



TAVOITEPOHJAINEN KÄYTTÖLIITTYMÄSUUNNITTELU

**Käytettävyyden kehittäminen GUIDe-
prosessimallilla**

Erkka Kinnunen

**Opinnäytetyö
Marraskuu 2008**

Liiketalous



**JYVÄSKYLÄN
AMMATTIKORKEAKOULU**

Tekijä(t) KINNUNEN, Erkki	Julkaisun laji Opinnäytetyö			
	Sivumäärä 62	Julkaisun kieli Suomi		
	Luottamuksellisuus <input type="checkbox"/> Salainen _____ saakka			
Työn nimi TAVOITEPOHJAINEN KÄYTTÖLIITTYMÄSUUNNITTELU Käytettävyyden kehittäminen GUIDe-prosessimallilla				
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma				
Työn ohjaaja(t) KIVIAHO, Niko				
Toimeksiantaja(t) Controla Oy				
<p>Tiivistelmä</p> <p>Käytettävyydellä on merkittävä vaikutus tietoteknisten laitteiden ja ohjelmistojen hyödyllisyyteen. Järjestelmän on oltava helppokäyttöinen, tehokas ja miellyttävä, jotta tietotekniikkainvestoinneilla tavoiteltavat hyödyt voidaan saavuttaa. Käytettävyyden kehittäminen on laadunvarmistuskeino, jonka avulla pyritään tehostamaan järjestelmän käyttöä ja pienentämään uuteen järjestelmään kohdistuvaa muutosvastarintaa.</p> <p>Opinnäytetyön lähtökohtana oli Controla Oy:n toimeksiannosta toteutettu Liikkuvan työn ohjausjärjestelmä TARMO 2.0 -käyttöliittymäsuunnitteluprojekti. Kehitystutkimuksen tavoitteena oli varmistaa järjestelmän selkeys ja helppokäyttöisyys jo määrittelyvaiheessa tavoitepohjaisen käyttöliittymäsuunnittelun GUIDe-prosessimallin avulla. Kehitystutkimuksen taustaksi selvitettiin, mitä on käytettävyys ja millä menetelmillä sitä voidaan kehittää. Käytettävyyden vaikutuksia tarkasteltiin asiakasorganisaation ja ohjelmistotalon liiketoiminnan kannalta.</p> <p>Suunnitteluprosessi aloitettiin tutustumalla asiakasyrityksen työntekijöihin ja heidän työnkulkuihinsa. Kentätutkimuksella selvitettyjen tietojen pohjalta laadittiin työntekijöiden tärkeimpiä työtehtäviä ja tavoitteita kuvaavat tavoitepohjaiset käyttötapaukset. Määriteltyjä käyttötilanteita simuloimalla suunniteltiin mahdollisimman hyvin asiakasyrityksen liiketoimintaa ja käyttäjien työstä nousevia tarpeita tukeva käyttöliittymäratkaisu. Käyttöliittymäprototyyppi mahdollisti järjestelmän testaamisen loppukäyttäjien oikeilla työtehtävillä jo määrittelyvaiheessa.</p> <p>Tuotettu käyttöliittymäkuvaus toimi lähtökohtana projektin toteutusvaiheelle. Suunnittelussa käytettyjä menetelmiä arvioitiin toimeksiantajan näkökulmasta, joten tutkimuksen tietoja voidaan hyödyntää Controlan uusien vaatimusmäärittelyprosessien kehittämisessä.</p>				
Avainsanat (asiasanat) käytettävyys, käyttäjätutkimus, käyttäjakeskeinen suunnittelu, GUIDe, ohjelmistotuotanto, vaatimusten määrittely, kenttätyönohjausjärjestelmä				
Muut tiedot				

Author(s) KINNUNEN, Erkka	Type of Publication Bachelor's Thesis			
	Pages 62	Language Finnish		
	Confidential <input type="checkbox"/> Until _____			
Title GOAL-DERIVED DESIGN Usability engineering with the GUIDe process model				
Degree Programme Business Information Systems				
Tutor(s) KIVIAHO, Niko				
Assigned by Controla Oy Ltd				
<p>Abstract</p> <p>Usability has a significant impact on usefulness of data processing software and equipment. In order to achieve the advantages from investing in information technology, the system must be effective, satisfying and easy to use. Usability engineering is a quality assurance method which improves technology acceptance in initialization phase and makes the use of information systems more effective.</p> <p>The basis for the thesis was the user interface design of Liikkuvan työn ohjausjärjestelmä TARMO 2.0 on assignment of Controla Oy Ltd. The aim of the development research was to ensure that the complete software would be explicit and user-friendly by using goal-derived design process GUIDe. Usability as a concept and its research and development methods were examined as the theoretical background of the study. The effects of usability engineering on business are discussed both from the viewpoint of clients and software houses.</p> <p>The process began by observing the client's employees and their work in order to find out the user requirements. The most essential tasks were described as goal-based use cases. By simulating the defined use cases, the user interface was designed to suit the needs of the users and improve the client organization's productivity. The user interface prototype enabled the system usability evaluation already in the definition phase and also that the testing could be performed by its actual users in authentic working situations.</p> <p>The user interface specification prepared the way for the system production. Development methods were examined from the assigning company's point of view and therefore, the information presented in this study is valuable in future system development projects.</p>				
Keywords usability, user study, user-centered design, GUIDe, software engineering, requirements engineering, field force management system				
Miscellaneous				

SISÄLTÖ

1 JOHDANTO	3
2 TUTKIMUKSEN TAUSTA JA TAVOITTEET	4
2.1 Toimeksiantajan esittely	4
2.2 Liikkuvan työn ohjausjärjestelmä TARMO	5
2.3 Toimeksiantajan nykyinen määrittelyprosessi	6
2.4 Tutkimuksen toteutus ja tutkimuskysymykset.....	7
3 KÄYTETTÄVYYS JA KÄYTTÖLIITTYMÄ.....	10
3.1 Käytettävyyden määritelmä	11
3.2 Käytettävyyden subjektiivisuus	14
4 KÄYTETTÄVYYDEN KEHITTÄMINEN.....	16
4.1 Käyttäjäkeskeisen suunnittelun periaatteet.....	17
4.2 Tavoitepohjainen käyttöliittymäsuunnittelu	19
4.3 Miksi käytettävyyttä kannattaa kehittää?	21
4.4 Käytettävyyden kehittämisen edellytykset.....	23
5 TARMO 2.0 -KÄYTTÖLIITTYMÄSUUNNITTELU	25
5.1 Tausta ja tavoitteet	25
5.2 Käyttäjätutkimus.....	26
5.3 Tutkimusaineiston analysointi	30
5.4 Käyttöliittymäsuunnittelu.....	36
5.5 Käyttöliittymäratkaisun arviointi.....	42

5.6 Projektin jatkotoimenpiteet.....	45
6 JOHTOPÄÄTÖKSET JA POHDINTA.....	47
LÄHTEET.....	53
LIITTEET.....	56
Liite 1. Käyttöliittymän UED-kaavio.....	56
Liite 2. Käyttöliittymäkuvat	57

KUVIOT

KUVIO 1. Yrityksen tietoteknisten apuvälineiden kokonaiskustannukset	16
KUVIO 2. GUIDe-mallin keskeisimmät vaiheet	20
KUVIO 3. Kertaluontoisen työtehtävän käyttötapaus.....	34
KUVIO 4. Viikoittain toistuvan työtehtävän käyttötapaus	35
KUVIO 5. Kertaluontoisen työtehtävän määrittämisen käyttöliittymäratkaisu.....	39
KUVIO 6. Viikoittain toistuvan työtehtävän määrittämisen käyttöliittymäratkaisu	41

1 JOHDANTO

Tietoteknisiä sovelluksia hyödynnetään nykyään lähes kaikilla liiketoiminnan sektoreilla. Kehitettävien tietojärjestelmien odotetaan kasvattavan liiketoiminnan tuottavuutta ja tehostavan työskentelyä. Aina yritysten tietotekniikkainvestoinnit eivät ole kuitenkaan johtaneet halutun suuruisiin vaikutuksiin. Kuopion yliopiston ja Savonia-ammattikorkeakoulun tekemän selvityksen mukaan organisaation liiketoimintaprosesseja onnistuneesti tukevien järjestelmien suunnitteleminen edellyttää käyttäjien työnkulkujen tuntemista ja koko työtoiminnan yhtäaikaista kehittämistä (Toivanen ym. 2007, 11 - 12). Järjestelmän soveltuvuuden lisäksi tietojenkäsittelytoiminnan onnistumiseen vaikuttaa ohjelmiston käytettävyys. Käytettävyydeltään korkeatasoinen järjestelmä helpottaa sovelluksen käyttöönottovaihetta, tehostaa työskentelyä ja pitää työntekijät tyytyväisenä.

Käyttäjän aktiiviseen huomioimiseen ja käytettävyyden systemaattiseen kehittämiseen perustuu käyttäjäkeskeisen tuotekehityksen suunnitteluprosessi, joka on määritelty standardissa ISO 13407. Prosessimallin tavoitteena on selvittää käyttäjien todelliset tavoitteet ja luonnolliset toimintatavat, joiden pohjalta pyritään luomaan mahdollisimman hyvin käyttäjän työtä ja työstä nousevia tarpeita tukeva ohjelmisto (SFS-EN ISO 13407, 12). Näitä periaatteita noudattaa myös Helsingin yliopistossa kehitetty GUIDe-prosessimalli (*Goals – User Interface Design – Implementation*).

Opinnäytetyössä tarkastellaan, miten tavoitepohjaisen käyttöliittymäsuunnittelun GUIDe-prosessimallia voidaan hyödyntää työn toimeksiantajan Controla Oy:n Liikuvan työn ohjausjärjestelmä TARMO 2.0 -kehitystyössä. Kenttätyön ohjaukseen tarkoitetun järjestelmän nykyinen versio sisältää lukuisia ylimääräisiä toimintoja ja sitä määrittävät monet toteutustekniset ratkaisut, jotka heikentävät sovelluksen käytettävyyttä. Järjestelmän helppokäyttöisyyteen ja tehokkuuteen olisi kuitenkin kiinnitettävä huomiota, sillä suurimmat yrityksen tietotekniikkainvestointeihin liittyvät kustannukset koituvat työntekijän varsinaiseen työtoimintaan kuulumattomien, mutta liiketoiminnan kannalta olennaisten sovelluksien käytöstä.

2 TUTKIMUKSEN TAUSTA JA TAVOITTEET

Tässä luvussa esitellään opinnäytetyön toimeksiantaja Controla Oy ja tarkastellaan yrityksen tarjoamia esitutkimus- ja vaatimusmäärittelypalveluita. Toimeksiantajan lisäksi esitellään kehitystutkimuksen kohteena oleva Liikkuvan työn ohjausjärjestelmä TARMO ja selvitetään järjestelmän ensimmäisen version kehitystyön lähtökohtia sekä tavoitteita. Luvussa 2.3 tarkastellaan toimeksiantajan nykyistä vaatimusmäärittelyprosessia ja arvioidaan käyttäjän tarpeiden selvittämisen sekä järjestelmävaatimuksien kuvaamisen menetelmiä, joita kehitysprojekteissa on tähän asti käytetty. Näille menetelmille pyritään opinnäytetyön aikana löytämään käyttäjäkeskeisempiä vaihtoehtoja. Opinnäytetyön tavoitteita ja toteutusta sekä työlle asetettavia tutkimuskysymyksiä tarkastellaan luvussa 2.4.

2.1 Toimeksiantajan esittely

Aluksi tietojärjestelmät olivat lähinnä laskennallisia sovelluksia tai ne oli kehitetty automatisoimaan jo olemassa olevia manuaalisia järjestelmiä. Järjestelmien kehittäjät ja käyttäjät olivat monesti samoja henkilöitä, joten käyttäjien tarpeisiin ja niiden mallintamiseen ei juurikaan kiinnitetty huomiota. Nykyisin tietojärjestelmät ovat kuitenkin erittäin monimuotoisia ja asiakkaiden tarpeisiin räätälöityjä. (Pohjonen 2002, 16.) Tämän vuoksi myös laajat, koko organisaation liiketoiminnan käsittävät järjestelmät on tavallista tilata ulkopuoliselta toimittajalta (Wiio 2004, 67). Sovelluksen toimittajan haasteena on asiakkaan työtoimintaa aidosti tukevan järjestelmän kehittäminen, mikä vaatii asiakasyrityksen liiketoiminnan ja toimintamallien tuntemista (Pohjonen 2002, 16).

Tutkimuksen toimeksiantaja on jyvaskyläläinen IT-alan konsultointipalveluja tarjoava Controla Oy. Pääasiassa pienille ja keskisuurille yrityksille kohdennettujen palvelujen tarkoituksena on selvittää esitutkimuksen ja vaatimusmäärittelyn keinoin, millaiset tietotekniset ratkaisut tukevat yrityksen kasvua tai tehostavat liiketoimintaa. ICT-esitutkimuksen avulla kartoitetaan yrityksen käytössä olevat IT-ratkaisut, kuten ohjelmistot ja tietojärjestelmät, arvioidaan niiden liiketoiminnalle tuomia hyötyjä sekä sel-

vitetään mahdolliset kehitystarpeet. Kun tarve yrityksen liiketoiminnan kehittämiseksi tunnustetaan, tehdään vaatimusmäärittely. (Controla Oy 2008.)

Vaatimusmäärittelyn tarkoituksena on selvittää tarvittavat laitteistot ja kehitettävälle järjestelmälle asetetut toiveet sekä tavoitteet, joiden pohjalta laaditaan suunnitelma siitä, millaista järjestelmää oikein ollaan kehittämässä. Vaatimusmäärittelyn rungon muodostavat käyttäjätarpeet, joilla tarkoitetaan käyttäjän työssä tarvitsemia toimintoja ja tietosisältöjä. Asiakkaan asettamien toiminnallisten vaatimusten lisäksi sovelluksen muotoon vaikuttavat asiakkaan eri sidosryhmien tarpeet sekä ei-toiminnalliset vaatimukset, jotka määrittelevät, minkälaisin reunaehdoin järjestelmä toteuttaa toiminnalliset vaatimuksensa. Käytännössä ei-toiminnallisilla vaatimuksilla tarkoitetaan ominaisuuksia, kuten kapasiteetti ja vasteajat sekä käyttöjärjestelmiä, joita ohjelmiston tulisi tukea. (Pohjonen 2002, 28.) Esimerkiksi terveydenhoitoalan järjestelmiä koskevat myös lukuisat lakisääteiset velvoitteet, jotka käsittelevät muun muassa potilastietojen kirjaamista ja järjestelmien tietoturvallisuutta (Malli terveydenhuollon toimintayksikön tietosuojaohjeiden rungoksi 2000, 2). On erityisen tärkeää, että vaatimusmäärittelyn tuloksena asiakkaan ja toimittajan näkemykset toteutettavasta järjestelmästä ovat yhteneväiset, sillä määrittelytyön pohjalta laadittu vaatimusmäärittelydokumentti toimii myös osapuolten välisenä tuotteen toimitussopimuksena (Wiio 2004, 67).

2.2 Liikkuvan työn ohjausjärjestelmä TARMO

Liikkuvan työn ohjausjärjestelmä TARMO on Controlan tarjoama, erityisesti hoiva-alan yrityksille suunnattu kenttätyönohjauksen ja raportoinnin työkalu. Järjestelmän ensimmäinen versio valmistui vuonna 2007 yhteistyössä kahden muun yrityksen kanssa. (Controla Oy 2008.) Kehitystyön pohjaksi tehtiin vaatimusmäärittelyprojekti, jonka yhteydessä tutkittiin kotisairaanhoidon- ja kotipalvelualalla toimivan asiakasyrityksen työnohjaukseen liittyvää toimintaprosessia ja sen kehittämisalueita. Käyttäjätutkimus suoritettiin haastattelemalla yrityksen työnohjauksesta ja raportoinnista vastaavia henkilöitä sekä yrityksen toimitusjohtajaa. Tutkimustyön avulla saatiin selvitettyä, että yrityksen sen hetkisen toimintaprosessin suurimmat ongelmat liittyivät työaikakirjanpitoon ja laskutusta sekä palkanlaskentaa varten tarvittavien tuntitietojen kirjaamiseen. Puutteellinen ja työläs asiakaskäyntien dokumentointikäytäntö johti hel-

posti siihen, että merkittävä osa tehdyistä työtehtävistä ja tarjottavista palveluista jäi asiakkailta laskuttamatta. (Kotihoidon langaton työnohjaus- ja raportointijärjestelmä 2007.)

Liikkuvan työn ohjausjärjestelmä TARMO pyrkii tehostamaan yrityksen liiketoimintaa automatisoimalla työtehtävien määräämiseen, tuntien kirjaamiseen ja laskutukseen sekä palkanlaskentaan liittyviä prosesseja. Suullisesti tai puhelimitse tapahtuvan työtehtävien jakamisen sijaan työnohjaaja antaa työmääräykset www-sovelluksen pohjautuvan järjestelmän avulla. Kentällä olevat työntekijät vastaanottavat tehtäviä ja kuittaavat ne tehdyiksi mobiilia päätelaitetta käyttäen. Työtehtävistä saadut tiedot kirjautuvat tietokantaan ja niistä voidaan laatia raportteja esimerkiksi laskutusta ja palkanlaskentaa varten. Järjestelmän avulla myös asiakas- ja työntekijärekisterit pysyvät ajan tasalla. (Liikkuvan työn ohjausjärjestelmä TARMO 2008.) Työtehtävien automatisointi säästää turhiin työtehtäviin kuluvaan aikaa ja vähentää tietojen ylimääräisen käsittelyn takia syntyviä virheitä.

2.3 Toimeksiantajan nykyinen määrittelyprosessi

Hoffman ja Lehner (2001) pitävät vaatimusmäärittelyä ohjelmistoprojektin kriittisimpänä vaiheena ja suurimpana yksittäisenä syynä kehitysprojektien epäonnistumiseen (Alanne 2002, 28). Puutteellisesta vaatimusmäärittelystä johtuvat ongelmat voivat johtaa siihen, että lopullinen ohjelmisto ei täysin sovellu sille tarkoitettuun tehtävään tai on pahimmillaan täysin käyttökelvoton (Makkonen 2006, 45).

Menestyksekkään kehitysprojektin edellytyksenä pidetään käyttäjien tarpeiden huolellista kartoittamista ja työnkulkujen ymmärtämistä (Kujala 2006, 9). Toimeksiantajan nykyisen prosessin mukainen käyttäjävaatimusten selvittäminen tapahtuu pääasiassa asiakkaan ja eri sidosryhmien kanssa käytävien keskustelujen avulla. Vaatimusten esiin saaminen haastattelemalla voi olla työlästä, koska haastateltava ei aina tiedä mitä kehitettävältä järjestelmältä haluaa, tai hänen on vaikea kuvailla riittävän tarkasti kuinka työtehtävät nykyisin suoritetaan. Käyttäjältä saatujen tietojen pohjalta laaditaan järjestelmän rakennetta kuvaava vaatimusmäärittelydokumentti. Järjestelmän toimintoja ja sille asetettuja rajoituksia kuvataan tavallisesti skenaariomuotoisten käyt-

tötapauskuvausten sekä erilaisten UML-standardin (*Unified Modeling Language*) mukaisten kaavioiden avulla. (Kotihoidon langaton työnohjaus- ja raportointijärjestelmä 2007.)

Skenaariomuotoisiin kuvauksiin pohjautuvan vaatimusmäärittelydokumentin ongelmana on se, että sitä voidaan tulkita monin eri tavoin, mikä lisää väärinymmärryksiä ja vaikeuttaa kehitettävän järjestelmän ominaisuuksista käytävää keskustelua (Pohjonen 2002, 29). Järjestelmän rakennetta ja yksittäisten toimintojen suorittamista voidaan havainnollistaa jonkin verran käyttötapauskaavioiden (*use cases*), sekvenssikaavioiden (*sequence diagram*) ja luokkakaavioiden (*class diagram*) avulla (mts. 153). Kaaviot kuvaavat kuitenkin enemmän ohjelmiston toimintaa kuin sitä ongelmaa, joka käyttäjän pitäisi työssään ratkaista, eivätkä näin ollen mahdollista ohjelmiston toimivuuden testaamista aidoissa käyttötilanteissa. (Hanhisalo 2007, 7.)

Toimeksiantajan käyttämän määrittelyprosessin vakavimmat ongelmat liittyvät siis oikeiden vaatimusten löytämiseen, niiden arvioimiseen ja lopulta testaamiseen käyttäjän todellisten työnkulkujen avulla. Helsingin yliopiston apulaisprofessorin Sari A. Laakson mukaan edellä mainittujen ongelmien lisäksi monien nykyisin yleisesti käytettyjen prosessimallien heikko kohta on riittämätön tai kokonaan puuttuva käyttöliittymäsuunnitteluvaihe. Esimerkiksi vesiputousmallin mukainen vaatimusmäärittely ei sisällä johdonmukaista menettelyä käyttöliittymään kohdistuvien vaatimusten määrittelemiseksi, vaan siinä keskitytään kuvaamaan järjestelmän arkkitehtuuria ja toimintalogiikkaa. Käytettävyyteen liittyvät vaatimukset esitetään muiden vaatimuksien yhteydessä ja sovelluksen käyttöliittymä muotoutuu vasta suunnitteluvaiheessa, jolloin projektin aiemmissa vaiheissa tehdyt päätökset järjestelmän toteutusratkaisusta määrittävät myös käyttöliittymän toimintoja ja heikentävät ohjelmiston käytettävyyttä. (Laakso & Laakso 2004, 2.)

2.4 Tutkimuksen toteutus ja tutkimuskysymykset

Tutkimuksen lähtökohtana on Liikkuvan työn ohjausjärjestelmä TARMOn uuden version käyttöliittymäsuunnittelu. Kehitystyön apuna käytetään tavoitepohjaisen käyttöliittymäsuunnittelun GUIDe-prosessimallia, jonka tarkoituksena on varmistaa loppu-

tuotteen helppokäyttöisyys ja hyödyllisyys jo kehitysprojektin määrittelyvaiheessa. Prosessimallin avulla pyritään myös välttämään edellisessä luvussa mainittuja perinteisen vaatimusmäärittelyn ongelmia.

Empiirisen tutkimuksen tavoitteena on kuvata työnohjausjärjestelmän käyttöliittymän systemaattista suunnitteluprosessia ja tuottaa käytettävyydeltään korkeatasoinen ja mahdollisimman hyvin asiakasyrityksen liiketoimintaa sekä käyttäjien työstä nousevia tarpeita tukeva käyttöliittymäratkaisu. GUIDe-prosessimallin menetelmien mukaisesti tuotettu työnohjausjärjestelmän käyttöliittymäkuvaus toimii syötteenä TARMO 2.0 -kehitysprojektin toteutusvaiheelle. Käyttöliittymäkuvauksen lisäksi työssä pyritään tuottamaan tietoa käytettävyyden kehittämisen periaatteista ja havainnollistamaan GUIDe-prosessimallin mukaista käyttöliittymäsuunnittelun etenemistä työnohjausjärjestelmän käyttäjien keskeisimpien työnkulkujen avulla. Kehitystyössä käytettäviä menetelmiä arvioidaan toimeksiantajan liiketoiminnan näkökulmasta, joten tietoa voidaan hyödyntää myös Controlan uusien vaatimusmäärittely- ja käytettävyyssuunnitteluprosessien luomisessa.

Kehitystutkimuksen teoriataustaksi pyritään selvittämään, mitä on käytettävyys, ja tarkastellaan koko ohjelmistotuotantoprojektin elinkaaren kestävän käytettävyyden kehittämisen menetelmiä. Käytettävyytyöhön panostamisen vaikutuksia tarkastellaan sekä asiakasyrityksen että ohjelmistotalon liiketoiminnan kannalta. Työn ulkopuolelle rajataan käytettävyyden visuaalinen puoli, sillä hyvä käytettävyys on muutakin kuin graafisesti näyttävä käyttöliittymä. Tutkimuksessa ei käsitellä myöskään olemassa olevien järjestelmien käytettävyyden arvioinnin menetelmiä, sillä ne tapahtuvat systemaattisen käytettävyyden kehittämisen kannalta liian myöhään. Tutkimuksen tavoitteista voidaan johtaa seuraavat kolme tutkimuskysymystä:

1. Mitä tarkoitetaan käytettävyydellä ja mikä on sen merkitys tietoteknisissä sovelluksissa?
2. Millä menetelmillä kehitettävän järjestelmän käytettävyys voidaan varmistaa?
3. Miten käyttöliittymälähtöiseen määrittelyyn perustuva prosessimalli soveltuu käytettäväksi toimeksiantajan vaatimusmäärittelyprojekteissa?

Käytettävyyttä ja sen merkitystä tarkastellaan käytettävyytystutkijoiden asettamien määritelmien sekä eri käytettävyyssattribuuttien kautta. Käytettävyyden vaikutuksia pyritään käsittelemään erityisesti sekä asiakasorganisaation että ohjelmistotalon liiketoiminnan kannalta. Luvuissa 3 ja 4 käytettävyyden ilmiötä käsitellään kvalitatiivisen tutkimuksen keinoin. Laadullista tutkimusta varten kerätty lähdeaineisto käsittelee muun muassa ihmisen ja tietokoneen välistä vuorovaikutusta sekä käytettävyyden psykologiaa. Muita tutkimuksen kannalta tärkeitä lähteitä ovat tutkimukset eri ohjelmistotalojen käytettävyyden kehittämisprosesseista ja GUIDe-prosessimalliin liittyvät artikkelit sekä Liikkuvan työn ohjausjärjestelmä TARMOn nykyisin käytössä olevan version dokumentaatiot. Tuloksien pohjalta vastataan ensimmäiseen ja toiseen tutkimuskysymykseen. Teoriaosuus luo samalla pohjan luvussa 5 kuvatulle TARMO 2.0 -kehitystutkimukselle, jonka antamien kokemusten ja käyttöliittymälähtöistä määrittelyä käsittelevien kirjallisuuslähteiden pohjalta vastataan kolmanteen tutkimuskysymykseen.

3 KÄYTETTÄVYYS JA KÄYTTÖLIITTYMÄ

Lähes jokaiseen laitteeseen tai ohjelmistoon voidaan liittää termi käytettävyys. Esi-merkiksi tuotantoympäristössä käytettävyys voi merkitä tuotteen saavutettavuutta (*availability*) eli palvelun tai järjestelmän tiettyjen resurssien puitteissa toteutuvaa toimivuusastetta. Yleisesti tietotekniikassa käytettävyydellä (*usability*) tarkoitetaan kuitenkin sitä, kuinka hyvin jonkin laitteen tai ohjelmiston toimintoja voidaan käyttää haluttuun tarkoitukseen (Kuutti 2003, 13). Myös tässä työssä käytettävyydestä puhuttaessa viitataan termin jälkimmäiseen merkitykseen.

Tietotekniikan historian alkuaikoina käytettävyyteen ei kiinnitetty juurikaan huomiota, sillä, kuten aiemmin mainittiin, sovellusten pääasiallisia käyttäjiä olivat joko ohjelmoijat itse tai muut alan asiantuntijat (Kuutti 2003, 140). Vasta tietotekniikan arkipäiväistymisen ja www-sivustojen yleistymisen myötä on käytettävyyden rooli kasvanut ja siitä on alettu puhua myös viihde-elektroniikan ja muiden arkipäiväisten esineiden yhteydessä. Käytettävyydestä puhuttaessa törmätään myös termiin käyttöliittymä (*user interface*). ATK-sanakirjan (2008) antaman määritelmän mukaan käyttöliittymällä tarkoitetaan välineitä ja toimintoja, joilla käyttäjä on vuorovaikutuksessa ohjelman tai järjestelmän kanssa. Käyttöliittymällä ei kuitenkaan tarkoiteta ainoastaan tietoteknisten tuotteiden ikkunoita ja valikoita tai esimerkiksi internetsivujen sisällön asemointia. Myös aivan tavallisilla arjen esineillä, kuten ovella tai vaikkapa hanalla, on käyttöliittymä, joka toimii vuorovaikutusrajapintana käyttäjän ja tuotteen välillä antaen käyttäjälle tietoa tuotteen toiminnasta. (Kuutti 2003, 13.) Käyttäjän näkökulmasta käyttöliittymä on yhtä kuin koko järjestelmä, minkä vuoksi sen toimivuuden merkitystä ei voida yliarvioida. Käyttöliittymän avulla käyttäjä muodostaa kuvan järjestelmästä. Se myös auttaa käyttäjää navigoimaan järjestelmässä ja antaa työkalut ajatteluun, päätöksentekoon ja tehtävien suorittamiseen. (Käyttöliittymän määrittelyä 2005.)

3.1 Käytettävyyden määritelmä

Monet tutkijat ovat tarkastelleet käytettävyyttä ja määritelleet käytettävyyden käsitettä kukin omista lähtökohdistaan, minkä vuoksi ei ole olemassa vain yhtä yleisesti hyväksyttyä käytettävyyden määritelmää. Yleisesti ottaen käytettävyydellä kuitenkin tarkoitetaan tuotteen laatuominaisuutta, joka kertoo, kuinka hyvin tuote soveltuu sille suunniteltuun käyttötarkoitukseen ja kuinka helppoa ja tehokasta sitä on käyttää. Ominaisuutena käytettävyys on abstrakti ja vaikeasti mitattavissa. Tästä syystä käsitettä on purettu konkreettisemmiksi osakokonaisuuksiksi.

Kansainvälinen standardointijärjestö ISO on esittänyt yleisesti hyväksytyn käytettävyyden määritelmän standardissaan ISO 9241-11 (Näyttöpäätteillä tehtävän toimiston ergonomiset vaatimukset. Osa 11: Käytettävyyden määrittely ja arviointi). Standardi määrittelee käytettävyyden tuottavuutena, tehokkuutena ja tyytyväisyytenä, jolla määritellyt käyttäjät saavuttavat haluamansa tavoitteet tietyssä käyttöympäristössä (SFS-EN ISO 9241-11). Käytettävyystudkija Jakob Nielsen on esittänyt toisen tunnetun ja yleisellä tasolla ISO-standardin kanssa yhtenevän määritelmän. Nielsen (1993, 26) asettaa käytettävyyden osaksi tuotteen käyttökelpoisuutta ja jakaa sen viiteen eri osatekijään eli attribuuttiin. Nämä osatekijät ovat opittavuus (*learnability*), tehokkuus (*efficiency*), muistettavuus (*memorability*), virheettömyys (*errors*) ja miellyttävyys (*satisfaction*). Seuraavassa tarkastellaan Nielsenin ja Antti Wiion asettamien mittareiden kautta käytettävyyden vaikutuksia työntekijöiden sekä koko organisaation toimintaan.

1. Opittavuus

Opittavuus eli intuitiivisuus on käytettävyyden arvioinnin keskeisimpiä mittareita (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 227). Helppokäyttöinen järjestelmä, jota käyttäjät pystyvät kohtuullisen opettelun jälkeen hyödyntämään työnteossa mahdollisimman tehokkaasti, vähentää koulutukseen ja tukipalveluihin tarvittavia resursseja. Parhaimmillaan intuitiivista tuotetta pystyy käyttämään sillä osaamisella ja kokemuksella, mikä käyttäjällä on entuudestaan. Opittava järjestelmä, jossa asiat on esitetty käyttäjän ymmärtämällä tavalla, vaikuttaa lisäksi positiivisesti työskentelymotivaatioon ja vähentää syntyvien virhetilanteiden määrää. (Nielsen 1993, 28 - 29; Onko käytettävyyteen panostamisessa järkeä? 2007.)

2. Tehokkuus

Tehokkuudella tarkoitetaan tuotteen opettelun jälkeen saavutettavaa työskentelytehoa (Nielsen 1993, 26). Käytettävyydeltään tehokkaassa järjestelmässä tärkeimmät toiminnot ovat nopeasti saatavilla ja harvemmin käytettävät löytyvät vaivattomasti. Myös suoritettavat tehtävät hoituvat nopeasti ja tarkasti ilman ylimääräisiä välivaiheita. (Parkkinen 2001.)

Tehokkuuden merkitys korostuu usean käyttäjän organisaatioissa, jolloin järjestelmän vaikutus liiketoiminnan tuottavuuteen voi olla valtava. Sadan hengen yrityksessä pienestä käytettävyysongelma johtuva muutaman minuutin päivittäinen työajan menetys voi vuositasolla koitua kymmenien tuhansien eurojen kustannukseksi. Organisaation operatiivisen tehokkuuden alenemisen lisäksi huono käytettävyys voi hankaloittaa päätöksentekoa. Mikäli tarvittavan tiedon etsiminen järjestelmästä on esimerkiksi asiakaspalvelutilanteessa liian hidasta tai vaivalloista, käyttäjä voi joutua tekemään mahdollisesti myös yrityksen talouteen vaikuttavia riskipäätöksiä omaan muihin turvautuen. (Sinkkonen 2006; Onko käytettävyyteen panostamisessa järkeä? 2007.)

3. Muistettavuus

Muistettavuudella mitataan, kuinka helppo tuotteen käyttö on aloittaa tauon jälkeen uudestaan ja kuinka nopeasti käyttäjä pystyy saavuttamaan taukoa edeltäneen tuottavuuden tason (Nielsen 1993, 26). Järjestelmän tulisi olla helposti muistettava, jotta satunnaiset käyttäjät kuten osa-aikatyöntekijät pystyisivät käyttämään järjestelmää ilman, että he joutuvat jokaisella käyttökerralla opettelemaan kaiken uudelleen (mts. 31 - 32). Opittavuuden ja helppokäyttöisyyden tavoin myös muistettavuus tehostaa työskentelyä vähentämällä käyttöohjeiden selaamiseen ja työkaverilta kysymiseen kuluvaa aikaa (Alanne 2002, 35).

4. Virheettömyys

Virheettömyydellä viitataan tuotteen käytössä syntyvien virhetilanteiden määrään ja niistä toipumiseen (Nielsen 1993, 26). Mitä helpompi tuotetta on käyttää, sitä vähemmän virhetilanteita syntyy. Huonosta käytettävyydestä johtuvat virheet ja odottamattomat toiminnot ovat harmillisia ja vaikuttavat mielikuvaan tuotteen laadukkuudesta, mutta pahimmillaan ne voivat johtaa hyvin vakaviin seurauksiin. (Kuparinen 2008, 31.) Koska virhetilanteita ei kuitenkaan voida välttää kokonaan, niistä tulisi selvittää

mahdollisimman helposti ilman ohjelmiston kaatumista tai tietokannan vioittumista. Katastrofaalisia virhetilanteita ei Nielsenin mielestä saisi sattua laisinkaan. (Nielsen 1993, 32 - 33.)

5. Miellyttävyys

Käytettävyydeltään hyvää tuotetta on myös miellyttävä käyttää. Timo Leinon mukaan yksi tietojenkäsittelytoiminnan onnistumisen avaintekijöistä on käyttäjän riittävä tyytyväisyys käytettäviin työvälineisiin (Alanne 2002, 24). Tyytyväisyys on suoraan verrannollinen työskentelytehokkuuteen ja sitä kautta koko organisaation liiketoimintaan. Käytettävyydeltään huonoilla järjestelmillä on vaikutusta työssä viihtymiseen ja ne voivat aiheuttaa työntekijöille muun muassa stressiä sekä turhautumista. (Nielsen 1993, 34.) Näistä työympäristön aiheuttamista fyysisistä ja psyykkisistä ongelmista johtuvista sairauspoissaoloista koituu ylimääräisiä kustannuksia myös yritykselle.

Myös esteettisyys on osa tuotteen miellyttävyyttä. Ohjelman ollessa auki näytöllä koko päivän, korostuvat myös visuaaliset seikat (Isotalo 2007, 4). Käyttöliittymän selkeys, johdonmukaisuus ja yksinkertaisuus yhdessä tasapainon, harmonian ja visuaalisten jännitteiden kanssa lisäävät käyttäjän tyytyväisyyttä ja vahvistavat myös kuvaa järjestelmän miellyttävyydestä ja helppokäyttöisyydestä (Sinkkonen ym. 2006, 157).

6. Ymmärrettävyys

Antti Wiio käyttää kirjassaan Käyttäjäystävällisen sovelluksen suunnittelu läheisesti Nielsenin opittavuuteen liittyvää termiä ymmärrettävyys. Wiion mukaan ymmärrettävän laitteen tai sovelluksen käyttäjän on helppo päätellä mitä laitteella voi tehdä ja kuinka sen avulla päästään haluttuun lopputulokseen (mts. 29). Ymmärrettävä laite antaa käyttäjälleen riittävästi palautetta, eikä jätä epäselvyyttä siitä, mitä milloinkin tapahtuu. Myös virheilmoitusten on oltava ymmärrettäviä. Tämä on erityisen tärkeää esimerkiksi ydinvoimaloiden, lääketeollisuuden ja terveydenhuollon sovelluksissa. Ymmärrettävyyttä voidaan edistää muun muassa vieraan termistön ja kielen välttämällä (Kuparinen 2008, 26).

7. Vaivattomuus

Toinen Wiion (2004) asettamista mittareista on vaivattomuus. Vaivattoman sovelluksen avulla käyttäjä voi suoriutua tehtävistään ja päästä tavoitteeseensa mahdollisimman suoraviivaisesti ja yksinkertaisesti. Esimerkiksi matkapuhelimella lähetettävät

tekstiviestit ovat tehokkaita ja ymmärrettäviä, mutta näppäimistön pienestä koosta johtuen moni voi kokea sen hyvin vaivalloiseksi. (Mts. 30.) Vaivalloinen toimenpide voi viedä kohtuuttomasti aikaa ja aiheuttaa sen vuoksi taloudellisia tappioita.

3.2 Käytettävyyden subjektiivisuus

ISO-standardin (SFS-EN ISO 9241-11) mukaan käytettävyys on subjektiivinen ilmiö eli riippuvainen käyttäjästä. Aloittelijoiden järjestelmälle ja käyttöliittymälle asettamat vaatimukset ovat erilaisia kuin edistyneemmillä käyttäjillä. Järjestelmän käyttöä opiteltaessa sovelluksen helppokäyttöisyyden ja opittavuuden merkitys korostuu, mutta käyttötaitojen kartuttua nousevat sovelluksen hyödyllisyys ja tehokkuus pääasiaksi. (Nielsen 1993, 28 - 30.)

Käyttäjän osaamistason lisäksi järjestelmän käytettävyyteen vaikuttaa sen käyttötarkoitus (Kuutti 2003, 15). Pankkien maksuautomaateilta ja julkisilta internetsivustoilta vaaditaan selkeyttä ja helppokäyttöisyyttä, mutta esimerkiksi ammattikäyttöön tarkoitettujen työkalusovellusten täytyy olla ensisijaisesti suunniteltuja tehokasta ja nopeaa työskentelyä varten, jolloin opittavuudesta ja myös miellyttävyydestä voidaan joustaa.

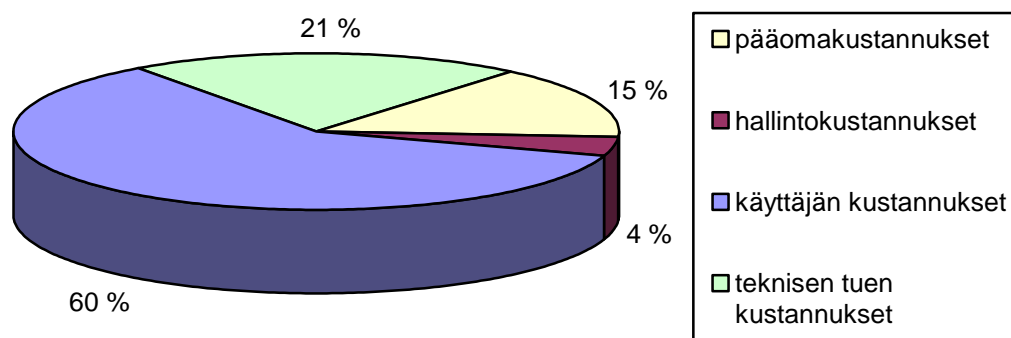
Kolmanneksi tarkastelun kohteeksi standardi asettaa järjestelmän käyttöympäristön, jolla tarkoitetaan käyttötilannetta ja käytettävää laitteistoa (Kuutti 2003, 15). Nykyaikana tietoteknisiä sovelluksia käytetään useilla erilaisilla päätelaitteilla. Työpaikalla käytettävien tietokonepäätteiden näytöt ovat huomattavasti suurempia kuin esimerkiksi matkapuhelimissa ja kämmentietokoneissa. Myös tietojen kirjaaminen on matkapuhelimella vaikeampaa, jolloin käyttöliittymäsuunnittelijan haasteena on pohtia, kuinka kaikki tarvittavat toimenpiteet saadaan pienen näytön ja näppäimistön ehdoilla jaettua toimivaksi ja helposti suoritettavaksi tehtäväketjuksi. Näin ollen laitteiston fyysiset ominaisuudet määrittävät esimerkiksi sekä kentältä matkapuhelimen että toimiston tietokoneen kautta käytettävän Liikkuvan työn ohjausjärjestelmä TARMO 2.0:n käyttöliittymäratkaisuja.

Käytettävyyden määritelmien asettamat mittarit ovat laajoja ja moniulotteisia. Tietämissä tehtävissä tarvittava tehokkuus voi olla absoluuttista suoritusnopeutta, mutta toisissa tilanteissa tehokkuutta on virheetön suoriutuminen. Tämän vuoksi käytettä-

vyyteen kohdistuvien vaatimusten tapauskohtainen määrittely ja käyttäjien sekä käyttöympäristön tunteminen on ensiarvoisen tärkeää uusia järjestelmiä suunniteltaessa. (Kuoppala & Nieminen 1998, 20.) Käyttäjän aktiiviseen huomioimiseen ja käytettävyyden systemaattiseen kehittämiseen perustuvan käyttäjäkeskeisen suunnitteluprosessin periaatteita käsitellään luvussa 4.1.

4 KÄYTETTÄVYYDEN KEHITTÄMINEN

Organisaatiot käyttävät hyväksi tietotekniikkaa tavoitteenaan kasvattaa liiketoiminnan tuottavuutta ja tehostaa työskentelyä. Aina tietotekniikkainvestoinnit eivät kuitenkaan ole johtaneet halutun suuruisiin vaikutuksiin tuottavuuden lisääntymisessä. Itse asiassa tietotyöläisten tuottavuuden kasvu on hidastunut (Alanne 2002, 24). Turun kauppa-
korkeakoulussa tehdyn tutkimuksen mukaan yrityksen tietoteknisten apuvälineiden kokonaiskustannukset koostuvat pääomakustannuksista, hallintokustannuksista, käyttäjän kustannuksista ja teknisen tuen kustannuksista. Kuviosta 1 voidaan nähdä, että liiketoiminnan kannalta merkittävimmät menoerät ovat teknisen tuen kustannukset sekä käyttäjän kustannukset, millä tarkoitetaan käytännössä työaikaa, joka kuluu tietokoneen ja tietojärjestelmien käyttöön liittyvissä mutta käyttäjän varsinaiseen työhön kuulumattomissa toimissa (mts. 23).



KUVIO 1. Yrityksen tietoteknisten apuvälineiden kokonaiskustannukset (Alanne 2002, 23)

Käytettävyys on käyttöliittymää suurempi asia ja kiinnittämällä siihen huomiota voidaan sekä helpottaa uuden järjestelmän käyttöönottoa että tehostaa työntekoa ja tätä kautta pienentää asiakasyrityksen liiketoiminnan kannalta vahingollisia piilokustannuksia (Wiio 2004, 21 - 22). Käytettävyydeltään intuitiivinen, tehokas ja miellyttävä järjestelmä vähentää koulutus- ja tukikustannuksia, lisää käyttäjien tyytyväisyyttä ja

parantaa niin työntekijöiden kuin koko organisaationkin operatiivista tehokkuutta. Taloudellisten ja sosiaalisten hyötyjen lisäksi yrityksen tarjoamien julkisten palveluiden ja tiedotuskanavien, kuten internetsivujen, käytettävyydellä on huomattava vaikutus koko yrityksen imagoon (mts. 31).

Tässä luvussa tutustutaan käytettävyyden kehittämisen ja käyttäjäkeskeisen suunnittelun periaatteisiin. Käyttäjäkeskeisen suunnittelun prosessimallin tarkoituksena on toimia järjestelmäkehittäjien ohjenuorana, minkä avulla voidaan varmistaa koko suunnitteluprosessin kestävä käytettävyyssnäkökulmien huomioiminen. Lisäksi tarkastellaan, kuinka käytettävyyden kehittämiseen panostaminen vaikuttaa ohjelmistotalon toimintaan.

4.1 Käyttäjäkeskeisen suunnittelun periaatteet

Ihmiset käyttävät sovelluksia ja palveluita vain apuvälineenä päästäkseen ensisijaiseen tavoitteeseen, joka voi olla esimerkiksi laskun maksaminen tai työtehtävän hoitaminen. Järjestelmä, joka tukee mahdollisimman hyvin näitä tavoitteita, voidaan toteuttaa vain, jos tiedetään, mitä käyttäjä todella tarvitsee ja minkälaisessa ympäristössä kehitettävää järjestelmää käytetään. (Lohikoski 2007, 49.)

Kehitettävän järjestelmän vaatimuksia määritettäessä olisi kiinnitettävä huomiota siihen, mitä käyttäjät tekevät, ei siihen, mitä he sanovat. Ohjelmistokehittäjän asiakkaan luona suorittamat käyttäjätarkkailut (*user observations*) antavat arvokasta tietoa asiakkaan työympäristöstä ja työhön liittyvistä ongelmakohdista, joita voidaan uuden järjestelmän avulla kehittää (Kuutti 2003, 144). Käyttäjien ja heidän tarpeidensa entistä parempi huomioiminen ohjelmistotuotantoprosessin aikana on noussut hyvin arvostetuksi lähestymistavaksi suunniteltaessa uusia järjestelmiä. Aikaisemmat tekniikkalähtöiset lähestymistavat ovat joutuneet väistymään uusien ihmissläheisempien menetelmien tieltä.

Käyttäjäkeskeinen suunnittelu (*user-centered design*), joka tunnetaan myös nimellä ihmiskeskeinen tuotekehitys, on monialainen kehittämismenetelmä, jonka tunnuspiirteitä ovat käyttäjän aktiivinen osallistuminen, käyttäjän tehtävien ja käyttöympäristön selvittäminen sekä koko kehitysprojektin ajan kestävä suunnittelun ja arvioinnin itera-

tiivisuus eli toistuvuus. (Lohikoski 2007, 49.) Tuotekehityksen tueksi ja käytettävyyden huomioimisen varmistamiseksi on kehitetty erilaisia suunnitteluprosesseja, jotka kuvaavat yleisellä tasolla käyttäjäkeskeistä suunnittelua ja sen vaiheita. Nykyisin suositeltava käytettävyyden kehittämisen prosessi on esitetty standardissa ISO 13407 (Vuorovaikutteisten järjestelmien käyttäjäkeskeinen suunnitteluprosessi). Standardi esittää neljä keskeisintä käyttäjäkeskeisen suunnittelun vaihetta, jotka tulisi sisällyttää järjestelmän kehittämisprojektiin (SFS-EN ISO 13407, 16):

1. käyttökontekstin ymmärtäminen ja määrittäminen
2. käyttäjä- ja organisaatiovaatimusten määrittäminen
3. suunnitteluratkaisujen tuottaminen ja
4. suunnitelmien arviointi verrattuna asetettuihin vaatimuksiin.

Käyttökontekstilla tarkoitetaan erityisesti kehitettävän järjestelmän käyttäjiä, sille asetettuja tavoitteita ja suunniteltuja tehtäviä sekä järjestelmän käyttöympäristöä. Käyttöympäristö käsittää sekä fyysiset tilat ja laitteistot että työhön vaikuttavat lait, säädökset ja standardit. (Mts. 18.)

Käyttäjä- ja organisaatiovaatimusten määrittely kokoaa yhteen käyttäjien ja organisaation tarpeista ja toiveista johdetut toiminnalliset vaatimukset sekä kehitettävälle järjestelmälle asetetut käytettävyyksivaatimukset. Kaikki vaatimukset täytyy priorisoida ja käytettävyyksivaatimukset on asetettava mitattavaan muotoon, jotta käytettävyyksiarvion tuloksista voidaan todeta, kuinka nämä vaatimukset täyttyvät. (Mts. 20.)

Suunnitteluratkaisujen ideoinnissa hyvä lähtökohta on etsiä optimaalinen ratkaisumalli kullekin käyttötapaukselle välittämättä ratkaisun teknisestä toteutuksesta. Lupaavia suunnitteluratkaisuja havainnollistetaan tekemällä niistä eritasoisia prototyyppkejä, jotka voivat olla yksinkertaisimmillaan paperiarkille piirrettyjä. Prototyyppkejä testataan yhdessä todellisten käyttäjien kanssa. Prototyyppit toimivat myös tärkeänä kommunikaation välineenä keskusteltaessa asiakkaan kanssa. (Mts. 22.) Nopeasti käsin piirrettyjen käyttöliittymäkuvien avulla varmistetaan kaikkien kehitysryhmän osapuolten yhtenäinen kuva järjestelmästä. Verrattuna yleisesti käytettyihin UML-käyt-

tötapauksiin, havainnolliset kuvat tekevät asiakkaan omien parannusehdotusten esittämisestä helpompaa. (Laakso 2006, 20.)

Suunnitteluratkaisujen arviointien tavoitteena on tarjota palautetta suunnitelmien parantamiseksi ja tarkistaa, täyttyvätkö toisessa vaiheessa annetut käyttäjä- ja organisaatiovaatimukset. Arvioinnit voidaan suorittaa tekemällä käyttäjätestejä tai käyttämällä asiantuntija-arviointia. Arviointien antamien tuloksien pohjalta suunnitteluratkaisuja parannetaan ja kehitetään iteratiivisesti. Jotta suunnitteluprosessi olisi käyttäjäkeskeinen, vähintäänkin järjestelmän lopullinen testaus tulisi suorittaa sovelluksen todellisilla käyttäjillä. (SFS-EN ISO 13407, 24.)

Käyttäjäkeskeinen suunnittelu on prosessi, joka aloitetaan ohjelmistotuotantoprojektin alkumetreiltä ja sitä jatketaan iteratiivisesti läpi koko järjestelmäkehityksen elinkaaren aina siihen saakka, kunnes asetetut vaatimukset täyttyvät. ISO 13407 -standardissa suunnitteluprosessi on määritelty hyvin abstraktilla tasolla, eikä se ota kantaa käytettäviin menetelmiin tai tekniikoihin. Mallin tarkoituksena on toimia ensisijaisesti käytettävyyden kehittämisen ohjenuorana projektipäälliköille ja yritysjohdolle. (Mts. 12.) Standardin pohjalta on kuitenkin johdettu ohjelmistokehityksen tarpeisiin monia prosessimalleja, jotka kannustavat käyttäjän huomiointiin ja suunnittelun iteratiivisuuteen. Tunnetuin ja eniten käytetty käyttäjäkeskeisen suunnittelun menetelmä on Hugh Beyerin ja Karen Holtzblattin 1990-luvulla kehittämä Contextual Design (Kuutti 2003, 142). Myös suomalaiset ovat osallistuneet käyttäjäkeskeisten suunnittelumenetelmien kehittämiseen. Helsingin yliopiston tietojenkäsittelytieteen laitoksella alkunsa saanutta tavoitepohjaisen käyttöliittymäsuunnittelun GUIDe-prosessimallia tarkastellaan seuraavassa luvussa.

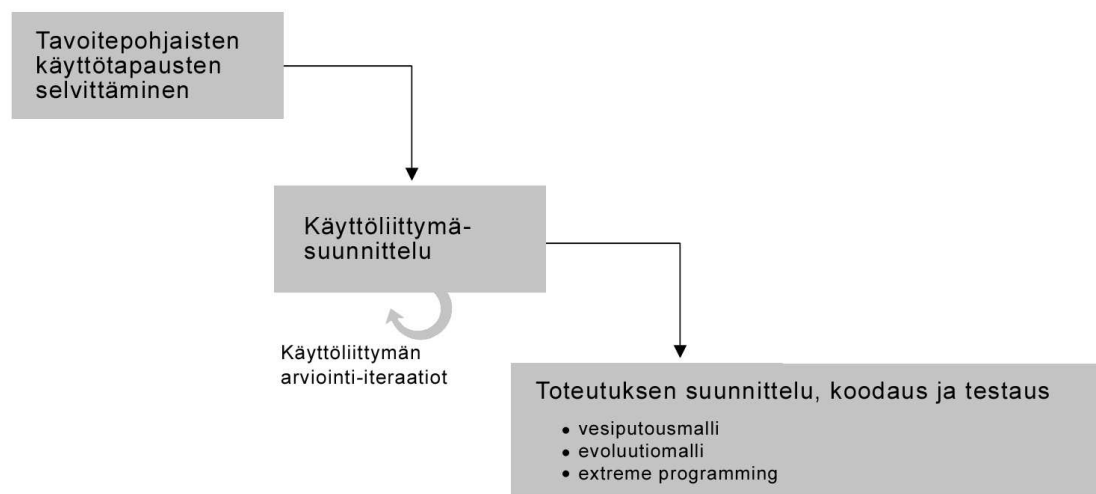
4.2 Tavoitepohjainen käyttöliittymäsuunnittelu

Monet käytettävyysongelmat huomataan vasta ohjelmiston valmistuttua, kun järjestelmällä yritetään ensimmäisen kerran suorittaa käyttäjien todellisia työtehtäviä.

Yleensä tämä johtuu siitä, että käyttöliittymää ei suunnitella eikä sen soveltuvuutta käyttötarkoitukseensa testata ennen järjestelmän käyttöönottoa. Esimerkiksi luvussa 2.3 kuvattu ja yleisesti käytetty vesiputousmallia muistuttava prosessimalli ei sisällä

varsinaista käyttöliittymäsuunnittelua, vaan sovelluksen käyttöliittymä muotoutuu projektin suunnitteluvaiheessa ikään kuin etukäteen määritellyn toiminnallisuuden päälle. Pahimmillaan käyttöliittymä syntyy ohjelmoinnin sivutuotteena. Sovelluksen toiminta ja soveltuvuus käyttäjän tarpeisiin konkretisoituu asiakkaalle vasta järjestelmän ensimmäisten versioiden valmistuttua. (Laakso & Laakso 2004, 1 - 2.)

Sen sijaan, että järjestelmän hyödyllisyys ja soveltuvuus todellisiin työtehtäviin selvittäisiin vasta käyttöönottovaiheessa, koko järjestelmä voidaan alusta lähtien suunnitella käyttötilanteiden ja käyttäjien tavoitteiden pohjalta. Helsingin yliopiston yhdessä Interacta Design Oy:n kanssa kehittämässä käyttöliittymälähtöiseen määrittelyyn perustuvassa GUIDe-prosessimallissa perinteisen vaatimusmäärittelyn ongelmat ratkaistaan projektin alkuun sijoitettavan systemaattisen käyttöliittymäsuunnittelun avulla (Laakso & Laakso 2004, 2). GUIDe-prosessimallin keskeisimmät vaiheet ja sen sijoittuminen osaksi projektin elinkaarta on esitelty kuviossa 2.



KUVIO 2. GUIDe-mallin keskeisimmät vaiheet (Laakso & Laakso 2004, 2)

Prosessimallin keskeisimmät työkalut ovat käyttäjän päivittäisiä työkulkuja kuvaavat tavoitepohjaiset käyttötapaukset (*goal-based use cases*) ja konkreettiset käyttöliittymäkuvat, jotka visualisoivat simuloitujen käyttötilanteiden perusteella laadittua käyttöliittymäratkaisua. Suunnittelun tuloksena syntyvän käyttöliittymäkuvauksen (*user interface specification*) avulla käyttöliittymän ja koko järjestelmän soveltumista käyttäjän työkulkuihin voidaan testata jo määrittelyvaiheessa. Käyttäjätestauksen avulla varmistetaan, että ohjelmisto sisältää kaikki käyttäjän tarvitsemat toiminnot, ja että

niiden käyttö on mahdollisimman suoraviivaista. (GUIDe-menetelmä 2008.) Käyttöliittymäratkaisu tuotetaan GUIDe-prosessimallin perustana olevan simulointipohjaisen käyttöliittymäsuunnittelun GDD-mallin (*Goal-Derived Design*) mukaisesti. Tämän suunnitteluprosessin ideana on simuloida käyttäjän työnkulkua askel askeleelta piirtämällä näkyviin käyttötilanteiden eri vaiheissa tarvittavat toiminnot ja tietosisällöt. (GUIDe-menetelmä 2008.)

Muita yleisesti käytettyjä käyttöliittymän systemaattisen suunnittelun prosessimalleja ovat muun muassa Virtual Windows ja myös aiemmin mainittu Contextual Design. Kaikille menetelmille on yhteistä käyttäjän työtehtävien selvittäminen ja käyttöliittymän suunnittelu niiden pohjalta. (Hanhisalo 2007, 9.) GUIDe ei itsessään käsitä koko ohjelmistotuotantoprojektia, mutta se täydentää hyvin esimerkiksi vesiputousmallia, jonka tyypillinen ongelma on liian myöhäisessä vaiheessa paljastuvat muutostarpeet (Laakso & Laakso 2004, 12). Kun GUIDen periaatteiden mukaisesti tuotettu käyttöliittymäratkaisu on testattu, vaatimukset ja tietosisällöt voidaan irrottaa käyttöliittymäkuvista ja dokumentoida esimerkiksi skenaariomuotoisina käyttötapauksina tuotekehityksen jatkoa varten (Laakso 2006, 20).

4.3 Miksi käytettävyyttä kannattaa kehittää?

Ihmisen ja tietokoneen välisen vuorovaikutuksen tutkiminen on tuonut paljon hyödyllistä tietoa ihmisten tarpeita vastaavien tietokonepohjaisten järjestelmien suunnitteluun. Myös käytettävyydeltään korkeatasoisten sovellusten on kiistattomasti todettu edistävän työntekijöiden hyvinvointia ja parantavan työskentelytehokkuutta. Kuitenkaan sovelluskehittäjät eivät kylliksi panosta käytettävyyden kehittämiseen ja se jää edelleen sivuseikaksi harmittavan monessa ohjelmistotuotantoprojektissa. Tekniikka & Talous -lehti uutisoi syksyllä 2003 suomalaisten terveydenhuollon sovellusten epäonnistuvan juuri kehnon käytettävyyden takia. Syynä tähän on usein tiedon puute. (Kuutti 2003, 19.)

Esimiestaso ei aina ymmärrä käytettävyyden laajuutta ja sen merkitystä vaan ajattelee käytettävyydellä ja siihen panostamisella tarkoitettavan graafista sommittelua ja tuotekehityksen lopussa tehtäviä käytettävyytestestejä (Isomäki 2006, 14). Projektijoh-

don lisäksi myös ohjelmiston tilaajalla voi olla hatara mielikuva käytettävyyssuunnittelun merkityksestä. Tästä johtuen kehitettävän järjestelmän helppokäyttöisyyttä pidetään ylimääräisenä menoeränä ja huomio kiinnitetään yksinomaan tuotekehityksen kustannuksiin, sisällytettäviin ominaisuuksiin ja projektiaikataulun toteutumiseen. (Kuparinen 2008, 37.)

Pinnalle liimattavan ominaisuuden sijaan käytettävyys on koko järjestelmän ydin ja systemaattisella käytettävyyden kehittämällä saavutettavien hyötyjen on todettu olevan kehittämiskustannuksia suuremmat (Alanne 2002, 38). Hofmann ja Lehner esittävät kirjassaan *Requirements Engineering as a Success Factor in Software Projects* (2001), että erityisen onnistuneita ovat ne projektit, joissa järjestelmän loppukäyttäjät toimivat kehitysryhmän kanssa tiiviissä yhteistyössä projektin alusta lähtien. Suunnittelijoiden konkreettisissa työskentelytilanteissa ja -ympäristöissä suorittamat käyttäjätarkkailut ja niiden pohjalta myös asiakkaan ymmärtämässä muodossa laaditut tehtäväkuvaukset yhdessä käyttöliittymäprototyyppien kanssa helpottavat asiakkaan kanssa järjestelmän ominaisuuksista käytävää keskustelua ja muutosehdotusten esittämistä. Samalla ne vähentävät vaatimusten määrittelyyn tarvittaviin palavereihin kuluva aikaa. Ulkopuolisen tarkkailijan on lisäksi mahdollista huomata kehittämistarpeita koko asiakasorganisaation työtoiminnassa, jolloin sopivan käyttöliittymäratkaisun tuomat vaikutukset ulottuvat pelkkää tietojärjestelmää kauemmaksi. (Mts. 28.)

Käyttäjäkeskeisten suunnittelumenetelmien avulla varmistetaan todellisten tarpeiden löytyminen ja se, että kaikkien osapuolten kehitettävää järjestelmää koskevat näkemykset ovat yhteneväisiä sekä totuudenmukaisia heti ensimmäisellä kerralla (SFS-EN ISO 13407, 16). Muutokset ovat kuitenkin aina mahdollisia, ja niiden tekeminen tai huonoksi todettujen ominaisuuksien karsiminen on määrittelyvaiheessa projektin myöhempisiin vaiheisiin verrattuna kymmeniä ellei satoja kertoja edullisempaa (Nielsen 1993, 8). On myös laskettu, että virheen korjaamisesta koituvat kustannukset ovat tuotteen luovuttamisen jälkeen jopa tuhatkertaiset kehitystyön määrittelyvaiheen korjauskustannuksiin verrattuna (Alanne 2002, 28). Virheiden havaitseminen ja korjaaminen suunnittelun alkuvaiheessa säästää ylimääräiseltä työltä ja ehkäisee kehitysryhmän turhautumista (Sinkkonen ym. 2006, 20). Huolellisesti laaditulla vaatimusmäärittelyllä voidaan vähentää ohjelmistoprojektin iteraatiokierroksia ja helpottaa

kehitysprojektin aikataulujen sekä kustannuksien arviointia (GUIDe-menetelmä 2008).

Käyttäjakeskeisen suunnittelun ja projektiryhmän sekä järjestelmän todellisten käyttäjien välisen yhteistyön on todettu vaikuttavan myös asiakasorganisaation toimintaan. Tutkimusten mukaan järjestelmän loppukäyttäjien osallistuminen ohjelmistokehitysprojektiin ja mahdollisuus päästä suoraan vaikuttamaan kehitettävään järjestelmään vahvistaa käyttäjien uutta järjestelmää kohtaan tuntemaa tyytyväisyyttä sekä mielikuvia järjestelmän hyödyllisyydestä ja helppokäyttöisyydestä. Järjestelmän käyttöönottoaiheessa positiivisilla mielikuvilla on suuri vaikutus työskentelymotivaatioon ja järjestelmän kohtaamaan muutosvastarintaan, joka voi johtaa helposti siihen, että tietotekniikan tarjoamat hyödyt jäävät saavuttamatta (Alanne 2002, 32 - 33). Kaikkia käyttäjäorganisaatiolle koituvia taloudellisia ja sosiaalisia hyötyjä, kuten käytön tehokkuutta, tukikustannusten vähenemistä ja pienempää muutosvastarintaa voidaan käyttää myyntiargumentteina.

Kaikkien edellisten huomioiden perusteella systemaattisen käytettävyyden kehittämisen voidaan katsoa tuovan ohjelmistotalolle laaja-alaisia hyötyjä. Tuotekehitysprosessin tehostumisen ja asiakkaalle koituvien etujen lisäksi tuotteen laatu paranee ja myynti lisääntyy. Laadukas ja käyttäjäystävällinen tuote on tärkeä valttikortti loppuunkilpaillulla IT-alalla (Kuutti 2003, 15). Helposti opittavan ja virheettömän järjestelmän vähäinen tuen tarve mahdollistaa myös kehittäjäorganisaatiolle edullisten tukipalvelusopimusten tarjoamisen (Alanne 2002, 35). Sisällyttämällä käytettävyyden huomioiminen osaksi organisaation tuotekehitystä ja jatkamalla kehittämisprosessia yli yksittäisten projektien rajojen saadaan käytettävyyteen panostamisesta koituvat lisäkustannukset katettua moninkertaisesti kehitys- ja tukikustannuksien alenemisella saavutettavilla säästöillä (mts. 44).

4.4 Käytettävyyden kehittämisen edellytykset

Käytettävyyden tehokas ja tulokellinen kehittäminen edellyttää panostamista koko organisaatiolta. Yrityksen työntekijöillä täytyy olla tarpeeksi tietoa käytettävyyden vaikutuksista ja riittävät taidot erilaisten käytettävyyden kehittämisen menetelmien

sekä työkalujen hyödyntämiseen (Alanne 2002, 36). Käytettävyys on tieteenalana poikkitieteellinen, minkä vuoksi kehitysryhmän jäsenten on pystyttävä toimimaan yhteistyössä eri alojen osaajien kanssa. Tutkimusten mukaan käyttäjäystävällisten tietojärjestelmien kehittämisen parissa työskentelevillä on hyvin erilaiset koulutustaustat. Käytettävyysammattilaiset ovat opiskelleet pääaineinaan muun muassa tietojenkäsittelytiedettä, psykologiaa, kognitiotieteitä, mediatutkimusta, prosessitekniikka, sosiologiaa, teollista muotoilua, viestintää ja kulttuuriantropologiaa (Isomäki 2006, 7). Hyvän käytettävyysasiantuntijan tulisikin olla yhdistelmä insinööriä, psykologia, kasvatustieteilijää ja taiteilijaa (Kuutti 2003, 14).

Myös organisaation ylimmän johdon ja projektipäälliköiden tulisi ymmärtää käytettävyystyön laajuus ja merkitys. SIGCHI Finland ry:n vuonna 2006 tekemän tutkimuksen mukaan keskeisin käytettävyyden kehittämisen epäonnistumiseen johtava tekijä on liian vähäiset resurssit (Isomäki 2006, 13). Käyttäjäkeskeisen tuotekehityksen prosessimalleissa projektin painopiste siirtyy vaatimusmäärittelyyn. Jos käytettävyyteen panostamisella saavutettavat hyödyt eivät ole kaikkien tiedossa, määrittelyvaiheen venyminen voi aiheuttaa turhautumista ja uudet työvaiheet koetaan helposti rasitteeksi, ja niihin ei haluta kuluttaa ylimääräistä aikaa tai rahaa (Alanne 2002, 37). Toisen tutkimuksessa selvinnyt ongelma on käytettävyystekijöiden liian myöhäinen huomioiminen. Käytettävyystyön ammattilaiset tulisi ottaa mukaan kehitystiimiin jo ennen asiakasvaatimusten määrittelyä, ja koko projektin elinkaaren kestäväälle käytettävyyden kehittämiselle ja arvioinnille tulisi varata riittävästi aikaa ja henkilöstöä. (Isomäki 2006, 13.)

Yhteenvetona projektikohtaisesti onnistuneen käytettävyyden kehittämisen voidaan katsoa edellyttävän projektipäälliköiden ja keskitason johdon tukea muun muassa riittävien aika- ja henkilöstöresurssien muodossa. Organisaatiolta menestyksekkäs käytettävyystyö puolestaan vaatii työntekijöiden riittävän laajaa ja monipuolista käytettävyysosaamista sekä asianmukaisia välineitä ja tiloja. Tämän lisäksi käytettävyyden kehittämiselle olisi hyvä asettaa konkreettisesti mitattavia tavoitteita ja suunnittelijoiden tulisi saada asiakaspalautetta kehitystyön tuloksista, mikä edelleen kannustaa käytettävyydeltään korkealaatuisten järjestelmien kehittämiseen. (Alanne 2002, 36.)

5 TARMO 2.0 -KÄYTTÖLIITTYMÄSUUNNITTELU

Tässä luvussa kuvataan Liikkuvan työn ohjausjärjestelmä TARMO 2.0 -kehitystyön vaiheita ja tarkastellaan käyttöliittymäsuunnitteluprosessin aikana käytettyjä menetelmiä. Suunnitteluprosessissa sovellettiin luvussa 4.2 esiteltyä tavoitelähtöisen käyttöliittymäsuunnittelun GUIDe-prosessimallia. Luvun tarkoituksena ei ole ainoastaan raportoida työohjausjärjestelmän käyttöliittymäkuvauksen tuottavan suunnitteluprosessin elinkaarta, vaan pikemminkin esitellä prosessimallin vaiheita liikkuvan työn ohjausjärjestelmän keskeisimpien käyttötapauksien avulla ja arvioida prosessimallin soveltuvuutta toimeksiantajan pienen mittakaavan vaatimusmäärittely- ja kehitysprojekteihin.

Suunnittelutyön rakenne noudattaa standardissa ISO 13407 määriteltyjä käyttäjakeskeisen tuotekehityksen vaiheita (ks. luku 4.1). Ensin valotetaan projektin lähtökohtia ja tavoitteita, minkä jälkeen esitellään vaihe vaiheelta työohjausjärjestelmän käyttöliittymäprototyypin tuottamiseen tähtäävän suunnittelutyön eteneminen ja siinä käytetyt menetelmät. Luvussa 5.2 esitellään työohjausjärjestelmän keskeisimmät käyttäjäryhmät ja käyttöympäristön sekä käyttäjien tarpeiden selvittämisessä käytettäviä menetelmiä. Käyttäjätutkimuksen jälkeen tarkastellaan tehtyjen havaintojen analysoimisprosessia ja esitellään työohjausjärjestelmän käyttöliittymäsuunnittelun kannalta keskeisimmät tavoitepohjaiset käyttötapaukset. Esiteltyjen käyttötapauksien pohjalta demonstroidaan simulointipohjaista käyttöliittymäsuunnittelua ja laaditaan työohjausjärjestelmän käyttöliittymäratkaisu uuden työtehtävän lisäämistä varten. Lopuksi tarkastellaan käyttöliittymäratkaisun arviointimenetelmiä ja tavoitepohjaisen käyttöliittymäsuunnitteluvaiheen jälkeisiä ohjelmistokehitysprojektin jatkotoimia.

5.1 Tausta ja tavoitteet

Tutkimuksen lähtökohtana on toimeksiantajan TARMO 2.0 -kehitysprojekti, jonka päämääränä on toteuttaa työohjausjärjestelmän uusi versio. Kehitystyötä varten tehdyn taustatutkimuksen avulla selvitettiin, millaisia kokemuksia käyttäjillä on työoh-

jausjärjestelmän nykyisestä versiosta. Käyttäjäpalautteesta selvisi, että vuonna 2007 valmistuneen järjestelmän käyttöliittymää määrittävät monet toteutustekniset ratkaisut, jotka yhdessä lukuisten työnteon kannalta ylimääräisten toimintojen kanssa heikentävät tuotteen käytettävyyttä. Näin ollen TARMOn uuden version kehitystyössä pyritään erityisesti huomioimaan järjestelmän käyttäjäystävällisyyteen liittyvät tekijät.

Ohjelmistotuotantoprojekti käynnistettiin kesäkuussa 2008 kolmehenkisen kehitysr ryhmän voimin. Määrittelytyön eri vaiheisiin osallistuivat myös asiakkaan edustajat. Tutkijan vastuualueeksi projektissa määräytyi käyttöliittymäsuunnittelu, ja muut ryhmän jäsenet huolehtivat asiakaskontakteista ja teknisestä toteutuksesta. Järjestelmän käytettävyyden varmistamiseksi tutkija ehdotti, että määrittelyvaiheessa käytettäisiin tavoitepohjaisen käyttöliittymäsuunnittelun GUIDe-prosessimallia. Suunnitteluprosessin avulla järjestelmästä pyrittiin kehittämään mahdollisimman selkeä ja käyttäjien todellisia tarpeita palveleva. Simulointipohjaisen suunnitteluprosessin avulla laadittu järjestelmän käyttöliittymäkuvaus toimii syötteenä kehitysprojektin toteutukselle.

5.2 Käyttäjätutkimus

ISO-standardi (SFS-EN ISO 13407, 12) mainitsee käyttäjäkeskeisen suunnitteluprosessin lähtökohdiksi käyttäjien huomioon ottamisen järjestelmän kehityksessä ja käyttäjävaatimusten sekä tehtävävaatimusten ymmärtämisen. Käyttäjän työnkulkuja tuntematta päädytään helposti järjestelmiin, jotka tukevat vain yksittäisiä toimintoja, mutta kokonaisten tehtävien tekeminen on vaivalloista tai mahdotonta (Hanhisalo 2007, 12). Liikkuvan työn ohjausjärjestelmä TARMO 2.0 -kehitysprojektin ensimmäisessä vaiheessa suoritettiin käyttäjätutkimus, jonka tarkoituksena oli selvittää, kuka kehittävä järjestelmää tulee käyttämään, millaisia tilanteita käyttäjä joutuu työssään kohtaamaan ja mitä tietoja näissä tilanteissa tarvitaan. Lisäksi pyrittiin muodostamaan kuva käyttäjän toimintaympäristöstä selvittämällä organisaation koko, käyttäjien tietotekninen osaamistaso, käytössä oleva teknologia ja muut liiketoimintaan tai sovel-lusalaan liittyvät tietotekniset ratkaisut. Kehitystyötä varten tarvittavat taustatiedot käyttäjäryhmistä ja toimintaympäristöstä hankittiin muun muassa työnohjausjärjestelmän nykyisen version vaatimusmäärittelydokumentteja ja asiakkaan sovellusalaasta kertovaa aineistoa tutkimalla. Työntekijöiden keskeisimmät työtehtävät ja kehitettä-

välle järjestelmälle asetetut käyttäjävaatimukset selvitettiin GUIDe-prosessimallin mukaisia kenttätutkimusmenetelmiä soveltaen.

Käyttöympäristön selvittäminen

Käyttäjätutkimus aloitettiin selvittämällä kotisairaanhoito- ja kotipalvelualalla toimivan asiakasyrityksen työntekijöiden keskeisimmät työtehtävät ja vastuualueet. Saatujen tietojen perusteella työntekijät jaoteltiin käyttäjäryhmiksi, joista jokaisella on oma roolinsa organisaation toiminnassa. Seuraavassa on esitelty yrityksen työnohjauksen ja raportoinnin toimintaprosessien kannalta olennaiset osapuolet.

- *Työnohjaaja* on yrityksen työntekijöiden lähin esimies. Hän laatii työvuorolistat, jakaa tehtävät kenttätyöntekijöille ja huolehtii asiakaskontakteista. Lisäksi työnohjaaja hallinnoi asiakkaiden, työntekijöiden ja työntekijöiden muodostamien tiimien tietoja sekä vastaa yrityksen palveluista ja tuotteista. Työnjohdosta vastaa yleensä yksi henkilö.
- *Kenttätyöntekijä* toteuttaa yrityksen suorittavaa työtä. Kenttätyöntekijä saa suoritettavat työtehtävänsä työnohjaajalta. Hänen vastuulla on kuitata tehtävä vastaanotetuksi, suorittaa se määräysten mukaisesti ja lopulta kuitata tehtävä tehdyksi. Kenttätyöntekijällä on oikeus tietää omat työvuoronsa kolme viikkoa etukäteen. Hän voi tietyssä tilanteessa nähdä myös muiden työntekijöiden työvuoroja ja työtehtäviä koskevat tiedot. Yrityksessä työskentelee tällä hetkellä noin 10 kenttätyöntekijää.
- *Toimitusjohtaja* vastaa yrityksen liiketoiminnasta. Hänen työtehtäviinsä kuuluu yrityksen asiakassuhteista ja yrityksen työntekijöistä huolehtiminen, mukaan lukien uusien asiakkaiden hankkiminen ja lisätyövoiman palkkaaminen. Toimitusjohtajalle liiketoiminnan kannalta tärkeitä tietoja ovat muun muassa yrityksen tuottavuus, eri tuotteiden ja palveluiden menekki sekä työntekijöiden tekemät työtunnit.

Ennen varsinaisen käyttäjätutkimuksen aloittamista käyttäjäryhmien lisäksi selvitettiin asiakasyrityksen toimialaan ja käytössä olevaan teknologiaan liittyviä seikkoja. Kotisairaanhoitoalaa ja potilastietojen käsittelemistä koskevat lukuisat lait ja säädökset, jotka asettivat vaatimuksia myös kehitettävän työnohjausjärjestelmän tietoturvalle.

Teknologiakartoituksen avulla selvitettiin muut yrityksen käyttämät tietotekniset ratkaisut. Asiakkaan käyttämät järjestelmät, kuten laskutuksen, kirjanpidon ja palkanlaskennan sovellukset, antavat kehittäjille arvokasta tietoa organisaation toiminnasta. Sovellukset on myös huomioitava kehitysprojektin myöhemmissä vaiheissa kehitettävän järjestelmän teknisiä rajapintoja suunniteltaessa. (Viitanen 2005, 18.)

Työnteon tutkiminen

Käyttäjärühmien ja käyttöympäristöön liittyvän tutkimuksen jälkeen selvitettiin järjestelmään kohdistuvat toiminnalliset vaatimukset. Koska TARMOn uuden version kehitystyössä tarkoituksena on parantaa jo ennestään olemassa olevaa järjestelmää, käyttäjätarpeiden selvittäminen voitiin aloittaa arvioimalla nykyistä tuotetta yhdessä käyttäjien kanssa. Saadun palautteen perusteella tarkennettiin nykyisen järjestelmän dokumentoinnissa määritellyjä vaatimuksia. Tarkennetut vaatimukset tarjosivat perustan uuteen versioon tehtäville muutoksille. Samalla voitiin välttää nykyisiin ratkaisuihin liittyvien ongelmien toistaminen. Useimmiten käyttäjät eivät kuitenkaan osaa sanoa suoraan, mitä haluavat tai millaisia ongelmia heidän nykyisiin tehtäviinsä tai työnkuvaansa liittyy. Tämän vuoksi käyttäjien piilotarpeita ja konkreettisia työnkulkuja selvitettiin myös asiakkaan Jyväskylän toimipisteessä suoritettujen kenttätutkimusten avulla. (Viitanen 2005, 18 - 19.)

Kenttätutkimuksen tarkoituksena on havainnoida käyttäjän työntekoa hänen todellisessa työskentely-ympäristössään, missä käyttäjällä on saatavilla muistilaput, asiakirjat ja muu työnkulkuun oleellisesti liittyvä materiaali (Kuutti 2003, 144). Laakson (2004, 7) mukaan kaksi tyypillistä työnkulkujen selvittämisessä käytettävää menetelmää ovat kontekstuaaliset käyttäjähaastattelut (*contextual interviews*) ja käyttäjätarkkailut (*user observations*). Käyttäjätarkkailussa seurataan vierestä käyttäjän toimia ja työtehtävien suorittamista. Passiivisen tarkkailun avulla nähdään muutaman tarkkailukerran aikana useita keskeisiä käyttötilanteita ja saadaan selville, millaisia vaatimuksia työtehtävien tekeminen kehitettävälle järjestelmälle asettaa (GUIDE-menetelmä 2008). Kontekstuaalinen käyttäjähaastattelu perustuu käyttäjälle esitettyihin kysymyksiin, joiden avulla pyritään selvittämään haastateltavan käyttäjän työnkulkuja. Haastattelu suoritetaan käyttäjän varsinaisessa työpisteessä, jossa haastateltava näyttää esimerkkien avulla, miten hän suorittaa keskeisimpiä työtilanteita. Kontekstuaalisen käyttäjähaastattelun avulla voidaan selvittää pitkäkestoisia tai harvoin tapahtuvia työtilanteita,

mutta menetelmän käyttö on käyttäjätarkkailuun verrattuna haastavampaa: Haastattel-tavan tehtävänä on työnkulkujen selvittämisen lisäksi saada käyttäjä kertomaan, miksi hän tekee asiat juuri valitsemallaan tavalla. Menetelmästä riippumatta käyttäjätutkimuksessa on tärkeää pyrkiä löytämään toimenpiteiden takana olevat korkean tason tavoitteet. Vaikka toteutustavat muuttuisivat, käyttäjän lopulliset tavoitteet pysyvät usein samoina ja niiden ymmärtäminen helpottaa koko työprosessin hahmottamista. (Hanhisalo 2007, 12 - 13.)

TARMO 2.0 -kehitystyön määrittelyvaiheessa järjestettiin kaksi asiakaspalaveria, joiden aikana esimerkkiyrityksen työnohjaaja demonstroi työtehtäviään nykyisin käytössä olevalla työnohjausjärjestelmällä. Käyttäjätutkimusta ei voitu suorittaa käyttäjän todellisen työn ohessa, joten työnkulkuja selvitettiin kontekstuaalisen käyttäjähaastattelun menetelmää soveltamalla. Haastattelut pidettiin käyttäjän omassa työpisteessä ja haastateltavaa pyydettiin näyttämään esimerkinomaisesti, miten hän työtehtäviään suorittaa. Tärkeimpänä pidettiin niiden työnkulkujen selvittämistä, joiden suorittaminen oli nykyisellä sovelluksella hankalaa. Keskeisimpiä käsiteltyjä toimintoja olivat työmääräyksen tekeminen ja tehtyjen työtuntien koostaminen raportiksi. Haastattelutilanteen aikana tehdyt huomiot kirjattiin ylös. Muistiinpanojen tekemisen lisäksi haastattelu- ja käyttäjätarkkailutilanteet voidaan tallentaa videoimalla ja selvitettävien työnkulkujen kriittisistä vaiheista voidaan ottaa digitaalisia valokuvia (Laakso & Latva-Koivisto 2006). Koska selvitettyjä työnkulkuja pystyttiin tarvittaessa simuloimaan TARMO:n aikaisemmalla versiolla myös itse, ei haastattelutilanteiden videoimisen katsottu olevan tarpeellista.

Suullisesti käytäviin vaatimusmäärittelypalaveriinkin verrattuna kenttätutkimusmenetelmät tuottavat kerralla totuudenmukaista tietoa käyttäjien ja organisaation tarpeista. Monesti vaatimusmäärittelypalaverissa kehitettävän järjestelmän ominaisuuksista päättävät tuotteen loppukäyttäjien sijaan johtoportaan edustajat, joiden ymmärrys käyttäjän työnkuluista voi olla virheellinen (Nielsen 1993, 14). Käyttäjätarkkailun avulla saadaan selville myös työtoiminnan kannalta ylimääräiset työvaiheet ja muut rutiinit, joita käyttäjä itse ei muista mainita tai pidä järjestelmäkehityksen kannalta tärkeänä (Hanhisalo 2007, 8). Työskentelyn pieniin rutiineihin vaikuttamalla on kuitenkin mahdollista suoraviivaistaa työskentelyä ja tehostaa koko organisaation liiketoimintaa (Wiio 2004, 22).

5.3 Tutkimusaineiston analysointi

Käyttäjätutkimuksen paljastama tieto potentiaalisista käyttäjistä, työympäristöstä ja työskentelyyn liittyvistä tavoista loivat hyvän lähtökohdan ja pohjan TARMOn uuden version vaatimuksien määrittämiselle. Tavoitepohjaisten käytötapausten määrittämisen kannalta tärkeimmäksi tiedoiksi muodostuivat haastattelujen ja käyttäjätarkkailujen avulla kerätty laadullinen aineisto käyttäjien tavoista sekä tottumuksista. Näitä tietoja täydensivät olemassa olevan työnohjausjärjestelmän vaatimusmäärittelydokumenteista ja käyttäjäpalautteesta sekä asiakasyrityksen työssä tarvittavista lomakkeista ja materiaaleista poimitut tiedot.

Käyttäjätutkimuksen aikana kerätyt tiedot hankittiin hyvin suppealta joukolta vain yhdenlaista työympäristöä tutkimalla. Valmiin järjestelmän pitäisi kuitenkin soveltua paljon laajemmalle, työskentelytottumuksiltaan ja tietoteknisiltä taidoiltaan erittäin heterogeeniselle käyttäjäjoukolle. Kotipalvelun lisäksi vastaavanlaiselle työnohjausjärjestelmälle on myös olemassa markkinat esimerkiksi kiinteistöhuoltoalalla. Tietojen analysoinnin tarkoituksena on siis kvalitatiivisia tutkimusmenetelmiä käyttäen muodostaa käyttäjätutkimuksen aikana tehdyistä havainnoista kattava ja yhtenäinen kuvaus järjestelmän tulevista käyttäjistä sekä heidän työtehtävistään (Kuutti 2003, 159).

Käyttäjäpersoonat

Suunnittelutyön avuksi analyysivaiheessa laadittiin kuvitteellisia käyttäjäpersoonia, joiden avulla pyrittiin mallintamaan mahdollisimman tarkasti järjestelmän potentiaalisia käyttäjiä. Käyttäjäpersoonia laadittaessa niiden pohjana käytettiin järjestelmän käyttäjäryhmäkuvauksia (luku 5.2.1). Käyttäjäpersoonista pyrittiin tekemään mahdollisimman luonnollisia. Niille annettiin nimi, ikä, perhesuhteita ja harrastuksia sekä erilaisia ominaisuuksia, jotka liittyivät ensisijaisesti henkilön tietotekniseen osaamistasoon. Käyttäjäpersooniin voidaan liittää myös kuva. (Kuutti 2003, 122.) Kuten alla esitetystä asiakasyrityksen henkilöstöjohtajan toimenkuvan pohjalta laaditusta persoonakuvauksesta voidaan nähdä, järjestelmän käyttäjät voivat työtehtäviensä puolesta kuulua moneen eri käyttäjäryhmään. Kun käyttöliittymäratkaisut suunnitellaan laadittujen käyttäjäpersoonien pohjalta, eivät työntekijän päällekkäiset käyttäjäroolit aiheuta järjestelmää käytettäessä esimerkiksi ylimääräistä navigointia tai muita käytettävyyssongelmia.

- *Tiina Toivanen*, 32, on kahden lapsen äiti ja koulutukseltaan terveydenhoitaja. Hän työskentelee kotisairaanhoidon- ja kotipalveluita tarjoavan yrityksen henkilöstöjohtajana Jyväskylän keskustassa. Työn ohessa Tiina suorittaa terveydenhoitoalan täydennyskoulutuksen kursseja avoimessa ammattikorkeakoulussa ja ATK-ajokorttia kansalaisopistossa. Pienessä yrityksessä henkilöstöjohtajan työnkuvaan kuuluvat asiakkaiden tilauksista huolehtimisen ja työnohjaajan tehtävien lisäksi satunnaiset asiakaskäynnit. Työnohjausjärjestelmän lisäksi Tiina joutuu työssään käyttämään palkanlaskentaohjelmaa ja tyypillisimpiä toimistosovelluksia. Kotoaan hänellä ei ole tietokonetta.

Kuvitteellisten käyttäjien ensisijainen tarkoitus on helpottaa suunnittelijoiden työtä ja toimia syötteenä käyttötapauksien laatimista varten (Laakso & Latva-Koivisto 2006). Mikäli käyttäjien henkilökohtaiset ominaisuudet eivät vaikuta käyttöliittymän integraatoratkaisuihin tai tiedon organisointiin, tarkkojen persoonakuvausten laatimisesta ei kuitenkaan ole suurta hyötyä. Beyerin ja Holtzblattin (1998) mukaan käyttäjäpersoonien sijasta voidaan tällöin käyttää esimerkiksi organisaation työtehtävien ja vastualueiden jakautumista mallintavia roolikuvauksia. (Lilja 2005, 48.) Merkittävin syy persoonakuvauksien laatimiselle on kuitenkin se, että ne auttavat suunnittelijoita huomioimaan ja ymmärtämään käyttäjän tarpeita. Kun potentiaalisia käyttäjiä ei edustakaan pelkkä laatikko, vaan heillä on nimet ja henkilökohtaisia ominaisuuksia, järjestelmästä on helpompi kehittää juuri tälle käyttäjäryhmälle hyödyllinen (Sinkkonen ym. 2006, 44).

Tavoitepohjaiset käyttötapaukset

Analyysivaiheessa luodaan pohja myös GUIDe-mallissa käytettävien tavoitepohjaisten käyttötapauksien laatimiselle. Käyttäjätutkimuksen avulla kerätyn aineiston perusteella vaikutti siltä, että erilaisia käyttötilanteita löytyy kymmeniä, ellei satoja. Laakson (2004, 8) mukaan monet laadituista käyttötapauksista ovat todellisuudessa toistensa variaatioita ja niitä on mahdollista tukea samalla käyttöliittymäratkaisulla. Ennen tavoitepohjaisten käyttötapauksien laatimista, samoja ominaisuuksia sisältävät käyttötilanteet kategorisoitiin. Tutkimusaineiston analysoinnin ja ryhmittelyn avulla laadittavien käyttötapauksien määrä saatiinkin vähennettyä hämmästyttävään pieneen ja edustavaan otokseen.

Analysoinnin apuna käytettiin erilaisia listoja, kaaviota ja affiniteettidiagrammeja. Affiniteettidiagrammi on laadullisen aineiston analysoinnissa käytettävä menetelmä, jonka avulla aineistoa voidaan ryhmitellä ja priorisoida sekä koota yhteen tutkimusten avulla selvitettyjä ilmiötä (Viitanen 2005, 21). Aineiston läpikäynnin yhteydessä eri käyttäjiin ja käyttötapauksiin liittyvät havainnot kirjoitettiin ylös pienille lapuille, jotka ryhmiteltiin kehitettävän järjestelmän toimintoja ja sisältöjä kuvaaviksi kokonaisuuksiksi. Järjestellyt laput muodostivat muun muassa seuraavat ryhmät: henkilötiedot, työtehtävien jakaminen, tekemättömät työtehtävät, työvuorot ja viestit. Esimerkiksi työtehtävän jakamisen ryhmään sisältyivät muun muassa seuraavat usein toistuvia työtehtäviä koskevat havainnot: *Sama työtehtävä voi toistua viikoittain aina eri työntekijän tekemänä ja Työtehtävä luodaan järjestelmään ja määrätään työntekijälle joka kerta uudestaan.* Nämä ja muut ryhmittelyn aikana esiin nousevat havainnot muodostivat lähtökohdan tavoitepohjaisille käyttötapauksille.

Käyttötapauksella tarkoitetaan tyypillisesti yksittäistä, tietyn tavoitteen täyttymiseen johtavaa tehtävää, jota käyttäjä tuotteella tekee. Käyttötapaukset esitetään yleensä tekstinä tai yksinkertaisina graafisina kaavioina. (Kuutti 2003, 132.) GUIDe-mallissa keskeisenä työkaluna käytetään tavoitepohjaisia käyttötapauksia. Tavoitepohjaiset käyttötapaukset ovat pikemminkin loppukäyttäjän näkökulmasta kirjoitettuja toimintatarinoita, jotka kuvaavat käyttäjän ongelmaa, tavoitteita ja senhetkistä tilannetta. Ne eivät kiinnitä erityisesti huomiota mihinkään tiettyyn tuotteeseen tai sen käyttämiseen, vaan käyttäjien ongelmia kuvataan yleisellä tasolla (mts. 134; Laakso & Latva-Koivisto 2006). Tästä eteenpäin käyttötapauksella tarkoitetaan tavoitepohjaista käyttötapusta.

Käyttäjä tutkimuksen aineistoa analysoitaessa havaittiin, että yrityksen tarjoamien palveluiden tilaamiseen liittyvistä asiakaskontakteista on olemassa noin puolenkymmentä erilaista variaatiota, joita edellisessä luvussa kuvattu henkilöstöjohtajan käyttäjäpersoonaa voi työssään kohdata. Tilanteita ovat esimerkiksi mahdollisimman nopeasti suoritettava työtehtävä, säännöllisin väliajoin toistuva työtehtävä ja tiettyyn päivämäärään mennessä suoritettava tehtävä. Näiden lisäksi esiintyy tilanteita, joissa tehtävän suorittajaksi halutaan jokin tietty työntekijä tai tehtävä vaatii sellaista erikoisosaamista, jota ei ole kaikilla yrityksen työntekijöillä. Erilaiset tilanteet kategorisoitiin ja niitä pyrit-

tiin kuvaamaan mahdollisimman kattavasti kuvioissa 3 ja 4 esitettyjen käyttötapausten avulla.

Laaditut käyttötapauskuvaukset koostuvat tavoiteosasta ja tilatiedot-osasta. Ne eivät ota kantaa järjestelmän toteutukseen, vaan niitä varten pyritään selvittämään käyttäjän konkreettiset ongelmat ja tavoitteet. (Laakso & Laakso 2004, 7 - 8). Kuvioiden 3 ja 4 käyttötapauksissa ei keskitytä uuden työtehtävän järjestelmään lisäämiseen tai sen jakamiseen työntekijöille, vaan käyttötapausten tavoiteosassa kuvataan käyttäjän korkeamman tason tavoitteita, jotka ovat juuri sopivasti suunniteltavan järjestelmän toiminnallisuuden yläpuolella. Esimerkiksi kuvion 3 käyttötapauksessa Tiinan tavoitteena on selvittää maanantaina 20.10. virka-aikaan vapaana olevat työntekijät. Tilatiedoista selviää tilanteen kannalta oleelliset taustatiedot, jotka vaikuttavat esimerkiksi toiminnan aktivoitumiseen. Näitä tietoja ovat muun muassa asiakkaan tekemän puhelinsoiton ajankohta ja Mikon sekä muiden lähihoitajien maanantain työtehtävät.

Käyttötapaus 1: Kertaluontoinen työtehtävä

Tiinan tavoite: Kotipalveluyrityksen työnhajaaja Tiina vastaanottaa Anteron, 76-vuotiaan eläkeläisen puhelun. Antero asuu Jyväskylän keskustassa, mutta tarvitsee jonkun kotipalveluyrityksen työntekijöistä tekemään tulevan maanantain päivittäistavaraostokset ja käymään apteekista verenpainelääkkeet, koska yleensä hänen asioista huolehtiva veljenpoikansa on työmatkalla Hollannissa. Mikko on Anterolle ennestään tuttu työntekijä yrityksen ateriapalvelun kautta, joten Tiina selvittää ensisijaisesti hänen vaurailanteensa.

Tilatietoja:

- Nyt on torstai 16.10. kello on 09:12. Tiina on työpaikallaan Jyväskylän keskustassa.
- Anteron tarvitsee apua asioinnissa 20.10. kello 9 - 16 välillä. Työtehtävä kestää 1–2 tuntia.
- Työtehtävän suorittaminen vaatii auton ja ajokortin.
- Tiina ei muista seuraavalle viikolle jaettuja työvuoroja tai työtehtäviä ulkoa.
- Yrityksessä työskentelevät Mikko, Suvi, Matti, Kaisa ja Tiina.

KUVIO 3. Kertaluontoisen työtehtävän käyttötapaus

Käyttötapaus 2: Viikoittain toistuva työtehtävä

Tiinan tavoite: Lonkkaleikkauksesta toipuva Sirkka, 79, soittaa kotoaan Muuramesta ja varaa Tiinalta puhelimitse viikoittaista siivousapua ainakin seuraavan kahden kuukauden ajaksi. Siivouspalvelua tarvitaan ensimmäisen kerran ennen tulevaa sunnuntaita, jolloin on Sirkkan 80-vuotissyntymäpäivät. Siivouspäivä voi olla myös tulevina viikkoina esimerkiksi perjantai. Sirkka ei ole ennen käyttänyt kodinhoitopalveluita ja on yrityksen uusi asiakas.

Tilatietoja:

- Nyt on torstai 16.10. kello on 11:50. Tiina on työpaikallaan Jyväskylän keskustassa.
- Sirkkan tietoja ei ole yrityksen asiakasrekisterissä.
- Ensimmäisen kerran siivous tulisi tehdä viimeistään lauantaina 18.10. kello 16:een mennessä. Työtehtävä kestää yhdeltä työntekijältä 2–3 tuntia.
- Kuka tahansa työvuorossa oleva vapaa työntekijä voi suorittaa tehtävän.
- Tiina ei muista seuraavalle viikolle jaettuja työvuoroja tai työtehtäviä ulkoa.
- Yrityksessä työskentelevät Mikko, Suvi, Matti, Kaisa ja Tiina.

KUVIO 4. Viikoittain toistuvan työtehtävän käyttötapaus

Käyttötapauksia laadittaessa olisi huomattava, että järjestelmän käyttäjä ei ole vain tietokoneen ääressä istuva ja työnohjausjärjestelmää hallinnoiva työntekijä, vaan myös muut tuotteen vaikutuspiiriin kuuluvat henkilöt. Edellä esitellyissä käyttötapauksissa kotipalveluyrityksen työnohjaaja on työnohjausjärjestelmän primäärikäyttäjä ja palveluita tilaava asiakas sekundaarikäyttäjä. (Kuutti 2003, 119.) Tehtävän suorittamiseen vaikuttavat kuitenkin ensisijaisesti yritykseen soittavan asiakkaan tilanne ja tarpeet. Niinpä sopiva käyttöliittymäratkaisu, jolla asiakas voisi itse varata tarvitsemansa palvelun, tekisi myös asiakkaasta järjestelmän primäärikäyttäjän, mikä puolestaan vähentäisi välikätenä toimivan työnohjaajan työmääräyksien tekemiseen kuluvaan aikaa.

Käyttäjätutkimuksen tietoja analysoiva ja käyttötapauksia laativa suunnitteluryhmä on parhaimmillaan mahdollisimman poikkitieteellinen. Psykologit ja käytettävyydestutkijat sekä sovellusalan asiantuntijat antavat kukin omanlaisen näkökulman käyttöliittymäratkaisujen suunnitteluun. (Vuori 1999, 19.) Työnohjausjärjestelmän kehittämistä varten suoritettun käyttäjätutkimuksen aineiston purkamiseen ja havaintojen ryhmittelyyn osallistuivat kaikki projektiryhmän jäsenet. Kuutin (2003, 164) mukaan olisi kuitenkin paras, jos suunnitteluryhmässä ei olisi mukana lainkaan järjestelmän toteutuksesta vastaavia henkilöitä. Tällöin suunnitteluratkaisut laaditaan varmasti loppukäyttäjän ehdoilla, eivätkä huomiot käytettävän tekniikan rajoituksista tai ajatukset järjestelmän tietokanta-arkkitehtuurista pääse vahingossakaan vaikuttamaan lopputulokseen. Tältä osin käyttäjäkeskeisen suunnittelun menetelmät skaalautuvat huonosti TARMO 2.0 -kehitystyön kokoisiin projekteihin.

5.4 Käyttöliittymäsuunnittelu

Käyttötapauskuvausten laatimisen jälkeen aloitettiin käyttöliittymäsuunnittelu. Kuutin (2003, 164) mukaan käyttäjäkeskeisen suunnittelun tai tuotekehityksen perimmäisenä tarkoituksena ei ole vain uuden ohjelmiston kehittäminen, vaan korkein ja kunnianhimoisin tavoite on suunnitella yritykselle kokonaan uusi ja aikaisempaa parempi tapa toimia. GUIDe perustuu käyttöliittymän simulointipohjaisen tuottamisen GDD-suunnittelumenetelmään, jonka vapaamuotoisesti kirjoitetut tehtäväkuvaukset ja laajempia toimintoketjuja kuvaavat käyttöliittymäprototyypit mahdollistavat yrityksen työnkul-

kujen sekä parhaimmillaan kokonaisten liiketoimintaprosessien suoraviivaistamisen (Hanhisalo 2007, 7 - 9).

Projektin suunnitteluvaiheessa voitiin suoraviivaistaa myös kuviossa 4 esitettyä työtehtävää. Käyttäjätutkimusvaiheessa selvitettyjen nykyisten työnkulkujen mukaan Tiinan on suoritettava useita erillisiä toimintoja, että hän saa 1) kirjattua Sirkan henkilötiedot asiakasrekisteriin, 2) selvitettyä työnohjaussivulla suoritettavien hakutoimintojen avulla kyseisenä ajankohtana käytettävissä olevat työntekijät ja 3) valittua työtehtävän lisäyssivulta työntekijän ja asiakkaan tiedot, kirjoitettua tehtäväkuvauksen ja määrättyä työtehtävän valitulle työntekijälle. Suurin ongelma on kuitenkin se, ettei nykyinen työnohjausjärjestelmä tue säännöllisin väliajoin toistuvia työtehtäviä, vaan tehtävä joudutaan luomaan ja työmääräys lähettämään joka viikko uudelleen. Kun edellä mainituista ongelmista tehdyt huomiot otetaan uuden työnohjausjärjestelmän suunnittelutyön lähtökohdaksi, voidaan työnkulkua oikaista ja tehdä työtehtävien ylimääräisestä toistosta tarpeetonta.

Käyttöliittymäsuunnitteluprosessin aluksi TARMOn keskeisimpien osien sisällöistä ja yhteyksistä laadittiin UED-kaavio (*User Environment Design*), eli niin sanottu näyttökartta (ks. liite 1). Kyseessä on Contextual Design -menetelmästä lainattu kuvaustekniikka, joka muistuttaa käyttöliittymäsuunnittelussa tavallisesti käytettyä dialogikaaviota. Näyttökartta laadittiin lähtökohdaksi käyttöliittymäsuunnittelulle ja sen ajateltiin auttavan projektin toteutuksesta huolehtivia tahoja, jotka eivät osallistuneet käyttöliittymäkuvien laatimisen, hahmottamaan erillisten näyttökuvien välisiä suhteita sekä niiden muodostamaa kokonaisuutta. Sovelluksen pohjapiirroksiksi laadittua näyttökarttaa testattiin arvioimalla, kuinka sujuvasti käyttötapauksissa kuvattujen työtehtävien tekemisen vaatimat toimenpiteet voidaan suorittaa kaavion eri tilojen välillä liikkumalla. (Kuutti 2003, 167.)

Koska kaikki järjestelmälle asetetut vaatimukset eivät ole keskenään yhtä tärkeitä, käyttötapaukset priorisoitiin. Priorisoinnilla varmistetaan usein toistuvien tai muilla tavoin kriittisten käyttötapauksien huomioiminen suunnittelussa ennen vähempiarvoisia käyttötapauksia (Viitanen 2005, 23). TARMOn käyttöliittymäsuunnittelussa tuetaan ensisijaisesti kuvioissa 3 esitettyä käyttötapausta. Edellä mainitun uuden työtehtävän lisäämisen käyttötapauksen ja sen eri variaatioiden avulla kuvataan myös koko käyttöliittymän suunnitteluprosessin etenemistä. Se, miksi tutkimuksen raportointi rajataan

näihin käyttötapauksiin ja toteutusratkaisuihin, johtuu siitä, että mainitut tehtävät ovat loppukäyttäjän työnkulun kannalta oleellisia ja niiden suorittamisessa käsitellään lähes kaikkia järjestelmään tallennettavia tietoja.

Simulointipohjaisen GDD-suunnitteluprosessin ideana on tuottaa mahdollisimman suoraviivainen käyttöliittymäratkaisu, jolla tavoitteeseen pääseminen on mahdollisimman suoraviivaista. Käyttöliittymää piirretään simuloimalla yhtä käyttötapausta kerrallaan ja lisäämällä tietoja sekä toimintoja näkyviin käyttöliittymäkuviin sitä mukaa kuin niitä työtehtävän suorittamisessa tarvitaan. Näin jatketaan, kunnes kaikki käyttötapaukset on käyty läpi ja niiden tarvitsemat toiminnot mallinnettu käyttöliittymään. (Laakso & Latva-Koivisto 2006.) TARMOn käyttöliittymäratkaisun näyttökuvien tuottaminen aloitettiin korkeimmalle priorisoidun käyttötapauksen (kuvio 3) pohjalta. Työtehtävän suorittamisen kannalta Tiinan ensimmäinen tavoite on selvittää kaikki yrityksessä työskentelevät lähihoitajat, jotka ovat työvuorossa seuraavan viikon maanantaina. Toiseksi Tiinan täytyy tietää, onko Mikolla tai jollakin muulla työntekijällä vapaata aikaa kello 9–16 välillä. Jos työtehtävä käsittäisi apteekkikäynnin lisäksi esimerkiksi lääkinnällisiä tehtäviä, valittavalla työntekijällä tulisi lisäksi olla terveydenhoitajan ammattitutkinnon antama riittävä pätevyys. Tässä tapauksessa työmääräyksen antamista rajoittava tekijä on muun muassa ajokortin puuttuminen.

Kuviossa 5 on esitetty mahdollisimman suoraviivainen käyttöliittymäratkaisu, johon on piirretty näkyviin uuden työtehtävän lisäämisen käyttötapauksen vaatimat tiedot ja toiminnot. Ikkuna on jaettu kahteen osaan, joista vasemmanpuoleisessa on kentät asiakkaan tietojen, työtehtävän tietojen sekä ajankohdan valitsemista varten. Jos Tiina haluaa etsiä ensin vain Mikon työtehtävät, hän voi lisätä hakukriteereihin myös työntekijän tiedot. Esitetietojen antamisen jälkeen suoritetaan hakutoiminto, minkä jälkeen ikkunan oikealle puolelle ilmestyvät niiden työntekijöiden tehtävälistat, jotka ovat käytettävissä valittuna ajankohtana ja pystyvät suorittamaan hakukriteerien mukaisen tehtävän. Mikäli hakua varten määritettiin myös työntekijä, silloin useamman työntekijän päiväkalenterien sijaan näytetään viikkokalenteri vain valitun työntekijän tehtävistä. Tehtävälistoista selviävät työntekijöiden työvuorot ja heidän suoritettavakseen määrättyt työtehtävät. Tehtävää suorittamaan määrättävä työntekijä valitaan napsauttamalla vastaavaa saraketta työntekijöiden päiväkalenterista.

nykyistä paremmin, minkä vuoksi edellä mainittu käyttötapaus priorisoitiin suunnittelutyössä korkealle.

Nykyinen käyttöliittymäratkaisu (kuvio 5) mahdollistaa vain yhden kerran tietynä ajankohtana suoritettavien työtehtävien määräämisen työntekijöille. Myöskään työtehtävän tekijöiden määrä ei voi olla muuta kuin yksi. Viikoittain toistuvan työtehtävän käyttötapausta varten käyttöliittymään tarvitaan toiminto, joka mahdollistaa työtehtävän määrittämisen toistuvaksi viikoittain seuraavan kahden kuukauden ajan sekä eri työntekijän valinnan jokaiselle suorituskerralle. Koska on selvää, että jokin toinen tehtävä voi toistua esimerkiksi päivittäin tai vaikka kuukausittain, järjestelmän on tuettava myös näitä tilanteita. Tässä vaiheessa ei oteta kantaa toimintoihin, joita vaaditaan Sirkan henkilötietojen lisäämiseksi asiakasrekisteriin.

Käyttöliittymäratkaisuun tehdyt muutokset ovat nähtävillä alla olevassa näyttökuvassa (kuvio 6). Edelleen kehitetyssä käyttöliittymäratkaisussa työtehtävän tiedot sisältävät työtehtävän tyypin ja ajankohdan lisäksi valintakentät, joiden avulla määritetään työtehtävän toistumisesta kertovat tiedot. Mikäli työtehtävä asetetaan toistuvaksi tietyin väliajoin, sen ensimmäinen suorittaja valitaan päiväkalenterin sarakkeista kuten ennenkin, mutta työtehtävän tulevia suorituskertoja symboloivat laatikot, joka ilmestyvät ikkunan alalaidassa olevaan määräämättömät -altaaseen (pool). Tähän altaaseen kerryvät kaikki työtehtävät, jotka tulevat toistumaan tiettyinä ajankohtina ja joille ei vielä ole määritetty seuraavaa suorittajaa. Tehtävä määrätään työntekijälle määräämättömät-altaassa olevaa laatikkoa painamalla, jolloin asiakkaan ja työtehtävän tiedot siirtyvät ikkunan vasemman puolen valintakenttiin. Tehtävän suorittava työntekijä valitaan samalla tavalla kuin ennenkin.

LUO Uusi TYÖTEHTÄVÄ

asiakas:

osoite:

tehtävä:

pöytänumero:

kellonaika:

työntekijä:

työntekijän kmr:

HAE

viikoittain ☐

ma ti ke to pe la su ☐

loppu pvm: saakka

MÄÄRÄÄMÄTÖMÄT TYÖTEHTÄVÄT

pvm	klo	tehtävä	asiakas
21.10.2008	10-12	Siivous	Sirkka Koppinen
28.10.2008	10-12	Siivous	Sirkka Koppinen
29.10.2008	9-10	Aloripulohu	Heli Rajala

TYÖNTEKIJÖIDEN PÄIVAKALENTERI, 17.10.2008

	Mikko	Suvi	Matti	Kaisa	Tina
7	Siivous		Polven		Aloripulohu
8					
9	Aloripulohu				Siivous
10					
11	Siivous			Siivous	
12	Aloripulohu				
13					
14					
15					

LÄHETÄ

KUVIO 6. Viikoittain toistuvan työtehtävän määräämisen käyttöliittymäratkaisu

Jos piirrettyjen ominaisuuksien täyttämät näyttökuvat alkavat näyttää liian sekavilta, käyttöliittymäratkaisu voidaan refaktoroida. Sillä tarkoitetaan käyttöliittymäratkaisun purkamista osiin ja järjestämistä uudelleen niin, että siitä tulee helpommin ymmärrettävä. (Laakso & Latva-Koivisto 2006.) Suunnittelun aikana piirrettyjä käyttöliittymäkuvia ja paperiprototyyppejä voidaan testata tuotteen loppukäyttäjien kanssa. Muutokset ovat nopeita tehdä, koska kaikki järjestelmässä tarvittava toiminnallisuus on edelleen olemassa vain kuvien ja kuvasarjojen muodossa.

TARMon käyttöliittymäratkaisun arviointia varten yksittäisistä paperiluonnoksista muokattiin kuvankäsittelyohjelmalla nopeasti laadittuja käyttöliittymäkuvia ja Power-Point-näyttökuvasarjoja, joilla havainnollistettiin käyttötapauksissa kuvattuja työtehtäviä. Tosin käytettävyydestä käsittelevän kirjallisuuden mukaan hieman suttuiset luonnokset ja käsin piirretyt paperiprototyyppit olisivat loppuun asti viimeisteltyjä näyttökuvia parempia työkaluja käyttäjien kanssa kommunikointiin. Käyttäjän on helpompaa ehdottaa parannuksia paperiarkeista koostuvaan käyttöliittymämalliin kuin lähes valmiilta tuotteelta näyttävään ja jopa joitakin toiminnallisuuksia sisältävään prototyyppiin. (Kuutti 2003, 108.)

5.5 Käyttöliittymäratkaisun arviointi

Tavoitepohjaisen käyttöliittymäsuunnittelun kehityssyklin viimeinen vaihe on tuotetun käyttöliittymäratkaisun arviointi. Arvioinneilla pyritään varmistamaan, että suunnittelussa käyttöliittymässä on juuri se toiminnallisuus ja tietosisältö, jota järjestelmää käytettäessä tarvitaan (*utility*) ja että käyttöliittymäratkaisut ovat suoraviivaisia, tehokkaita sekä yksiselitteisen ymmärrettäviä (*usability*). (Laakso & Laakso 2004, 16.)

Toisin kuin käyttötapausskenaarioiden ja UML-kaavioiden avulla laaditut vaatimusmäärittelydokumentit, käyttöliittymäkuvat mahdollistivat tuotetun ratkaisun testaamisen oikeita työtilanteita simuloimalla yhdessä sekä kehitysryhmän että järjestelmän loppukäyttäjien kanssa. Asiakkaan toimipisteessä www-sivujen avulla havainnollistettua käyttöliittymäratkaisua testattiin antamalla käyttäjien suorittaa työtehtäviään vaihe vaiheelta. Kun käyttöliittymäprototyypillä yritettiin oikeasti työskennellä, nähtiin, kuinka asiakkaan työntekijät reagoivat suunniteltuihin ominaisuuksiin ja muuttuneisiin käytäntöihin. Lisäksi saatiin varmistettua, onko prototyypin toiminnallisuuksissa tai tiedoissa sellaisia puutteita, jotka lopullisessa järjestelmässä olisivat heidän työskentelynsä kannalta haitallisia. (Laakso 2006, 20.)

Tässä luvussa esiteltävät tavoitepohjaisessa käyttöliittymäsuunnitteluprosessissa tyyppisesti käytetyt arviointimenetelmät ovat simulointipohjainen asiantuntija-arvio (*expert review*), läpikäyntipalaverit (*walkthroughs*) ja käytettävyytestit (*usability tests*). Menetelmiä voidaan käyttää niin suunnitteluvaiheen yksittäisten käyttötapausketjujen testaamiseen, kuin myös lopullisen käyttöliittymäkuvauksen ja käyttöliittymäprototyypin arviointiin. (Laakso & Laakso 2004, 9 - 10.)

Käyttöliittymän arviointimenetelmät

Simulointipohjainen asiantuntija-arvio tai simulointitestaus on arviointimenetelmä, jossa kokenut käyttöliittymäsuunnittelija tai käytettävyydsiantuntija ennustaa käyttöliittymäratkaisun ongelmakohtia simuloimalla moneen kertaan erilaisten käyttötilanteiden suorittamista (GUIDe-menetelmä 2008). Arvioija pyrkii selvittämään, mitä toimintoja käyttäjän pitää missäkin vaiheessa tehdä päästäkseen tavoitteeseensa mahdollisimman suoraviivaisesti ja millaisia mentaalisia toimenpiteitä järjestelmän käytön eri vaiheet vaativat (Hanhisalo 2007, 15). TARMO 2.0 -kehitysprojektissa simulointitestausta käytettiin jo suunnitteluvaiheessa. Sen avulla varmistettiin, ettei uusien käyt-

tötapausten lisääminen käyttöliittymäratkaisuun ole tehnyt muiden toimintojen suorittamisesta vaivalloisempaa. Simulointitestauksen menetelmä vaatii arvioijalta asiantuntemusta ja kokemusta, mutta sen avulla saadaan hyvin selville käyttöliittymän suurimpia tehokkuusongelmia ja käyttötilanteiden kannalta huonosti organisoituja tietoja (GUIDe-menetelmä 2008). Koska simulointipohjaisissa asiantuntija-arvioinneissa ei ole mukana järjestelmän loppukäyttäjiä, se ei ole paras työkalu järjestelmän toiminnallisuuteen ja tietosisältöön liittyvien puutteiden selvittämiseen. Näitä puutteita voidaan selventää läpikäyntitestauksen avulla (Hanhisalo 2007, 15).

Käyttöliittymän läpikäyntipalaverit ovat kehitysvaiheessa käytettävyystestejä kevyempi ja joustavampi tapa arvioida prototyypin toimivuutta. Menetelmän etuna on se, että käyttäjät voivat testata järjestelmää jo hyvin varhaisilla kehitysvaiheen luonnoksilla. (GUIDe-menetelmä 2008.) Palavereihin osallistuu tyypillisesti muutamia järjestelmän käyttäjiä sekä ainakin yksi käyttöliittymäasiantuntija, joka ohjailee järjestelmän läpikäyntiä tarkoituksenmukaisesti suuntiin (Laakso & Laakso 2004, 9). Läpikäyntipalavereilla saadaan kustannustehokkaasti selville puuttuvaa toiminnallisuutta ja väärinkäsityksiä työtehtävien kulkujen ymmärtämisessä. Järjestelmän läpikäynnin aikana voi paljastua myös poikkeustapauksia, joihin ei olla aiemmissa kenttätutkimuksissa osuttu. (GUIDe-menetelmä 2008.)

Erityisesti järjestelmän ensimmäisillä käyttökerroilla eteen tulevia opittavuuteen ja intuitiivisuuteen liittyviä ongelmia voidaan selvittää käytettävyystesteillä. Käytettävyystestissä testikäyttäjä tekee itsenäisesti testitehtäviä yrittäen käyttää testattavaa järjestelmää parhaaksi katsomallaan tavalla. Arvioija kirjaa muistiin, miten käyttäjät tekevät suoritti (Laakso & Laakso 2004, 10). Käytettävyystestejä voidaan tehdä jo tuotekehityksen alkuvaiheessa, mutta parhaiten menetelmä soveltuu valmiin käyttöliittymän tai lähes valmiin käyttöliittymäprototyypin testaamiseen. Menetelmä on työläs ja vaatii testitehtävien sekä -laitteiston osalta ennakoivaltuutuksia, mutta tuottaa luotettavaa tietoa käyttöliittymästä ja sen käytettävyysongelmista. (Kuutti 2003, 68 - 73.)

Jokaisen arviointikierroksen jälkeen löytyneet ongelmat korjattiin ja käyttöliittymäkuvia päivitettiin. Koska käyttöliittymäratkaisut ja järjestelmässä tarvittava toiminnallisuus olivat vielä testausvaiheessa olemassa ainoastaan kuvien ja kuvasarjojen muodossa, oli käyttöliittymäratkaisun muuttaminen korjausehdotuksia vastaavaksi nopeaa

ja edullista (Laakso & Laakso 2004, 9). Vaikka muutostöiden tekeminen onkin tavallista helpompaa, käyttöliittymälähtöisen määrittelyn tarkoituksena on ennen kaikkea minimoida tarvittavien testauskierrosten määrä. Koska järjestelmä on alusta asti suunniteltu tukemaan todellisia käyttötilanteita, ei käyttäjätestauksessa yleensä ilmene laajoja muutostarpeita. (GUIDE-menetelmä 2008.)

Järjestelmän käyttöliittymäkuvaus

Kun ensimmäinen lopullinen versio käyttöliittymästä saatiin valmiiksi ja käyttöliittymäratkaisun tärkeimmät käyttötapausketjut olivat testattu vähintään simulointitestauksella tai niitä oli käyty läpi järjestelmän loppukäyttäjien kanssa, yksittäisten näyttökuvien ja suunnitteluvaiheen luonnoksien pohjalta laadittiin järjestelmän käyttöliittymäkuvaus. Käyttöliittymäkuvaus on tavallaan suunnitteluvaiheen lopputulos ja ikään kuin ohjelmiston pienoismalli. Se visualisoi sen, miltä kehitettävän järjestelmän käyttöliittymä tulee näyttämään ja esittelee järjestelmän tietosisältöä sekä toimintologiikkaa. Käyttöliittymäkuvauksen avulla ohjelmistoa voidaan koekäyttää, ennen kun riviäkään on ohjelmoitu. (Laakso & Laakso 2004, 16.)

Käyttöliittymäkuvaus voi koostua yksittäisistä näyttökuvista tai se voi olla toiminnallisempi prototyyppi järjestelmän käyttäjärajapinnasta. Suunnitteluvaiheen näyttökuviiin verrattuna TARMOn käyttöliittymäratkaisusta laadittu käyttöliittymäkuvaus oli tarkempi ja siinä näkyvät tiedot sekä toiminnot olivat niillä paikoilla, missä ne tulevat olemaan myös lopullisessa järjestelmässä. Käyttöliittymäkuvauksen ohella suunnittelutyön tuloksia havainnollistettiin asiakkaalle www-pohjaisen käyttöliittymäprototyypin avulla (ks. liite 2). Eri näyttöjen sisältöjen lisäksi prototyypistä kävi ilmi työnohjausjärjestelmän graafinen ilme ja navigointirakenne. Prototyypin ulkoasun laadittiin XHTML- ja CSS-kuvauskielten avulla. Navigoinnin toiminnallisuus toteutettiin PHP-ohjelmointikieltä käyttäen. Dynaamisten elementtien ohjelmointi lisäsi kehitystyöhön kulunutta aikaa, mutta niiden avulla prototyyppi oli helpommin muokattavissa.

Ennen järjestelmäkehityksen toteutusvaiheen aloittamista käyttöliittymäkuvausta voidaan vielä arvioida esimerkiksi läpikäyntipalavereiden tai käytettävyyystutkimuksen avulla (Laakso & Laakso 2004, 9). TARMO-projektin käyttöliittymäsuunnittelijan tekemien simulointitestausten lisäksi käyttöliittymäratkaisuja katselmoivat ja testasivat kehitysryhmän muut jäsenet. Testattavat käyttöliittymäratkaisut oli kuvattu diasarjojen avulla, minkä lisäksi kalenterinäkömää ja työtehtävän määräämisen toimin-

toja havainnollistettiin tarkemmilla näyttökuvilla. XHTML-käyttöliittymäprototyyppi oli ensimmäinen projektin asiakkaan luona arvioitu käyttöliittymäratkaisu.

5.6 Projektin jatkotoimenpiteet

Projektin alun vaatimusmäärittelyä ja käyttöliittymäsuunnittelua seuraa järjestelmäarkkitehtuurin ja tietokantarakenteiden suunnittelu sekä lopulta varsinainen toteutusvaihe (Pohjonen 2004, 32 - 34). GUIDe-prosessimallilla suunniteltu käyttöliittymäkuvaus toimii syöteenä projektin seuraaville vaiheille ja asettaa tiettyjä vaatimuksia näiden toteutusratkaisujen suunnittelulle. GUIDe ei kuitenkaan ota kantaa toteutusvaiheessa käytettävään prosessimalliin, vaan projektia voidaan jatkaa esimerkiksi vesiputousmallin vaiheita noudattaen. (Laakso & Laakso 2004, 10.)

Systemaattisen käyttöliittymäsuunnitteluvaiheen tavoitteena oli tuottaa täsmällinen käyttöliittymäkuvaus, jonka tarkoituksena oli havainnollistaa, miltä lopullinen työohjausjärjestelmä tulee näyttämään ja esittää kehitysryhmän laatiman vaatimusmäärittelyn tulokset mahdollisimman yksiselitteisesti. Koska käyttäjien työkuvaukset ja käyttöliittymäkuvat ovat se osa järjestelmää, jota asiakkaat sekä järjestelmän käyttäjät ymmärtävät, käyttöliittymälähtöinen määrittelyprosessi edistää osapuolten välistä yhteisymmärrystä ja pienentää vääränlaisen käyttöliittymän tai järjestelmän tuottamiseen liittyviä riskejä (Laakso & Laakso 2004, 10 - 12).

Käyttöliittymäsuunnittelun jälkeen TARMO 2.0 -kehitysprojektia jatkettiin tietokantarakkitehtuurin suunnittelulla ja koodauksen aloittamisella. Tuotannon tulisi olla valmis joulukuussa 2008. Tavoitepohjaisen käyttöliittymäsuunnittelun tuottaman käyttöliittymäprototyypin avulla asiakas näki jo elokuussa, ennen toteutuksen aloittamista, millaisen järjestelmän hän tulee projektin päätyttyä samaan ja sovelluksen käyttäjät pystyivät ehdottamaan prototyyppiin tehtäviä parannuksia. Tämän toivottiin vähentävän kehitysprojektin seuraavissa vaiheissa esitettäviä korjauspyyntöjä. Liian myöhään paljastuvat muutostarpeet ovat etenkin vesiputousmallin yleinen ongelma (mts. 12).

Tarvittavat tiedot ja toiminnallisuudet sisältävä sekä käyttäjillä testattu käyttöliittymäprototyyppi toimi projektin seuraavien vaiheiden lähtökohtana. Toteutustekniset ratkaisut suunnitellaan valmiin käyttöliittymän päälle, joten ne eivät pääse vaikuttamaan

lopullisen järjestelmän käytettävyyteen. Mikäli järjestelmän määrittelyyn tarvitaan käyttöliittymäprototyypin lisäksi myös muita kuvaustapoja, kuten tietokannan tietomalleja tai UML-käyttötapauksia, ne voidaan laatia käyttöliittymäkuvauksen perusteella (GUIDe-menetelmä 2008). Systemaattisen käyttöliittymäsuunnittelun prosessimallissa toimitaan siis päinvastaisessa järjestyksessä moniin muihin nykykäytäntöihin verrattuna: ensin suunnitellaan käyttöliittymä, josta poimitaan tietosisällöt ja toiminnalliset vaatimukset toimintokuvauksia sekä järjestelmän arkkitehtuurimallia varten. (Laakso 2006, 20.)

Täsmällisestä käyttöliittymäkuvauksesta huolimatta toteutuksen suunnittelun ja koodauksen edetessä käyttöliittymäratkaisuihin voidaan joutua tekemään joitakin kompromisseja. Joidenkin käyttäjän kannalta erinomaisien ratkaisujen toteutus voi olla liian vaivalloista ja kallista, minkä vuoksi käytettävyyden tasosta voidaan joutua hieman tinkimään. Kun toteutusvaiheen kompromissin tarve näyttää ilmeiseltä, ongelma kuvataan käyttöliittymäsuunnittelijalle ja kompromissiratkaisun suunnittelu delegoidaan hänelle. Tällöin ohjelmoijan ei tarvitse ylittää osaamisalueitaan ja kompromissiratkaisu saadaan optimoitua käyttäjän näkökulmasta mahdollisimman hyväksi. (Laakso & Laakso 2004, 11.)

6 JOHTOPÄÄTÖKSET JA POHDINTA

Tietotekniset laitteiden ja sovellusten kehitys on viimeisten vuosikymmenien aikana ollut nopeaa ja ne ovat vallanneet yhä enemmän alaa lähes jokaisella elämän osa-alueella. Järjestelmien tehtävänä on tehostaa yhä monimuotoisemman, eri-ikäisistä ja koulutustaustaisista ihmisistä koostuvan käyttäjäkunnan työtehtäviä sekä helpottaa päivittäisiä rutiineja. Tekniikan kehittymisen myötä keskustelu tietojärjestelmien soveltuvuudesta käyttäjilleen on noussut merkittävässä määrin esille ja niiden suunnittelussa on pyritty kiinnittämään yhä enemmän huomiota käyttäjien ominaisuuksiin sekä ihmisen ja tietokoneen välisen vuorovaikutuksen tutkimiseen. Aiheen ajankohtaisuus tuli jälleen kerran ilmi lokakuussa 2008, kun kunnallisvaalien yhteydessä kokeiltiin sähköistä äänestämistä kolmessa Uudenmaan vaalipiirin kunnassa. Kokeilun seurauksena 232 annettua ääntä jäi kirjautumatta sähköiseen vaaliuurna. Äänestysjärjestelmän toimittaja TietoEnator Oyj koekäytti järjestelmää vain omalla henkilöstöllään ja eri paikkakuntien vaalivirkailijoilla, eikä käyttäjätestien yhteydessä tullut esille mahdollisuutta, että äänestäjä voisi keskeyttää koneella äänestämisen erehdyksessä. (Sähköinen äänestäminen 2008; Sähköistä äänestystä ei testattu ulkopuolisilla 2008; yli 200 sähköistä ääntä hukkaan.)

Oikeusministeri Tuija Braxin mukaan äänien katoaminen ei johtunut järjestelmästä, vaan vika oli laitetta käyttäneissä ihmisissä (Brax: Vaalitulosta ei voi perua hukkaan äänien takia 2008). Tämä ei kuitenkaan pidä paikkaansa, sillä ongelma olisi voitu välttää paremmalla käyttöliittymäsuunnittelulla ja testaamalla järjestelmää sen todellisilla loppukäyttäjillä. Tässä tutkimuksessa selvitettiin, mitä on käytettävyyys, mitä vaikutuksia sillä on tietojenkäsittelytoiminnan onnistumiseen ja millä menetelmillä käytettävyyttä kehitetään. Näitä menetelmiä kokeiltiin käytännössä tutkimuksen kohteena olleen tuotekehitysprojektin käyttöliittymäsuunnitteluprosessin avulla.

Käytettävyyys on tuotteen laatuominaisuus ja Nielsenin (1993, 25) mukaan yksi järjestelmän hyväksyttävyyden tekijöistä, jota ilman tuote ei voi olla hyödyllinen. Käytettävyydeltään korkeatasoinen järjestelmä tekee tietyn käyttäjän tietystä käyttöympäristössä suorittamien tiettyjen tehtävien tekemisestä tuloksellista, tehokasta ja miellyttävää. Käytettävä järjestelmä saa aikaan hyödyllisyyden ja helppokäyttöisyyden tunteen,

mikä kasvattaa käytön määrää ja edistää käyttäjän työskentelytehokkuutta entisestään. Käytettävyyden kehittämisessä ei aina pyritä mahdollisimman helppokäyttöisen järjestelmän tuottamiseen, vaan käytettävyyden eri osatekijöiden merkitys riippuu käyttökontekstista. Toisinaan keskeisin käytettävyyden mittari voi olla esimerkiksi tehokkuus. Liikkuvan työn ohjausjärjestelmä TARMO 2.0:n käyttäjäkunnan tietoteknisen osaamistason vaihtelevuuden vuoksi suunnittelussa painotettiin tuotteen selkeyttä ja helppokäyttöisyyttä. Käytettävyyttä kehitettäessä on hyvä muistaa, että tuotteen lopputekijä muodostaa kuvan järjestelmän toiminnasta sen käyttöliittymän perusteella. Sovelluksen kehittäjät tuntevat käyttöliittymän lisäksi järjestelmän sisäisen rakenteen, minkä vuoksi suunnittelijan mentaalimalli järjestelmän toiminnasta on erilainen kuin käyttäjällä. Tämän takia onnistuneen suunnittelutyön tueksi tarvitaan palautetta kehitysryhmän ulkopuolisilta käytettävyyssammattilaisilta ja ennen kaikkea järjestelmän tulevilta käyttäjiltä.

Käytettävyyden kehittäminen on laadunvarmistuskeino, jonka avulla pyritään tehostamaan järjestelmäkehitysprosessia ja järjestelmän käyttöä sekä pienentämään järjestelmän käyttöönottovaiheessa ilmenevää muutosvastarintaa. Käytettävyyttä ei voida kehittää millään yksittäisellä toimenpiteellä, eikä sitä voida liimata tuotteen pinnalle jälkeinpäin. Käytettävyyden kehittäminen on prosessi, joka alkaa projektin alussa ja jatkuu läpi koko järjestelmäkehityksen elinkaaren.

Systemaattisen käytettävyyden kehittämisen lähtökohtana on käyttäjän ja käyttöympäristön tunteminen. Kehitysprojektin aluksi selvitetään, keitä ovat kehitettävän järjestelmän käyttäjät ja tutustutaan heidän työympäristöönsä sekä työtoimintaansa liittyviin prosesseihin. Selvitetyn aineiston pohjalta määritellään järjestelmän käyttäjäryhmät sekä siihen kohdistuvat vaatimukset. Käytettävyyden kehittämisprosessin kannalta on oleellista, että samanaikaisesti määritellään myös ohjelmiston käytettävyyssvaatimukset ja asetetaan ne mitattavaan muotoon. Määrittelyvaiheessa asetettujen vaatimuksien perusteella suunnitellaan käyttöliittymän lopullinen rakenne ja ulkoasu. Suunnitteluvaiheessa ei huomioida järjestelmän teknistä toteutusta, vaan pyritään laatimaan mahdollisimman hyvin käyttäjien tehtäviä ja tavoitteita tukeva käyttöliittymäratkaisu. Suunnittelutyö on iteratiivista ja tuotettuja ratkaisuja on testattava järjestelmän todellisilla käyttäjillä. Asetettujen käytettävyyssvaatimusten ja käyttäjien todellisiin työnkuluihin perustuvan suunnitteluprosessin avulla lopputuotteen käyttäjystävällisyys voi-

daan varmistaa jo määrittelyvaiheessa. Laadittu käyttöliittymäratkaisu toimii lähtökohtana järjestelmän toteutuksen suunnittelulle.

Käyttäjakeskeisen suunnittelutyön avuksi on kehitetty erilaisia työkaluja ja menetelmiä, joiden avulla voidaan tutkia käyttäjien työtoimintaa ja mallintaa käyttökontekstia sekä järjestelmään kohdistuvia vaatimuksia. Määrittelyvaiheen alussa käyttäjistä ja heidän työskentelystään pyritään saamaan ensikäden tietoa, mikä ei onnistu vain kyselyjä ja haastatteluja tekemällä. Niiden tilalla käytettäväksi kenttätutkimuksen menetelmiksi suositellaan muun muassa käyttäjätarkkailua sekä kontekstuaalista haastattelua. Kehitettävän järjestelmän tulevia käyttäjiä voidaan mallintaa esimerkiksi käyttäjäryhmä- ja käyttäjäpersoonakuvausten avulla. Käyttäjäpersoonat kuvaavat jokaisen ryhmän tyypillistä edustajaa, ja ne helpottavat kunkin käyttäjäryhmän huomioimista suunnittelun myöhemmissä vaiheissa. Järjestelmään kohdistetuista vaatimuksista johdetaan vapamuotoisesti kirjoitettuja käyttötapauksia. Käyttötapaukset eivät kuvaa sovelluksen käyttöä tai minkään tietyn toiminnon suorittamista, vaan niiden avulla pyritään mallintamaan käyttäjien ongelmia ja korkean tason tavoitteita. Suunnitellun käyttöliittymäratkaisun sisältöä ja ulkoasua voidaan havainnollistaa muun muassa käsin piirretyillä käyttöliittymäkuvilla sekä eritasoisilla käyttöliittymäprototyypeillä. Käyttöliittymän ominaisuuksia ja sen soveltumista käyttäjien työnkulkuhiin on hyvä arvioida mahdollisimman aikaisin, sillä matalan tason käyttöliittymäprototyypeillä iterointi on nopeaa ja kustannustehokasta. Suunnitteluvaiheen arviointeihin soveltuvat parhaiten kevyet ja joustavat testausmenetelmät, kuten käytettävyyssasiantuntijoiden tekemät simulointitestaukset sekä läpikäyntipalaverit. Jotta suunnitteluprosessi olisi käyttäjakeskeinen, järjestelmän käytettävyyttä on testattava myös järjestelmän todellisten loppukäyttäjien kanssa. Prototyypin helppokäyttöisyyttä ja käytettävyyden onnistumista voidaan arvioida käyttäjien kanssa suunnitteluvaiheen lopussa tehtävien käytettävyytestien avulla.

Systemaattisesta käytettävyyden kehittämisestä ja käyttäjien aktiivisesta huomioimisesta hyötyvät sekä järjestelmän kehittämisestä vastannut ohjelmistotalo että valmista sovellusta käyttävä asiakasorganisaatio. Asiakasyritykselle käytettävyyden tuomat edut näkyvät järjestelmän käytön tehostumisena ja tietotekniikkainvestoinneista koituvien ylimääräisten kustannuksien vähenemisenä. Helppokäyttöinen ja opittava sovellus vähentää uuteen järjestelmään kohdistuvaa muutosvastarintaa ja yrityksen kan-

nalta kustannusmielessä tehottoman perehdyttämiskoulutuksen tarvetta. Tehokas tuote vähentää tuottamattoman työn tekemiseen kuluvaa aikaa ja edistää työntekijöiden työskentelymotivaatiota. Miellyttävä järjestelmä voi vähentää työntekijöiden stressiä ja sen myötä ehkäistä jopa mahdollisia sairauspoissaoloja.

Kehittäjäorganisaatio hyötyy ennen kaikkea käytettävyyssosaamisen ja käytettävyydeltään korkeatasoisten tuotteiden luomasta kilpailuedusta. Käyttäjäkeskeisen tuotekehityksen menetelmät tehostavat yrityksen ohjelmistokehitysprosessia ja vaikuttavat positiivisesti kehitettävien tuotteiden laadukkuuteen. Käyttäjän tarpeiden pohjalta suunniteltua järjestelmää on myös helpompi markkinoida: Kun tunnetaan potentiaalisten käyttäjien sovellusala ja tarpeet, tuotetta on pystytään mainostamaan juuri niillä ominaisuuksilla, joita ostaja pitää kaikista tärkeimpinä. Käyttäjäkeskeisen suunnittelun menetelmät saattavat pidentää yksittäisen projektin määrittelyvaihetta, mutta koska kenttätutkimukset edistävät oikeiden vaatimusten löytymistä heti ensimmäisellä kerralla ja havainnollinen prototyyppi helpottaa asiakkaan kanssa käytäviä keskusteluja kehitettävistä ominaisuuksista, niin projektin myöhemmissä vaiheissa esiintyvät muutostarpeet vähenevät. Muutostarpeiden väheneminen puolestaan tehostaa kehitysprojektin toteutusvaihetta. Käyttäjäkeskeisen tuotekehityksen menetelmät edellyttävät koko organisaatiolta uudenlaista lähestymistapaa ohjelmistokehitykseen. Lisäksi tuloksellinen käytettävyystyö vaatii laaja-alaista käytettävyyssosaamista ja riittävästi resursseja koko projektin ajan kestäväan käytettävyyden kehittämiseen.

Liikkuvan työn ohjausjärjestelmä TARMO 2.0 -käyttöliittymäsuunnitteluprojektin tavoitteena oli määrittää työnohjausjärjestelmän käyttäjien tarpeet ja luoda käyttöliittymäkuvaus, joka toimii lähtökohtana mahdollisimman hyvin näitä tarpeita vastaavan tietojärjestelmän kehittämiseksi. Kehitystyötä rajoitti projektiryhmän koko, joka osaltaan vaikutti myös siihen, miten menestyksekkäästi tavoitepohjaisen käyttöliittymäsuunnittelun menetelmiä voitiin projektissa hyödyntää. Esimerkiksi erillisten suunnittelu- ja toteutusryhmien perustaminen ei ollut pienessä projektissa aivan ongelmattonta, joten käyttäjätutkimukseen sekä tutkimusaineiston analysointiin osallistuivat kaikki kehitysryhmään kuuluvat henkilöt. Tavoitepohjaisten käyttötapauksen laatiminen ja käyttöliittymäratkaisun suunnittelun oli täysin tutkijan vastuulla.

Käyttäjätutkimusvaiheessa asiakasyrityksen työntekijöitä ei pystytty tarkkailemaan heidän työskentelyn ohessa. mutta haastattelut tehtiin yrityksen toimipisteessä, jossa

työntekijät näyttivät keskeisimpien työtehtävien suorittamista työnohjausjärjestelmän nykyisellä versiolla. Näin päästiin hyvin lähelle käyttäjien todellisia työtilanteita. Tutkimus osoitti, että nykyinen järjestelmä vastasi toiminnallisuuksiltaan ja tietosisällöltään työnteon tarpeita, joten uuden version kehitystyön pohjana voitiin käyttää jo olemassa olevaa vaatimusmäärittelyä. Haastattelujen avulla keskityttiin selvittämään minkälaisia TARMOn nykyisen version tarjoamia työnkulkuja käyttäjät ovat työnsä tekemiseen omaksuneet. Selvitettyjen työnkulkujen pahimpia mutkia pyrittiin suorittamaan suunnitteluvaiheessa. Suunnitteluryhmä koki sovellusalan tutkimisen ja käyttäjien työtoimintaan tutustumisen lisänsen ymmärrystä käyttäjien tarpeista, mikä auttoi käyttäjävaatimuksien priorisoinnissa ja käyttöliittymäratkaisujen suunnittelussa. Ilman käyttäjä tutkimusta oikeiden vaatimusten löytyminen olisi todennäköisesti kes- tänyt pidempään ja suunnitteluvaiheen muutospyynnöt lisääntyneet.

Projektin aikana tavoitepohjaiset käyttötapauskuvaukset laadittiin työnohjaukseen, eli työtehtävien ja työvuorojen lisäämiseen sekä muokkaamiseen liittyvistä työnkuluista. Asiakas- ja työntekijärekisterin tietojen käsittelyyn liittyvien toimintojen tehostaminen ei ollut ensisijaisen tärkeää, joten kyseisten näyttöjen suunnittelua varten keskityttiin selvittämään mitä tietoja järjestelmään tallennetaan. Erillisten näyttöjen lisäksi suunnittelun tavoitteena oli kehittää järjestelmälle mahdollisimman selkeä, mutta työnkulkua tehokkaasti tukeva navigointirakenne. Käyttöliittymän selkeys pyrittiin varmistamaan simulointipohjaisella suunnittelumenetelmällä, jonka tarkoituksena on ehkäistä työskentelyn kannalta ylimääräisten toimintojen syntyminen.

Kokonaisuudessaan käyttöliittymäratkaisujen tuottaminen ei täysin toteutunut GDD-suunnitteluprosessin mukaisesti. Luonnoksia laadittaessa suunnittelija turvautui käyttötapausten vaiheittaisen simuloinnin sijasta helposti omaan intuition ja kokemukseen. Työtehtävien suorittamista simuloitiin käyttötapausten kiinnittämisen välissä, mutta käyttötapausten yksittäisiä tietoja ja toiminnallisuuksia ei aina piirretty näkyviin niin systemaattisesti, kuin suunnitteluprosessissa on tarkoitus. Suunnitteluvaiheessa ideoitii monia käyttäjän työtoimintaa helpottavia käyttöliittymäratkaisuja, jotka asettivat uusia vaatimuksia muun muassa toteutusvaiheessa käytettävälle ohjelmointitekniikoille. Hyvissä ajoin laadittu käyttöliittymäkuvaus helpotti näiden ratkaisujen käytännön toteutuksen suunnittelua, jolloin käyttäjän kannalta hyvien ideoiden toteutusmahdollisuudet paranivat.

Projektiaikataulusta johtuen tutkimuksessa pystyttiin käsittelemään TARMO 2.0 -kehitysprojektia vain käyttöliittymäsuunnittelun osalta, joten vielä on mahdoton sanoa, kuinka käyttöliittymäkuvauksessa esitetyt ratkaisut pystyttiin toteuttamaan käytännössä. Luonnollinen aihe opinnäytetyön jatkotutkimukselle olisikin selvittää, millaisen lähtökohdan systemaattinen käyttöliittymäsuunnittelu antoi kehitysprojektin toteutukselle, ja mitä kompromisseja toteutusvaiheessa jouduttiin tekemään. Toinen mielenkiintoinen jatkotutkimusaihe olisi tarkastella tavoitepohjaisen käyttöliittymäsuunnittelun konkreettisia vaikutuksia tuotteen tehokkuuteen ja helppokäyttöisyyteen verrattuna työnohjausjärjestelmän ensimmäiseen versioon. Objektiiivisesti mitattavia tuloksia työnohjausjärjestelmän eri versioiden käytettävyysongelmista voitaisiin hankkia esimerkiksi kattavien käytettävyytestien avulla.

Suomessa on vahvaa käyttöliittymäsuunnittelun, käytettävyystudiumin ja teollisen muotoilun osaamista sekä yliopistoissa että yksityisellä sektorilla. Kognitiotieteitä ja ihmisen sekä tietokoneen välistä vuorovaikutusta tutkitaan muun muassa Jyväskylän yliopistossa sekä Tampereen yliopiston tietojenkäsittelytieteiden laitoksen TAUCHI-yksikössä. Tutkimus avasi uuden näkökulman ohjelmistotuotantoon ja tarjosi mielenkiintoisen sovellusalueen tietojärjestelmien kehittämisen metodeille. Käyttäjakeskeisen suunnittelun menetelmiä sovellettiin tutkimuksen aikana myös käytännössä, mutta vasta työssä esitetyn teorian ja kehitystyön lopullisten tulosten arvioinnin perusteella voidaan suunnittelumenetelmiä ja työkaluja muokata paremmin toimeksiantajan käyttöön sopiviksi.

LÄHTEET

Alanne, M. 2002. Käytettävyyden kehittämisprosessin uudistaminen isolle ohjelmistotalolle. Pro gradu -tutkielma. Helsingin kauppakorkeakoulu, tietojärjestelmätiede. Viitattu 27.5.2008. <http://www.soberit.hut.fi/T-121>, opinnäytekirjasto.

ATK-sanakirja. 2008. MOT ATK-sanakirja 4.0. Kielikone Oy. Viitattu 19.10.2008. <http://www.kielikone.fi>.

Brax: Vaalitulosta ei voi perua hukkaäänien takia. 2008. Helsingin sanomat. Artikkelijulk. 28.10.2008. Viitattu 2.11.2008. <http://www.hs.fi/politiikka/artikkeli/Brax+Vaalitulosta+ei+voi+perua+hukka%C3%A4%C3%A4nien+takia/1135240616263>.

Controla Oy. 2008. Yritysesittely. Viitattu 27.10.2008. <http://www.controla.fi>, Controla Oy.

GUIDe-menetelmä. 2008. Interacta Design Oy. Viitattu 21.4.2008. <http://www.interacta.fi>.

Hanhisalo, J. 2007. Määrittelyvaiheen vaatimusten vaikutus käyttöliittymään. Pro gradu -tutkielma. Helsingin yliopisto, tietojenkäsittelytieteen laitos. Viitattu 10.5.2008. <http://urn.fi/URN:NBN:fi-fe200802191096>.

Isomäki, H. 2006. Käytettävyyden vaikuttavuus SIGCHI Finland ry:n jäsenistön näkökulmasta. SIGCHI Finland. Viitattu 13.9.2008. <http://www.sigchi.fi/toiminta/SIGCHI-raportti.pdf>.

Isotalo, K. 2007. Käytettävyys tuo työniloa. TeklaNews 2007. Artikkelijulk. luettavissa myös sähköisesti. Viitattu 13.9.2008. http://www.tekla.com/fi/about-us/news/magazine/previous/Documents/TeklaNews_2007.pdf.

Kotihoidon langaton työnohjaus- ja raportointijärjestelmä. 2007. Vaatimusmäärittelydokumentti. Seasam Service Oy.

Kujala, S. 2006. Käyttäjäkeskeinen suunnittelu. Systemityö-lehti 2006, 2, 9 - 11. Artikkelijulk. luettavissa myös sähköisesti. <http://www.ttlry.fi/yhdistykset/sytyke>, Systemityö-lehti.

Kuoppala, H. & Nieminen, M. 1998. Standardin mukainen käytettävyys – tulevaisuuden vaatimus?. Systemityö-lehti 1998, 4, 19 - 21. Artikkelijulk. luettavissa myös sähköisesti. <http://www.ttlry.fi/yhdistykset/sytyke>, Systemityö-lehti.

Kuparinen, L. 2008. Käytettävyyden merkitys ohjelmiston valinnassa. Pro gradu -tutkielma. Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos. Viitattu 4.8.2008. <http://urn.fi/URN:NBN:fi:ju-200805215349>.

Kuutti, W. 2003. Käytettävyys, suunnittelu ja arviointi. Helsinki: Talentum.

- Käyttöliittymän määrittelyä. 2005. Virtuaali ammattikorkeakoulu. Luentomateriaali. Viitattu 4.8.2008. <https://www.amk.fi/opintojaksot/030308/1111676348138.html.stx>.
- Laakso, S. 2006. Käyttöliittymälähtöinen vaatimusmäärittely. *Systeemityö-lehti* 2006, 2, 19 - 21. Artikkelin luettavissa myös sähköisesti. <http://www.ttlry.fi/yhdistykset/sytyke>, *Systeemityö-lehti*.
- Laakso, S. & Laakso, K-P. 2004. Hyvän käyttöliittymän varmistaminen GUIDE-prosessimallilla. Viitattu 21.4.2008. <http://www.cs.helsinki.fi/u/salaakso/papers/GUIDE-suomeksi.pdf>.
- Laakso, S. & Latva-Koivisto, A. 2006. Käyttöliittymät II: luentomonisteet. Helsingin yliopisto, tietojenkäsittelytieteen laitos. Viitattu 27.5.2008. <http://www.cs.helsinki.fi/u/salaakso/kl2-2006>.
- Liikkuvan työn ohjausjärjestelmä TARMO. 2008. Tuote-esittely. Controla Oy.
- Lilja, T. 2005. Roolikuvauksiin perustuva lähestymistapa käyttäjävaatimusten selvittämisessä. Diplomityö. Teknillinen korkeakoulu, tietotekniikan osasto. Viitattu 12.9.2008. <http://www.soberit.hut.fi/T-121>, opinnäytekirjasto.
- Lohikoski, N. 2007. Myynnin tukijärjestelmien kehittäminen liiketoimintaa tukevaksi kokonaisuudeksi. Diplomityö. Teknillinen korkeakoulu, automaatio- ja systeemitekniikan osasto. Viitattu 27.5.2008. http://www.media.hut.fi/~julkaisut/diplomityot/DI_N_Lohikoski_2007.pdf.
- Makkonen, P. 2006. Tietohallinto ja tietojärjestelmien kehittämisen perusteet. Luentomoniste. Jyväskylän yliopisto, tietojenkäsittelytieteiden laitos. Viitattu 7.9.2008. <http://users.jyu.fi/~pmakkone/tjta111/>.
- Malli terveydenhuollon toimintayksikön tietosuojaohjelmien rungoksi. 2000. Tietosuojavaltuutetun malli 2000, 2. Tietosuojavaltuutetun toimisto. Viitattu 17.9.2008. <http://www.tietosuoja.fi/7260.htm>, Tietosuojavaltuutetun mallit -sarja.
- Nielsen, J. 1993. Usability Engineering. Boston: Academic Press.
- Onko käytettävyyteen panostamisessa järkeä?. 2007. Avania Consulting. Viitattu 27.5.2008. <http://www.avania.fi>, artikkelit.
- Parkkinen, J. 2001. Käytettävyys, mitä se on?. *Hila-verkkolehti* 2001. Artikkelin luettavissa myös sähköisesti. <http://www.adage.fi/julkaisut/arkisto>.
- Pohjonen, R. 2002. Tietojärjestelmien kehittäminen. 2. p. Jyväskylä: Docendo.
- SFS-EN ISO 13407. 2003. Vuorovaikutteisten järjestelmien käyttäjäkeskeinen suunnitteluprosessi. Helsinki : Suomen Standardisoimisliitto SFS.
- SFS-EN ISO 9241-11. 2000. Näyttöpäätteillä tehtävän toimistotyön ergonomiset vaatimukset. Osa 11: Käytettävyyden määrittely ja arviointi. Helsinki : Suomen Standardisoimisliitto SFS.
- Sinkkonen, I. 2006. Käyttöliittymät ja käytettävyys. Adage Usability -julkaisu. Viitattu 20.5.2008. <http://www.adage.fi/julkaisut/arkisto>.

Sinkkonen, I., Kuoppala, H., Parkkinen, J. & Vastamäki, R. 2006. Käytettävyyden psykologia. 3. uud. p. Helsinki: Edita, IT Press.

Sähköinen äänestäminen. 2008. Oikeusministeriö. Viitattu 2.11.2008.

[Http://www.vaalit.fi/42321.htm](http://www.vaalit.fi/42321.htm).

Sähköistä äänestystä ei testattu ulkopuolisilla. 2008. Helsingin sanomat. Artikkelijulk. 29.10.2008. Viitattu 2.11.2008.

[Http://www.hs.fi/politiikka/artikkeli/S%C3%A4hk%C3%B6ist%C3%A4+%C3%A4%C3%A4nestyst%C3%A4+ei+testattu+ulkopuolisilla/1135240638305](http://www.hs.fi/politiikka/artikkeli/S%C3%A4hk%C3%B6ist%C3%A4+%C3%A4%C3%A4nestyst%C3%A4+ei+testattu+ulkopuolisilla/1135240638305).

Toivanen, M., Luukkonen, I., Ensio, A., Häkkinen, H., Ikävalko, P., Jaatinen, J., Klemola, L., Korhonen, M., Martikainen, S., Miettinen, M., Mursu, A., Röppänen, P., Silvennoinen, R., Tuomainen, T. & Palmén, M. 2007. Kohti suunnitelmallisia muutoksia: opas terveydenhuollon tietojärjestelmien toimintalähtöiseen kehittämiseen. Kuopio. Kuopion yliopisto. Viitattu 21.4.2008. [Http://www.uku.fi/zipit/](http://www.uku.fi/zipit/).

Viitanen, J. 2005. Vaatimusmäärittely käyttäjäkeskeisessä tuotekehityksessä – Lähtökohtana verkkopohjaisen oppimisympäristön suunnittelu. Diplomityö. Teknillinen korkeakoulu, sähkö- ja tietoliikennetekniikan osasto. Viitattu 17.9.2008.

[Http://www.soberit.hut.fi/T-121](http://www.soberit.hut.fi/T-121), opinnäytekirjasto.

Vuori, M. 1999. Havaintoja suomalaisen tuotekehityksen ja käyttöliittymäsuunnittelun tilasta ja kehittämistarpeista. Käyttäjäkeskeisen tuotekehityksen tietotuki –työraportti 14. VTT Automaatio. Viitattu 16.10.2008.

[Http://www.kotiposti.net/~xmvuori/julkaisuluettelo](http://www.kotiposti.net/~xmvuori/julkaisuluettelo), tuotekehitys.

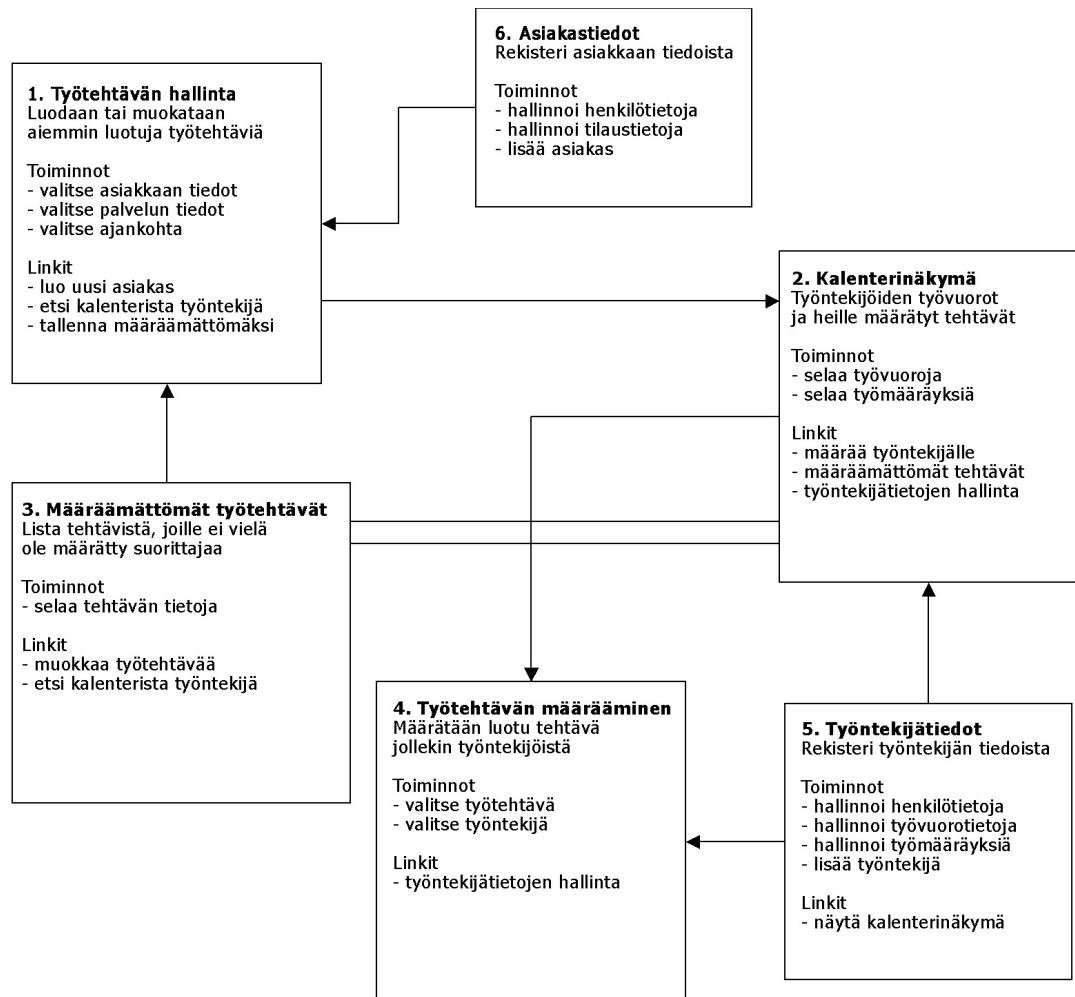
Wiio, A. 2004. Käyttäjäystävällisen sovelluksen suunnittelu. Helsinki: Edita.

Yli 200 sähköistä ääntä hukkaan. 2008. Helsingin sanomat. Artikkelijulk. 28.10.2008. Viitattu 2.11.2008.

[Http://www.hs.fi/politiikka/artikkeli/Yli+200+s%C3%A4hk%C3%B6ist%C3%A4+%C3%A4%C3%A4nt%C3%A4+hukkaan/1135240610727](http://www.hs.fi/politiikka/artikkeli/Yli+200+s%C3%A4hk%C3%B6ist%C3%A4+%C3%A4%C3%A4nt%C3%A4+hukkaan/1135240610727).

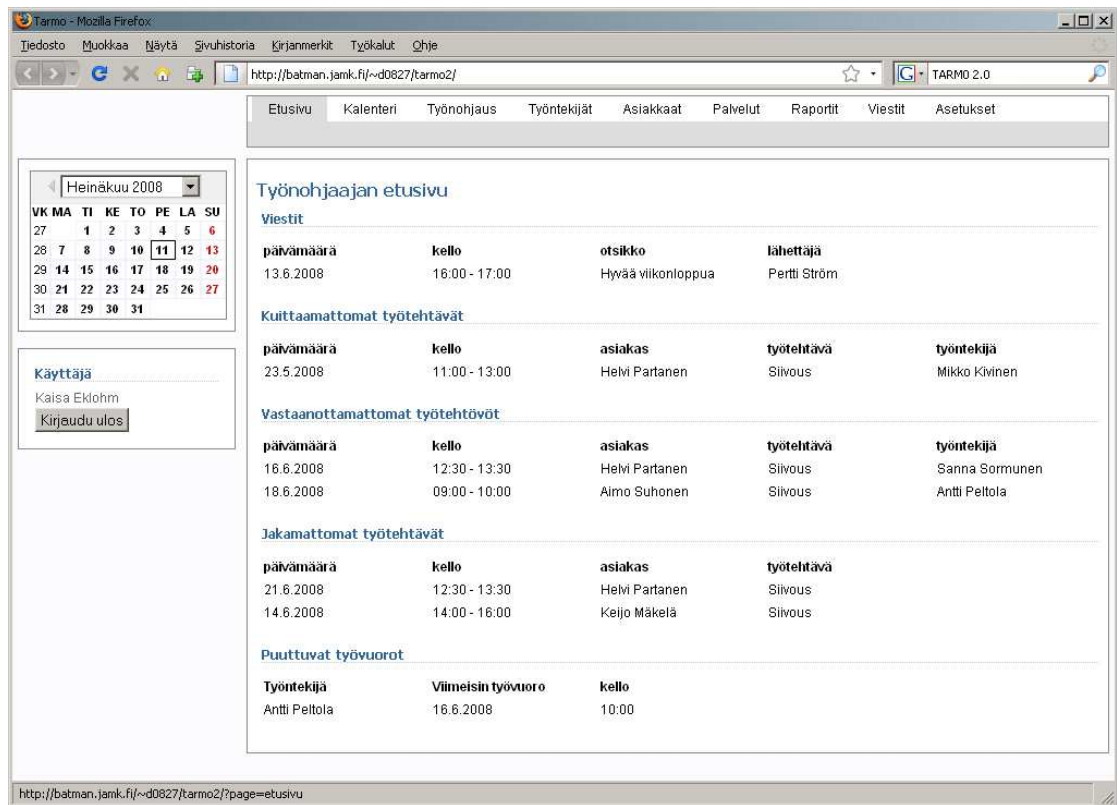
LIITTEET

Liite 1. Käyttöliittymän UED-kaavio



Liite 2. Käyttöliittymäkuvat

Kuvat kehitystutkimuksen lopputuloksena tuotetun Liikkuvan työn ohjausjärjestelmä TARMO 2.0 -käyttöliittymäprototyypin tärkeimmistä näytöistä.



Kuva 1. Työnohjaajan etusivu. Kirjautumisen jälkeen käyttäjä ohjataan etusivulle, joka kokoaa yhteen määrättyjen ja määräämättömien työtehtävien tilatiedot, työntekijöiltä puuttuvat työvuorot sekä järjestelmän sisällä lähetetyt viestit ja tiedotteet.

The screenshot shows a web browser window displaying a daily calendar application. The browser's address bar shows the URL: `http://batman.jamk.fi/~d0827/tarmo2/?subpage=paivakalenteri&tyontekijat=pilota`. The application has a navigation menu with options: Etusivu, Kalenteri, Työnohjaus, Työntekijät, Asiakkaat, Palvelut, Raportit, Viestit, and Asetukset. Below this, there are links for Päiväkalenteri, Viikkokalenteri, and Työvuorokalenteri.

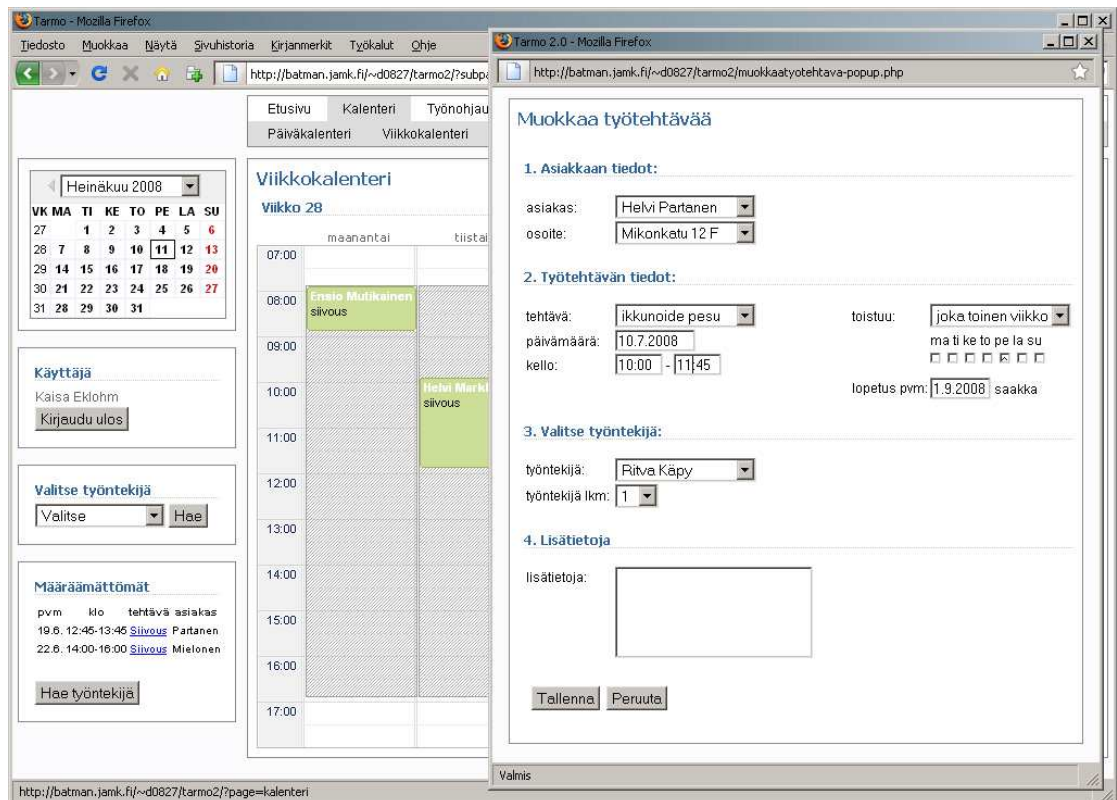
The main content area is titled "Päiväkalenteri" and shows the date "11.7.2008". On the left, there is a date selector for "Heinäkuu 2008" and a calendar grid. Below the calendar, there is a section for "Käyttäjät" (Users) with a list of names: Kaisa Ekholm, and a button "Kirjaudu ulos".

Below the user list, there is a section for "Valitse työntekijät" (Select employees) with a list of names and checkboxes: Suvi Alanko, Kaisa Ekholm, Matti Elonen, Riiva Käpy, Jaana Kelomäki, Tiina Seppänen, Pertti Ström, and Ari Särkkä. A "Näytä" (Show) button is at the bottom of this list.

The main calendar grid shows tasks assigned to specific employees at specific times. The tasks are as follows:

Time	Suvi Alanko	Matti Elonen	Kaisa Ekholm	Ari Särkkä	Pertti Ström	Tiina Seppänen	Riiva Käpy
07:00							
08:00	Ensio Mutikainen siivous			Helmi Markkanen siivous			
09:00							
10:00		Helmi Markkanen siivous		Helmi Markkanen ruuanlaitto	Aune Bergström siivous		
11:00						Helmi Markkanen siivous	
12:00							
13:00			Veli Mustonen ikkunoiden pesu	Ilkka Hopo siivous			
14:00							
15:00			Anu Koponen siivous		Helmi Markkanen siivous		
16:00							
17:00							

Kuva 2. Työntekijöiden päiväkalenteri. Kalenterista näkyvät valittujen työntekijöiden työvuorot ja työtehtävät valitun päivämäärän osalta. Ikkunan vasemmasta reunasta valitaan näytettävä päivä ja ne työntekijät, joiden tietoja halutaan tarkastella.



Kuva 3. Työtehtävän tiedot. Uusi työtehtävä lisätään päivä- tai viikkokalenterin tyhjää solua painamalla. Avautuvaan ponnahdusikkunaan määritellään lisättävän tehtävän solua painamalla. Uusi työtehtävä lisätään päivä- tai viikkokalenterin tyhjää solua painamalla. Avautuvaan ponnahdusikkunaan määritellään lisättävän tehtävän tiedot. Työntekijöille jo ennestään määrättyjen tehtävien ja ikkunan vasemman laidan määräämättömien tehtävien tietoja muokataan samanlaisen ponnahdusikkunan kautta.

Tarmo - Mozilla Firefox

Tiedosto Muokkaa Näytä Sivuhistoria Kirjanmerkit Työkalut Ohje

http://batman.jamk.fi/~d0827/tarmo2/?subpage=uusityovuoro

TARMO 2.0

Etusivu Kalenteri Työnohjaus Työntekijät Asiakkaat Palvelut Raportit Viestit Asetukset

Uusi työtehtävä Uusi työvuoro

Heinäkuu 2008

VK	Ma	Ti	Ke	To	Pe	La	Su
27		1	2	3	4	5	6
28	7	8	9	10	11	12	13
29	14	15	16	17	18	19	20
30	21	22	23	24	25	26	27
31	28	29	30	31			

Käyttäjää

Kaisa Ekholm

Kirjaudu ulos

Luo uusi työvuoro

1. Valitse työntekijät

☒ Suvi Alanko
 ☐ Jaana Kelomäki
☒ Kaisa Ekholm
 ☐ Tiina Seppänen
☐ Matti Elonen
 ☒ Pertti Ström
☐ Ritva Käpy
 ☐ Ari Särkkä

2. Valitse päivä ja työaika

«« Heinäkuu 7/2008 »»»

v.	Ma	Ti	Ke	To	Pe	La	Su
27		1	2	3	4	5	6
28	7	8	9	10	11	12	13
29	14	15	16	17	18	19	20
30	21	22	23	24	25	26	27
31	28	29	30	31			

kello: 7:45 - 16:00

Tallenna Tyhjennä

[takaisin](#)

http://batman.jamk.fi/~d0827/tarmo2/?page=tyonohjaus

Kuva 4. Työvuoron tiedot. Uusi työvuoro lisätään valitsemalla ensin yksi tai useampi työntekijä, joille työvuoro luodaan. Kuukausikalenterista valitaan päivämäärät, jolloin työvuoro on voimassa. Lopuksi määritellään työpäivän pituus.

Tarmo - Mozilla Firefox

Tiedosto Muokkaa Näytä Sivuhistoria Kirjanmerkit Työkalut Ohje

http://batman.jamk.fi/~d0827/tarmo2/?subpage=uusityontekija&muokkaa=yes

TARMO 2.0

Etusivu Kalenteri Työnohjaus Työntekijät Asiakkaat Palvelut Raportit Viestit Asetukset

Uusi työntekijä

Heinäkuu 2008

VK	MA	TI	KE	TO	PE	LA	SU
27	1	2	3	4	5	6	
28	7	8	9	10	11	12	13
29	14	15	16	17	18	19	20
30	21	22	23	24	25	26	27
31	28	29	30	31			

Käyttäjät

Kaisa Ekholm

Kirjaudu ulos

Valitse työntekijä

Valitse Hae

Työntekijän tiedot

1. Henkilötiedot

etunimi: Kaisa

sukunimi: Ekholm

puhelin: 044-5523111

sähköposti: kaisa.ekholm@gmail.com

katuosoite: Polttolinja 14 D 22

postinumero: 40520

postitoimipaikka: Jyväskylä

2. Käyttäjätiedot

käyttäjätunnus: kaisaekholm

salasana:

käyttäjärooli: työnohjaaja

3. Lisätietoja

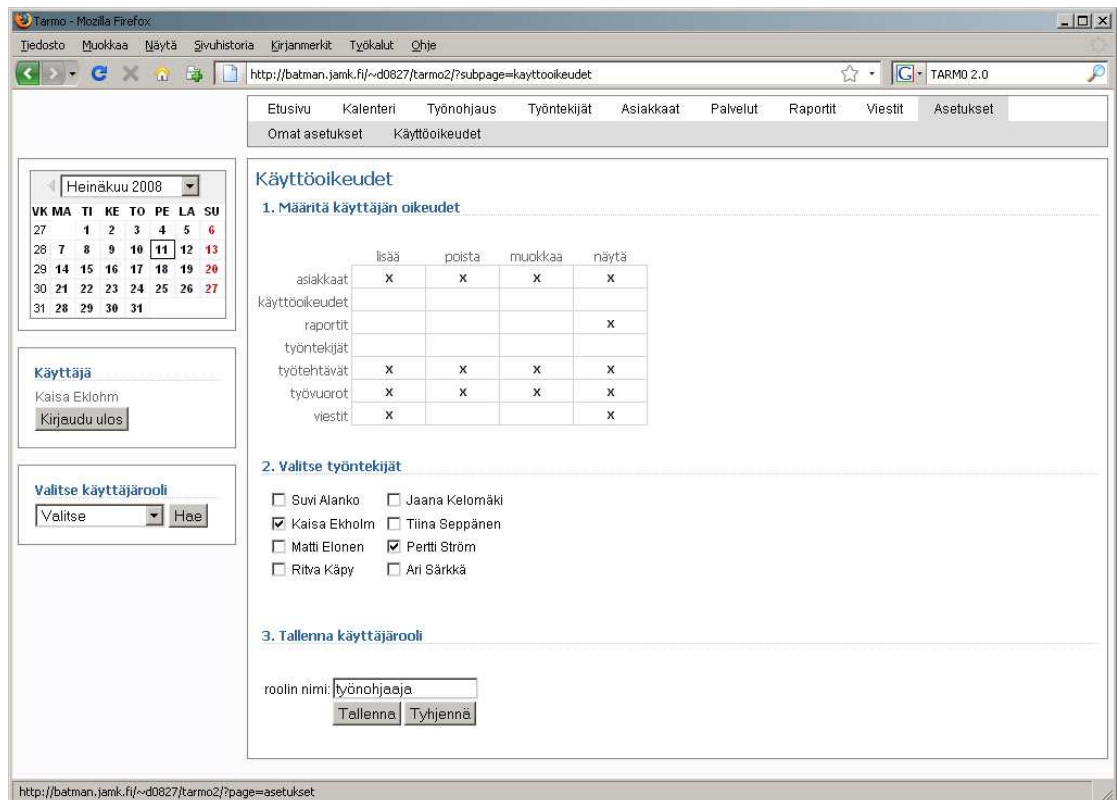
lisätietoja:

Lisää Tyhjennä

[takaisin](#)

http://batman.jamk.fi/~d0827/tarmo2/?page=tyontekijat

Kuva 5. Työntekijän tiedot. Sivua käytetään yrityksen uusien työntekijöiden tietojen lisäämiseen ja työntekijärekisterin tietojen hallintaan. Tiedot on jaoteltu työntekijän henkilötietoihin ja järjestelmän käyttäjätietoihin.



Kuva 6. Käyttöoikeudet. Järjestelmän asetussivulla määritellään kullekin järjestelmän käyttäjälle käyttäjärooli, joka rajoittaa tai sallii eri toimintojen suorittamista. Järjestelmän ylläpitäjä voi määrittellä jokaisen käyttäjän oikeudet erikseen tai luoda valmiita käyttäjärooleja.