facilitate the comparison.

In UsAGE (Uehling & Wolf, 1995) and QUIP (Helfrich & Landay, 1999), the goal is to automate the detection of serious usability problems by comparing the users' task to the task performed in the "right" manner. What constitutes the "right" manner is defined by the developer of the system. Ivory and Hearst (2001) call this kind of approach Task-Based Analysis of Log Files. In UsAGE and QUIP, the serious usability problems are localized at the level of single actions and the results are shown in a graph of the action nodes. Each node stands for an action defined to be the user action, such as menu selection or clicking the Open button. The evaluator makes the decision on usability problems, based on the graphical data. UsAGE supports only the user interfaces created with the TAE Plus user interface management system, and QUIP requires the modification of the target application source code.

USINE (Lecerof & Paternò, 1998) also deploys automated log file analysis. Ivory and Hearst (2001) call this kind of approach the Hybrid Task-Based Pattern-Matching method. USINE is an automatic usability evaluation method for Java applications, enabling the use of the task models along with log files for analyzing empirical data. Tasks are decomposed into

Table 7. Review of Alternative Methods Based on General Dimensions.

Evaluation method	UI	Method class	Method type	Automation type	Effort level
UsAGE	WIMP	Performance measurement, Usability testing	Performance measurement, Log file analysis	Capture, Analysis	IF
QUIP	WIMP	Usability testing	Log file analysis	Analysis	IF
USINE	WIMP	Usability testing	Log file analysis	Analysis	IFM
RemUSINE	WIMP	Usability testing	Log file analysis	Analysis	IFM
WebRem-USINE	Web	Usability testing	Log file analysis	Analysis	IFM
D-TEO	Web	Usability testing	Performance measurement, Log file analysis	Analysis, (Critique)	IFM

Table 8. Review of Alternative Methods based on Specific Dimensions.

Evaluation method	Concepts and constructs	Criteria	Interpretation of criteria
UsAGE	Supports only the user interfaces created with <i>TAE Plus</i> user interface management system. Each <i>node</i> stands for the action that is defined to be the <i>user action</i> , such as menu selection.	Comparing event logs for expert user and novice user. Designer is also an expert user typically. In addition to a graph, the percentage of expert nodes matched to novice nodes, ratio of novice to expert nodes, and percentage of unmatched novice nodes are analyzed.	Two files ("expert" and "novice") are automatically compared by the tool and the results are shown graphically. Based on this, a usability analyst figures out where the usability problems exist.
QUIP	Requires the modification of the target application <i>source code</i> . Each <i>node</i> stands for the action that is defined to be the <i>user action</i> , such as menu selection or clicking the Open button.	Comparing task flows for UI designer and multiple test users. The trace of the UI designer represents the expected use. Quantitative time and trace-based information is encoded into directed graphs.	The evaluator makes the decisions based on the graphs by analyzing them manually.
USINE	Developed for usability evaluating of <i>WIMP interfaces</i> . Requires X Window environment. Requires comprehensive modeling and formalization of <i>user tasks</i> .	The accomplished tasks, the failed tasks, the never tried tasks, numerical and temporal information of the user errors, how long each task took to complete, the times for the abstract tasks, the errors occurred instances, task patterns, the test time, number of scrollbar movements, and the number of windows resized.	Evaluators make the decisions about how to improve user interface based on the simulator data. (The suggested interface changes are not drastic; e.g., there should be more difference between button and images.)

RemUSINE	<p>Developed for remote usability evaluating of graphical Java applications.</p> <p>Requires the comprehensive modeling and formalization of user tasks.</p>	<p>Tasks related criteria (single user session): Completed tasks, failed tasks, never tried tasks, errors, task patterns, tasks/time, errors/time, tasks/errors, tasks/completed.</p> <p>Tasks related criteria (groups of user sessions): Completed tasks, failed tasks, never tried tasks, errors, task patterns, tasks/time, errors/time, tasks/errors, tasks/completed.</p>	
WebRem-USINE	<p>Java based tool developed for <i>remote usability evaluation</i> of Web sites.</p> <p>Requires comprehensive modeling and formalization of <i>user tasks</i>.</p>	<p>Tasks related criteria (single user): Completed tasks, missed tasks, never performed tasks, errors, task patterns, error/time, task/time, tasks/errors, tasks/completed.</p> <p>Tasks related average times and standard deviations (number of users): Total time taken by user session, number of completed tasks, number of errors, number of scrollbar movements and change dimensions events.</p> <p>Pages related criteria (single user): Visited pages, never visited pages, scroll and resize, page patterns, download time, visit time, page/access, page/scroll/resize.</p> <p>Pages related criteria (number of users): Average number of accesses in to each page, average frequency of patterns, average downloading time, average visit time.</p>	<p>Evaluators make the decisions based on the rich simulator data (e.g. identify what tasks create problems and what tasks are efficiently performed).</p>
D-TEO	<p>Developed for locating <i>usability problems</i> in <i>directed information search</i> from Web sites.</p> <p>Is based on defining the <i>optimal paths</i> composed of <i>operations</i> needed to navigate from the entry Web pages to the terminal Web pages in order to find the target <i>information items</i>.</p>	<p>Operation-specific values (single participant): success value, duration time.</p> <p>Aggregated task-specific values (All tasks and participants): average success value for an operation, average duration for an operation, standard deviation of durations.</p> <p>Criterion for problematic Web pages: based on fractiles.</p>	<p>Problematic Web pages are identified with quantitative measures. Based on this information, usability evaluators can focus their attention on those pages.</p>

subtasks (sets of activities) that are related to each other within temporal relationships. The results derived by USINE include quite extensive numerical information about the tasks and subtasks, such as which tasks have been accomplished, which have failed, which have never been tried, user errors, and so on. Evaluators make decisions on how to improve the user interface based on the log data related to the tasks and subtasks. USINE does not enable comparing these results across study participants, so it is based on only task-related criteria of single user sessions. This means that it does not aggregate data, such as the average times of

subtasks across the participants or which tasks have been accomplished or failed across the participants. We suggest that a subtask-level comparison between the participants could bring essential knowledge to advance locating usability problems. USINE is also a rather laborious method, requiring the construction of comprehensive task models.

RemUSINE (Paternò & Ballardin, 1999, 2000), as an extension of USINE, enables capturing data remotely and comparing the results across the participants. RemUSINE employs task-related criteria for both single user sessions and groups of user sessions. In RemUSINE, evaluators make decisions regarding how to improve user interface based on the simulator and log data. RemUSINE, like USINE, was originally developed for evaluating Java applications and, as Paganelli and Paternò (2002) state, it is not suitable for evaluating how information is accessed through user interfaces in Web sites.

WebRemUSINE (Paganelli & Paternò, 2002) has its origin in USINE and RemUSINE. WebRemUSINE uses task-related criteria and page-related criteria for both single user sessions and the number of users. In WebRemUSINE, the evaluators make decisions on usability problems based on the rich simulator data. WebRemUSINE is a Java-based tool developed for remote usability evaluation of Web sites, and it requires the comprehensive modeling and formalization of user tasks. However, despite constructing a comprehensive task model and comparing the results across the participants by using rich quantitative data, neither RemUSINE nor WebRemUSINE provide an exact way to locate problematic subtasks. For example, even if the average times of the subtasks are known across the participants, the critical question about when the average time for a specific subtask is too long remains unanswered. We argue that there must be some rules, whether strict or heuristic, for helping determine some limits for crucial measures. Our suggestion is the use of the fractiles.

The D-TEO method is based on the use of two quantitative usability criteria, search success and search time, aiming at revealing how a user performs as a navigator in a web of hypertext. The former criterion is evaluated through operation-specific Success Value, calculated by the number of attempts required to find the optimal path. The latter criterion is expressed by duration time between the executions of two sequential operations. The D-TEO method is engineered to distinguish a part of a Web site that contains the most likely usability problems in directed searches. The results enable usability specialists to concentrate their efforts on making a deeper analysis of that particular area. The method does not aim at giving special guidance on the examination of what kinds of problems there are and how they are solved. Of course, if some Web page appears to be the one in which users tend to get lost, it is justifiable to expect that, for instance, the navigation, search, layout, typography, or content organization on that Web page is insufficiently designed. Similar goals are pursued by a number of interaction design patterns. What makes our method different when compared to other models is the use of heuristic rules for determining critical limits for the assessment of a certain Web page as problematic one. These rules are based on fractiles, which are selected in a situational manner. This semiautomated help is indicated in Table 7 by presenting (in parenthesis) a critique in the column of Automation type.

To summarize, the literature provides a large variety of methods for testing the usability of Web pages by observing users carrying out tasks, whether given wholly for, or as part of, their daily work. Our method differs favorably from them in the following aspects. First, the method is rather lightweight, meaning that instead of constructing a comprehensive task model, as required in USINE, RemUSINE, and WebRemUSINE, only the optimal paths for the test tasks have to be specified in D-TEO. Second, our method supports making decisions

on the limits of critical values. This is particularly beneficial in the situations where explicitly defined goals are not expressed in the quantitative measures. Finally, D-TEO distinguishes those parts of a Web site that are more problematic. As a result, scant resources can be focused on making a deeper evaluation of those areas only.

SUMMARY

The Internet holds an increasingly more important position in today's information dissemination. Diffusion of electronic commerce by enterprises and eServices provided by municipalities and government agencies have advanced the Internet as a daily means for both professionals and diverse audiences for interpersonal interaction and information searches. In attempting to serve these multiple audiences, Web site designers are challenged to meet the needs of a heterogeneous user population, from novices to heavy users, from persons having significant training for use to those with poor skills and low interest in information search. To ensure that all the people can find, with modest effort, what they are seeking, Web interfaces must meet high usability standards.

In this study, we have proposed a novel method, called D-TEO, which supports Website testing to find problematic Web pages. This semiautomated method is based on the analysis of interaction tasks in directed searches within the evaluated Web site. It provides a stepwise procedure that starts with defining the goals and user profiles and ends with analyzing the collected data and drawing conclusions. The method guides a test organizer in devising test tasks, decomposing them into elementary operations, and defining the optimal path for each task. Users are observed as they execute the test tasks and, for each operation, the time spent and the deviations from the optimal path are recorded. Using statistical methods, the collected data are analyzed to reveal which Web pages are problematic. This enables the test organizer to concentrate on more careful examination and analysis of particular small set of Web pages. Compared to most of the existing methods, D-TEO is lightweight because it does not require the comprehensive modeling and formalization of user tasks (as in Lecerof & Paternò, 1998), nor the existence of site maps. What also makes D-TEO beneficial is the support it provides regarding the situationally determined limits of critical measures for considering whether or not a Web page is included within a set of problematic Web pages.

The D-TEO method is still under research and development. At the moment, we are enhancing the method to encompass a wider variety of interaction tasks, not simply directed searches. The current procedure should be engineered toward a more fully automated mode, thus decreasing the need for human resources with required expertise in Web usability for the analysis aspect of the method. At the same time, it should be stated more clearly which type of methods—heuristic, pattern, or rule-based—are recommended as methods (e.g., Nielsen & Molich, 1990; Tidwell, 2005) applied prior to and/or following the deployment of D-TEO. In the future, we will consider how to integrate the method with qualitative methods in order to provide more flexibility for distinguishing problematic Web pages, analyzing them, and finding solutions to them. Until now, we have applied D-TEO in only small cases. To have stronger evidence of its feasibility, we will apply the method in a wider diversity of cases.

ENDNOTE

1. The Web site <http://www.jyvaskylanseutu.fi> was tested for the Jyväskylä region.

REFERENCES

- Baresi, L., Garcotto, P., & Paolini, P. (2001). Extending UML for modeling Web applications. In *Proceedings of 34th Annual Hawaii International Conference on Systems Sciences* (pp. 1285–1294). Washington, DC, USA: IEEE Computer Society.
- Batra, R., Bishu, R., & Donohue, B. (1993). Effects of hypertext topology on navigational performance. *Advances in Human Factors and Ergonomics*, 19, 175–180.
- Bernard, M. (2002). *Examining a metric for predicting the accessibility of information within hypertext structures*. Doctoral dissertation, Wichita, KS, USA: Wichita State University.
- Beyer, H., & Holtzblatt, K. (1998). *Contextual design: A customer-centered approach to system design*. San Diego, CA, USA: Academic Press.
- Blackmon, M. H., Polson, P. G., Kitajima, M., & Lewis, C. (2002). Cognitive walkthrough for the web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 463–470). New York: ACM.
- Canter, D., Rivers, R., & Storrs, G. (1985). Characterizing user navigation through complex data structures. *Behaviour and Information Technology*, 4, 93–102.
- Card, S. K., Pirulli, P., Wege, M. V. D., Morrison, J. B., Reeder, R. W., Schraedley, P. K., & Boshart, J. (2001). Information scent as a driver of web behavior graphs: Results of a protocol analysis method for web usability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 498–505). New York: ACM.
- Chi, E. H., Rosien, A., Supattanasiri, G., Williams, A., Royer, C., Chow, C., Robles, E., Dalal, B., Chen, J., & Cousins, S. (2003). The bloodhound project: Automating discovery of web usability issues using the InfoScent simulator. In *CHI '03: Proceedings of the SIGCHI Conference on Human Factors in Computing System* (pp. 505–512). New York: ACM.
- Cockton, G., & Woolrych, A. (2002). Sale must end: Should discount be cleared off HCI's shelves? *Interactions*, 9, 13–18.
- Constantine, L. L., & Lockwood, L. A. D. (1999). *Software for use: A practical guide to the models and methods of usage-centered design*. New York: ACM/Addison-Wesley Publishing Co.
- Cordella, A. (2007). E-government: Towards the e-bureaucratic form? *Journal of Information Technology*, 22, 265–274.
- De Angeli, A., Sutcliffe, A., & Hartmann, J. (2006). Interaction, usability and aesthetics: What influences users' preferences? In J. M. Carroll, S. Boedker, & J. Coughlin (Eds.), *Proceedings of the 6th ACM Conference on Designing Interactive Systems 2006 (DIS2006)* (pp. 271–280). New York: ACM.
- Dillon, A., McKnight, C., & Richardson J. (1990). Navigation in hypertext: A critical review of the concept. In D. Diaper (Ed.), *Human-Computer Interaction—INTERACT '90* (pp. 587–592). Amsterdam: Elsevier.
- Duh, H. B.-L., Tan, G. C. B., & Chen, V. H.-h. (2006). Usability evaluation for mobile device: A comparison of laboratory and field tests. In M. Nieminen & M. Røykkee (Eds.), *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services 2006 (MobileHCI '06)* (pp. 181–186). New York: ACM.
- Dumas, J. S., & Redish, J. C. (1993). *A practical guide to usability testing*. Westport, CT, USA: Greenwood Publishing Group Inc.

- Freeman, M., Norris, A., & Hyland, P. (2006). Usability of online grocery systems: A focus on errors. In T. Robertson (Ed.), *Proceedings of the 18th Australia Conference on Computer-Human Interaction: Design, Activities, Artefacts and Environments* (OZCHI Vol. 206; pp. 269–275). New York: ACM.
- Goldberg, J. H., Stimson, M. J., Lewenstein, M., Scott, N., & Wichansky, A. M. (2002). Eye tracking in web search tasks: Design implications. In *ETRA '02: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications* (pp. 51–58). New York: ACM.
- Gwizdka, J., & Spence, I. (2007). Implicit measures of lostness and success in web navigation. *Interacting with Computers*, 19, 357–369.
- Helfrich, B., & Landay, J. A. (1999). QUIP: Quantitative user interface profiling. Retrieved on July 4, 2008, from <http://www.helcorp.com/bhelfrich/helfrich99quip.pdf>
- Hennicker R., & Koch N. (2001). Systematic design of Web applications with UML. In K. Siau & T. Halpin (Eds.), *Unified modeling language systems analysis, design and development issues* (pp. 1–20). Hershey, PA, USA: IDEA Group Publishing.
- International Organization for Standardization [ISO]. (1998, March). *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*. (Standard No. 9241-11). Geneva, Switzerland: ISO.
- Isakowitz, T., Stohr, E., & Balasubramanian P. (1995). RMM: A design methodology for structured hypermedia design. *Communications of the ACM*, 38, 34–44.
- Ivory, M. Y., & Hearst, M. A. (2001). The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys*, 33, 470–516.
- Kim, H., & Hirtle, S. C. (1995). Spatial metaphors and disorientation in hypertext browsing. *Behaviour and Information Technology*, 14, 239–250.
- Lecerof, A., & Paternò F. (1998). Automatic support for usability evaluation. *IEEE Transactions on Software Engineering*, 24, 863–888.
- Lee, H., Lee, C., & Yoo, C. (1998). A scenario-based object-oriented methodology for developing hypermedia information systems. In *Proceedings of the 31st Annual Hawaii International Conference on System Sciences* (pp. 47–56). Washington, DC, USA: IEEE Computer Society.
- Lee, H., & Suh, W. (2001). A workflow-based methodology for developing hypermedia information systems. *Journal of Organizational Computing and Electronic Commerce*, 11, 77–106.
- Martin, R., & Weiss, S. (2006). Usability benchmarking case study: Media downloads via mobile phones in the US. In M. Nieminen & M. Røykkee (Eds.), *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services 2006* (MobileHCI'06; pp. 195–198). New York: ACM.
- Masemola, S. S., & De Villiers, M. R. (2006). Towards a framework for usability testing of interactive e-learning applications in cognitive domains, Illustrated by a case study. In J. Bishop & D. Kourie (Eds.), *Proceedings of the 2006 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries* (SAICSIT 2006; pp. 187–197). Somerset West, South Africa: South African Institute for Computer Scientists and Information Technologists.
- Matera, M., Rizzo, F., & Carughi, G. T. (2006). Web usability: Principles and evaluation methods. In E. Mendes & N. Mosley (Eds.), *Web engineering* (pp. 143–180). Berlin, Germany: Springer.
- Mayhew, D. J. (1999). *The usability engineering lifecycle: A practitioner's handbook for user interface design*. San Francisco: Morgan Kaufmann Publishers.
- Netcraft. (2009). January 2009 Web server survey. Retrieved January 20, 2009, from http://news.netcraft.com/archives/2009/01/16/january_2009_web_server_survey.html
- Nielsen, C. M., Overgaard, M., Pedersen, M. B., Stage, J., & Stenild, S. (2006). It's worth the hassle! The added value of evaluating the usability of mobile systems in the field. In A. Moerch, K. Morgan, T. Bratteteig, G. Ghosh, & D. Svanaes (Eds.), *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles 2006* (NordiCHI 2006; pp. 272–280). New York: ACM.
- Nielsen, J. (1993). *Usability engineering*. Boston: Academic Press.

- Nielsen J., & Mack R. (1994). *Usability inspection methods*. New York: Wiley.
- Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People 1990* (CHI 1990; pp. 249–256). New York: ACM.
- Norman K., & Chin J. (1988). The effect of tree structure on search performance in a hierarchical menu selection system. *Behaviour and Information Technology*, 7, 51–65.
- Paganelli, L., & Paternò, F. (2002). Intelligent analysis of user interactions with Web applications. In *Proceedings of the 7th International Conference on Intelligent User Interfaces* (ACM IUI'02; pp.111–118). New York: ACM.
- Paternò, F., & Ballardin, G. (1999). Model-aided remote usability evaluation. In A. Sasse & C. Johnson (Eds.), *Proceedings of the IFIP TC13 7th International Conference on Human-Computer Interaction (INTERACT '99)*; pp. 434–442). Amsterdam: IOS Press.
- Paternò, F., & Ballardin, G. (2000). RemUSINE: A bridge between empirical and model-based evaluation when evaluators and users are distant. *Interacting with Computers*, 13, 229–251.
- Paulson, L. (2005). Building rich Web applications with Ajax. *IEEE Computer*, 38, 14–17.
- Preciado, J., Lanaje, M., Sanchez, F., & Comai, S. (2005). Necessity of methodologies to model rich Internet applications. In *Proceedings of the 7th IEEE International Symposium on Web Site Evolution* (pp. 7–13). Washington, DC, USA: IEEE Computer Society.
- Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-computer interaction*. Wokingham, UK: Addison-Wesley.
- Rossi, G., & Schwabe, D. (2006). Model-based web application development. In E. Mendes & N. Mosley (Eds.), *Web engineering* (pp. 303–333). Berlin, Germany: Springer.
- Saariluoma, P., Parkkola, H., Honkaranta, A., Leppänen, M., & Lamminen, J. (2009). User psychology in interaction design: The role of design ontologies. In P. Saariluoma & H. Isomäki (Eds.), *Future interaction design II* (pp. 69–86). Berlin, Germany: Springer.
- Seffah, A., Donyaee, M., Kline, R. B., & Padda, H. K. (2006). Usability measurement and metrics: A consolidated model. *Software Quality Control*, 14, 159–178.
- Schneiderman, B. (1992). *Designing the user interface: Strategies for effective human-computer interaction*. Reading, MA, USA: Addison-Wesley.
- Schneiderman, B. (1998). *Designing the user interface: Strategies for effective human-computer interaction* (3rd ed.). Reading, MA, USA: Addison-Wesley.
- Scholtz, J., & Laskowski, S. (1998). Developing usability tools and techniques for designing and testing web sites. Retrieved on July 2, 2008, from http://www.itl.nist.gov/iad/IADpapers/hf_web.pdf
- Shackel, B. (1991). Usability: Context, framework, definition, design and evaluation. In B. Shackel & S. Richardson (Eds.), *Human factors for informatics usability* (pp. 21–38). Cambridge, MA, USA: Cambridge University Press.
- Smith, P. A. (1996). Towards a practical measure of hypertext usability. *Interacting with Computers*, 8, 365–381.
- Stone, D., Jarrett, C., Woodroffe, M., & Minocha, S. (2005). *User interface design and evaluation*. San Francisco: Morgan Kaufmann Publishers.
- Tidwell, J. (2005). *Designing interfaces: Patterns for effective interaction design*. Sebastopol, CA, USA: O'Reilly Media.
- Uehling, D. L., & Wolf, K. (1995). User action graphing effort (UsAGE). In *Proceedings of the Conference on Human Factors in Computing Systems* (pp. 290–291). New York: ACM.
- Van Duyn, D. K., Landay, J. A., & Hong, J. I. (2006). *The design of sites: Patterns for creating winning Web sites*. Upper Saddle River, NJ, USA: Prentice-Hall.
- Virzi, R. (1992). Refining the test phase of usability evaluation: How many subjects is enough? *Human Factors*, 34, 457–468.

Lamminen, Leppänen, Heikkinen, Kämäräinen, & Jokisuu

- White, H., Wright, T., & Chawner, B. (2006). Usability evaluation of library online catalogues. In W. Piekarski (Ed.), *Proceedings of the 7th Australasian User Interface Conference* (Vol. 50; AUI2006; pp. 69–72). New York: ACM.
- Wixon, D., & Wilson, C. (1997). The usability engineering framework for product design and evaluation. In M. Helander, T. Landauer, & P. Prabhu (Eds.) *Handbook of human-computer interaction* (pp. 653–688). Amsterdam: Elsevier.
- Wolf P., & Krcmar H. (2008). Needs driven design for eGovernment value webs. In the *Proceedings of the 41st Hawaii International Conference on System Sciences* (p 220). Washington, DC, USA: IEEE Computer Society.

Authors' Note

All correspondence should be addressed to:
Juha Lamminen
Agora Center
University of Jyväskylä
PL 35, University of Jyväskylä
40014 Finland
juha.e.lamminen@jyu.fi

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi

APPENDIX

The test tasks in the case study

1. You want to have a new hobby. What kinds of leisure activities can the Community College of the Jyväskylä region offer you in the spring 2007?
2. You promised your friend that you will take a trip together to Hankasalmi for one day in July. What kinds of events take place there on 21 July, 2007?"
3. You are going to spend the weekend in Hankasalmi and you will need a place to stay overnight. What kind of camping sites are there in Hankasalmi?"
4. Assume that you are living in Petäjävesi. What kind of sport can you exercise in the local sport clubs?
5. There is one ringette team in the Jyväskylä region. What is the name of this team?
6. The southern part of Uurainen is bit more than 20 km from the center of Jyväskylä. Which kinds of properties are there for sale in Uurainen?
7. You would like to contact the project manager of the Health and Special Sport project (TERLI). Find information about the project on the Web site of the Jyväskylä region.
8. Try to find out when the Web site of the Jyväskylä Region was opened.