

# Asiantuntijajärjestelmän laajentaminen uusille käyttäjärhmille

Petteri Suontama

11. tammikuuta 2005

TEKNILLINEN KORKEAKOULU Tietotekniikan osasto		DIPLOMITYÖN TIIVISTELMÄ	
Tekijä  Petteri Suontama	Päiväys 11.1.2005		
	Sivumäärä 59		
Työn nimi  Asiantuntijajärjestelmän laajentaminen uusille käyttäjärhyhmille			
Professori Käyttöliittymät ja käytettävyys		Koodi T-121	
Työn valvoja Professori Marko Nieminen			
Työn ohjaaja TkL Sirpa Riihiahho, DI Mika P. Nieminen			
<p>Tässä diplomityössä kuvataan, kuinka alun perin pienelle asiantuntijaryhmälle suunnattu matkapuhelinverkkojen simulaatiojärjestelmän käyttöliittymä muokattiin suuremmalle käyttäjäkunnalle sopivaksi.</p> <p>Järjestelmällä oli alun perin 19 käyttäjää. Käyttäjien tarpeita tutkittiin haastattelujen, kyselyn ja käytettävyydestien avulla. Saatujen käyttäjävaatimusten perusteella järjestelmän kehitys tapahtui iteratiivisesti. Tuotteesta rakennettiin prototyyppisiä, joita muokattiin iteraatioissa kohti valmista tuotetta. Tehtyjä prototyyppisiä arvioitiin ja niissä olevia käytettävyysongelmia etsittiin käyttäjäkeskeisten menetelmien avulla. Toteutettu valmis käyttöliittymä tuki alkuperäisten ja uusien käyttäjärhyhmien toimintamalleja. Sen käyttö osoittautui selkeäksi ja helpoksi.</p> <p>Käyttöliittymän kehitys tapahtui tuotteen pääkäyttäjärhyhmän tiloissa osallistuvan suunnittelun mukaisesti. Näin saatiin varmistettua tehdyn käyttöliittymän erinomaisuus sen tärkeimmän käyttäjärhyhmän näkökulmasta.</p>			
Avainsanat käytettävyys, käyttöliittymä, käyttäjäkeskeiset menetelmät, ohjelmistotuotanto			

HELSINKI UNIVERSITY OF TECHNOLOGY Department of Computer Science and Engineering		ABSTRACT OF MASTER'S THESIS	
Author  Petteri Suontama		Date 11.1.2005	Pages 59
Title of thesis Extending an expert system to new user groups			
Professorship User interfaces and usability		Professorship Code T-121	
Supervisor Professor Marko Nieminen			
Instructor Lic.Sc. Tech Sirpa Riihiaho, M. Sc. Techn Mika P. Nieminen			
<p>This master's thesis describes how a simulator system for mobile networks, originally designed for a small group of experts, was modified for a wider range of users.</p> <p>The system had originally 19 users. The needs of the users were studied using interviews, surveys and usability tests. Based on the found user needs, the system development was carried out iteratively. Product prototypes were made and modified in iterations towards the final product. The prototypes were evaluated and user-centered methods were used to find usability problems in the prototypes. The final user interface supported the ways of operation of the original and the new user groups. The use of the product was found to be clear and easy.</p> <p>The development of the user interface took place at the premises of the main user group, like in participatory design. This way it was possible to guarantee the quality of the user interface from the point of view of the most important user group.</p>			
Keywords usability, user interface, user centered methods, software engineering			

## **Alkulause**

Valmiin diplomityön äärellä haluan kiittää työn valvojaa Marko Niemistä sekä ohjaajiani Mika Niemistä ja etenkin Sirpa Riihiahoa. Sirpa on käytännössä toiminut opinto-ohjaajanani vastaamalla kaikkiin käytettävyyttä koskeviin kysymyksiini. Iso kiitos kuuluu myös kaikille niille ystäväilleni, joilla olen työtäni luetuttanut. Varsinkin Vesa Suontaman ja Henri Seppäsen kommentit ovat auttaneet työtäni eteenpäin ja Katri Hirvensalon oikoluku paransi työni luettavuutta. Lisäksi haluan kiittää NRC:ä ja Petri Karppista tarjoamastaan mahdollisuudesta tehdä diplomityö.

Helsingissä 11.1.2005

Petteri.Suontama@iki.fi

# Sisältö

<b>1 Johdanto</b>	<b>1</b>
1.1 Diplomityön rakenne . . . . .	2
<b>2 Käsitteiden selvitys</b>	<b>3</b>
2.1 Asiantuntijajärjestelmä . . . . .	3
2.2 Käytettävyys . . . . .	3
2.3 Sovellus ja sen käyttöliittymä . . . . .	5
<b>3 Käyttäjakeskeisyys tuotekehityksessä</b>	<b>7</b>
3.1 Yleistä . . . . .	7
3.2 Käyttäjakeskeisiä standardeja . . . . .	8
3.2.1 ISO 13407 . . . . .	9
3.2.2 ISO 9241-11 . . . . .	10
3.3 Käyttäjakeskeisiä menetelmiä . . . . .	11
3.3.1 Käytettävyysarvio käyttäjien kanssa . . . . .	11
3.3.2 Asiantuntija-arviointi . . . . .	13
3.3.2.1 Kognitiivinen läpikäynti . . . . .	13
3.3.2.2 Heuristinen asiantuntija-arvio . . . . .	14
3.3.3 Haastattelu . . . . .	14

3.3.4	Tilannesidonnainen läpikäynti . . . . .	15
3.3.5	Osallistuva suunnittelu . . . . .	16
3.3.6	Prototyypit . . . . .	17
<b>4</b>	<b>Alkuperäisen järjestelmän esittely</b>	<b>18</b>
4.1	Yleistä . . . . .	18
4.2	Alkuperäinen käyttäjäryhmä . . . . .	19
4.3	Alkuperäinen toiminnallisuus . . . . .	20
4.4	Ongelmat alkuperäisessä järjestelmässä . . . . .	22
4.5	Alkuperäiset vaatimukset uudelle järjestelmälle . . . . .	23
4.6	Tavoiteltu uusi käyttäjäkunta . . . . .	24
<b>5</b>	<b>Toteutunut prosessi</b>	<b>26</b>
5.1	Ongelmakenttään tutustuminen . . . . .	26
5.1.1	Lähtötilanne . . . . .	27
5.1.2	Toteutus . . . . .	27
5.1.2.1	Ensimmäinen prototyyppi . . . . .	27
5.1.3	Kokemukset . . . . .	28
5.2	Toinen prototyyppi ja vaatimusten keräys . . . . .	28
5.2.1	Lähtötilanne . . . . .	31
5.2.2	Toteutus . . . . .	32
5.2.2.1	Käyttäjäkysely . . . . .	32
5.2.2.2	Prototyypin rakentaminen . . . . .	33
5.2.3	Yhteenveto . . . . .	34
5.3	Käytettävyydestä ja prototyypin parantelu . . . . .	37
5.3.1	Lähtötilanne . . . . .	37

<i>SISÄLTÖ</i>	iii
5.3.2 Käytettävyydestä . . . . .	38
5.3.3 Prototyypin parantelu . . . . .	38
5.3.4 Tilanne osion jälkeen . . . . .	41
5.4 Tuotteen julkaiseminen . . . . .	41
5.4.1 Käytettävyydestä . . . . .	41
5.4.2 Lopullinen tuote . . . . .	42
5.4.3 Yhteenveto . . . . .	43
<b>6 Johtopäätöksiä ja pohdintaa</b>	<b>47</b>
6.1 Eri käyttäjäryhmien huomioiminen . . . . .	47
6.2 Käytetyt menetelmät . . . . .	49
6.2.1 Prototyypit ja osallistuva suunnittelu . . . . .	49
6.2.2 Kyselyt . . . . .	49
6.2.3 Asiantuntija-arvio . . . . .	50
6.2.4 Käytettävyyden arviointi käyttäjien kanssa . . . . .	50
6.3 Aikataulu, resurssit ja laajuus . . . . .	50
6.4 Yhteenveto . . . . .	51
<b>A Ensimmäisen osion aikana lähetetty sähköpostikysely</b>	<b>55</b>

# Luku 1

## Johdanto

Tämän diplomityön aikana tutkitaan, kuinka pienelle asiantuntijajoukolle suunnattua ohjelmistoa voidaan laajentaa suuremmalle käyttäjäkunnalle sopivaksi. Työssä pohditaan, miten voidaan varmistaa alkuperäisen käyttäjäkunnan tyytyväisyys, kun ohjelmaa lähdetään muokkaamaan myös muiden kuin heidän tarpeita varten sopivaksi.

Eri käyttäjäryhmillä voi olla ristiriitaisia vaatimuksia tuotetta kohtaan. Alkuperäinen käyttäjäryhmä voi olettaa tuotteen olevan suunnattu vain heille ja näin ollen ajatella, että heidän näkökulmansa tuotteen käyttötavoista on oikea. Työn aikana tutkitaan, kuinka voidaan muodostaa toimivia kompromisseja eri käyttäjäryhmien tarpeiden välille niin, että sekä alkuperäiset että uudet käyttäjäryhmät ovat tyytyväisiä lopputulokseen. Mitä tekijöitä on otettava huomioon, jotta alkuperäisillä käyttäjillä ei heräisi muutosvastarintaa ohjelmaa kehitettäessä? Kuinka voidaan varmistaa, että uudet käyttäjäryhmät kokevat tuotteen omakseen, vaikka ohjelma olisi alun perin suunnattu toiselle käyttäjäryhmälle?

Tutkimuksen avuksi otettiin konkreettinen esimerkki matkapuhelinverkkoja simuloivasta ohjelmistosta. Tämä ohjelmisto oli alun perin suunnattu vain 19 hengelle. Ohjelman käyttö ja tulosten tulkitseminen vaati erityisasiantuntemusta matkapuhelinverkkojen toiminnasta. Ohjelman käyttötarpeita tutkittiin uusien ja alkuperäisten käyttäjien näkökulmasta. Löydettyjen käyttötarpeiden mukaan ohjelman käyttöliittymää kehitettiin niin, että se soveltui alkuperäistä laajemmalle käyttäjäkunnalle.

Kehitettävä ohjelma oli matkapuhelinverkkoja simuloiva järjestelmä. Sen avulla voitiin testata matkapuhelimiin tehtäviä sovelluksia todellisuutta vastaavissa verkko-olosuhteissa. Verkon



simuloinnilla voitiin testata tilanteita, jossa puhelimen verkkoyhteys ei toimi ideaalisesti.

Käyttöliittymän kehitykseen varattiin yhdeksän kuukautta aikaa. Tänä aikana rakennettiin uusi aikaisempaa helppokäyttöisempi käyttöliittymä, joka soveltui niin alkuperäiselle käyttäjäryhmälle kuin uusille käyttäjille.

Tehokkaan käyttöliittymän toteutus ei ole suoraviivaista. Käyttäjakeskeisillä tuotekehitysmenetelmillä pyritään käyttöliittymästä tekemään käyttäjien näkökulmasta parempi kuin perinteisillä ohjelmistokehityksen menetelmillä. Prosessit, jotka kuvaavat käyttöliittymän käyttäjakeskeistä toteutusta, ovat usein iteratiivisia. Käyttäjakeskeisessä toteutuksessa tuotteen ominaisuudet ja toiminnallisuudet pyritään saamaan todellisen loppukäyttäjän tarpeiden mukaisiksi. Tuotteen loppukäyttäjiltä saadut mielipiteet ohjaavat prosessin etenemistä.

Käyttöliittymän suunnittelijalle muodostuu väistämättä ohjelman käytöstä erilainen malli kuin sen käyttäjälle (Norman, 1988). Jotta tehty ohjelma vastaisi käyttäjiensä tarpeita, työssä käytettiin erilaisia käyttäjakeskeisiä ohjelmistokehityksen menetelmiä ja standardeja apuna.

Suurimmassa osassa käyttöliittymiä ja käytettävyyttä koskevia standardeja kuvataan yleisiä periaatteita ei niinkään konkreettisia menetelmiä, joilla hyvä käyttöliittymä saadaan aikaiseksi (Bevan, 2001). Tämän takia työssä esitellään ja tutkitaan joidenkin käyttäjakeskeisten ohjelmistotuotannon menetelmien toimivuutta käytännön työssä ja niiden soveltuvuutta tilanteeseen, jossa käyttäjäkuntaa halutaan laajentaa.

## 1.1 Diplomityön rakenne

Luvussa 2 selvennetään työssä käytettyjä termejä. Luvussa 3 tutustutaan oleviin ohjelmistoprosesseihin ja menetelmiin. Luvussa 4 kerrotaan, minkälainen kehitettävä simulaatio-ohjelmisto oli ennen siihen tehtyjä muutoksia. Luvussa 5 on kuvattu toteutunut ohjelmiston kehitysprosessi. Lopuksi luvussa 6 on pohdittu työn tuloksia, käytettyjä menetelmiä ja niistä on tehty johtopäätöksiä.

# Luku 2

## Käsitteiden selvitys

### 2.1 Asiantuntijajärjestelmä

Työn kohteena olevaa ohjelmistoa kuvataan sanalla asiantuntijajärjestelmä. Tällä tarkoitetaan, että ohjelman käyttäjillä on jotain erityistuntemusta eli asiantuntemusta kyseessä olevasta alasta. Asiantuntijajärjestelmässä voi olla termejä, jotka eivät ole yleisesti tunnettuja tämän asiantuntijaryhmän ulkopuolella. Mikäli ohjelmaa ei ole edes suunnattu tämän käyttäjäryhmän ulkopuolelle, voidaan ohjelmassa käyttää tämän asiantuntijaryhmän erityissanastoa.

Asiantuntijajärjestelmällä voidaan käsittää myös ohjelma, joka itsessään sisältää tietyn alan asiantuntemusta. Asiantuntijajärjestelmä pystyy tekemään jonkinlaisen tekoälynä perusteella johtopäätöksiä eli ikään kuin toimimaan alansa *asiantuntijana*.

Kehitettävä ohjelma voitiin luokitella asiantuntijajärjestelmäksi. Ohjelma suunnattiin verkko-asiantuntijoille tai ihmisille, joilla oli riittävä tietämys verkon toiminnasta. Toisaalta ohjelma tuotti graafisia ja numeraalisia yhteenvetoja, joihin oli sisällytetty verkkoalan asiantuntemusta.

### 2.2 Käytettävyys

Käytettävydestä on useita toisiaan täydentäviä tai eri näkökulmasta olevia määritelmiä. Kirjallisuudessa eniten viitattuja määritelmiä ovat ISO 9241-11 standardin määritelmä (ISO 9241-

11) ja Nielsenin määritelmä (Nielsen, 1993, s. 26).

**ISO 9241-11:**

Käytettävyys on mitta, joka kertoo, miten hyvin määrätyt *käyttäjät* voivat käyttää tuotetta määrättyssä käyttötilanteessa saavuttaakseen määritetyt tavoitteet *tuloksellisesti, tehokkaasti ja miellyttävästi*.

**Käyttäjä:** Henkilö, joka on vuorovaikutuksessa tuotteen kanssa.

**Tuloksellisuus:** Tarkkuus ja täydellisyys, jolla käyttäjät saavuttavat määritetyt tavoitteet.

**Tehokkuus:** Voimavarojen käyttö suhteessa tarkkuuteen ja täydellisyyteen käyttäjien saavuttaessa tavoitteet.

**Miellyttävyys:** Epämukavuuden puuttuminen ja myönteinen suhtautuminen tuotteen käyttöön.

**Nielsen:**

“... käytettävyys ei ole yksittäinen yksiulotteinen käyttöliittymän ominaisuus. Käytettävydessä on monia osia ja se perinteisesti liitetään näihin viiteen tekijään:

**Opittavuus:** Järjestelmän tulisi olla helppo oppia, jotta käyttäjä pystyy nopeasti aloittamaan työnteon järjestelmällä.

**Tehokkuus:** Järjestelmän tulisi olla tehokas käyttää, jotta käyttäjä voi opittuaan järjestelmän käytön saavuttaa korkean tuottavuuden tason.

**Muistettavuus:** Järjestelmän tulisi olla helppo muistaa, jotta satunnainen käyttäjä pystyisi ilman uudelleenopetteluja palaamaan käyttämään järjestelmää oltuaan käyttämättä sitä jonkin aikaa.

**Virheet:** Järjestelmässä tulisi olla alhainen virhetaso, jotta käyttäjät tekisivät vain vähän virheitä sitä käyttäessään, ja vaikka tekisivätkin, he selviävät virheistä helposti. Edelleen, katastrofaalisia virheitä ei saisi tapahtua.

**Tyytyväisyys:** Järjestelmän tulisi olla miellyttävä käyttää, jotta käyttäjät olisivat tyytyväisiä järjestelmän käyttöön ja pitäisivät siitä.”

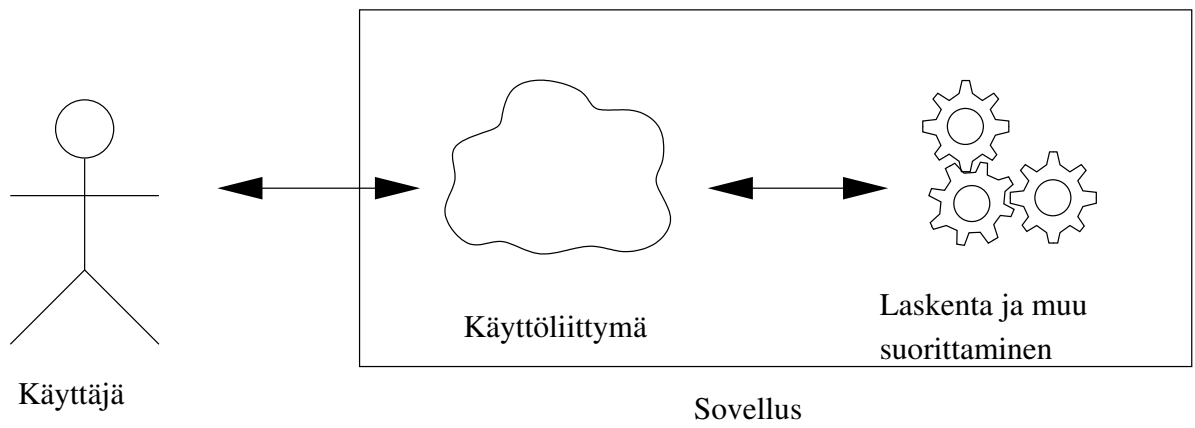
Hyvällä käytettävyydellä voidaan ennaltaehkäistä monia käyttäjän kohtaamia ongelmia. Käyttäjälle tulee tarjota vain sitä tietoa, jota hän kaipaa. Jos tietoa on tarjolla liikaa, ei käyttäjän huomio keskity olennaiseen vaan jää harhailemaan toistensa kanssa kilpailevien tietolähteiden kanssa.

Käytettävyys on monen tekijän summa. Kirjallisuudessa löytyy lähteitä, joissa on pohdittu käytettävyyden sisältöä ja merkitystä tätä työtä laajemmin (mm. Bevan, 2001). Tämän työn käyttöliittymäsuunnittelussa on pyritty saavuttamaan käytettävyydeltään hyvä lopputulos. Tällä tarkoitetaan järjestelmää, jossa on otettu huomioon käytettävyyden eri osa-alueet.

## 2.3 Sovellus ja sen käyttöliittymä

Tässä työssä sovelluksella tarkoitetaan tietokoneen suorittamaa ohjelmakoodia. Käyttäjä käyttää sovellusta saavuttaakseen tietyn tavoitteen. Käyttöliittymällä ymmärretään se ohjelman osuus, jonka käyttäjä näkee ja jonka kanssa hän on vuorovaikutuksessa. Käyttöliittymä tarjoaa käyttäjälle sovelluksen toiminnallisuudet ja näyttää sovelluksen suorituksen tulokset. Käyttöliittymällä pyritään saamaan monimutkainen tietokoneen ymmärtämä ohjelmistokokonaisuus käyttäjälle ymmärrettävään muotoon. Tietokoneohjelman käyttöliittymä on siis kaikkea, jonka käyttäjä näkee ohjelmasta käyttäessään sitä (ks. kuva 2.1). Käyttöliittymän kautta suoritettavalle ohjelmalle annetaan käsky toimia tietyllä tavalla ja tiedot suorituksen tuloksista annetaan takaisin käyttäjälle käyttöliittymän kautta. Käyttöliittymä kommunikoi sovelluksen laskentapuolelle, mitä sen kuuluu tehdä. Työssä olevassa simulaatiosovelluksessa laskenta koostuu useasta prosessista, joita yksi ja sama käyttöliittymä ohjaa.

Yleisiä käyttöliittymätyylejä ovat komentorivikäyttöliittymä ja graafinen käyttöliittymä. Komentorivikäyttöliittymän etu on, että sen käyttö on tehokasta. Komentorivillä on helpompi tehdä usean komennon koosteita eli skriptejä, jotka suorittavat usean käskyn nopeasti peräjälkeen (Miller, 2000). Esimerkkinä tällaisesta voidaan pitää Linuxin komentoriviä. Komentorivikäyttöliittymä on graafista käyttöliittymää vaikeampi oppia, sillä siinä on muistettava komentokäskyjä, joilla toiminnot suoritetaan. Graafinen käyttöliittymä on helpompi oppia ja siinä pystytään estämään virhetilanteita tarjoamalla käyttäjälle vain sellaisia toiminnallisuuksia, jotka on mahdollista tehdä. Windows on yleisesti tunnetuin käyttöjärjestelmän graafinen käyttöliittymä. Vaikka yksittäiset toimenpiteet ovat graafisella käyttöliittymällä helppoa tehdä, on sen automatisointi vaikeampaa kuin komentorivikäyttöliittymässä (Miller, 2000).



Kuva 2.1: Käyttäjä, käyttöliittymä ja sovellus

## Luku 3

# Käyttäjakeskeisyys tuotekehityksessä

Tässä luvussa kuvataan, erilaisia työn kannalta oleellisia tuotekehitysprosesseja ja -menetelmiä.

### 3.1 Yleistä

Kirjallisuudessa esitettyjen arvioiden mukaan jopa 60% ohjelmistoprojekteista ei saavuta niille asetettuja tavoitteita (Shneiderman, 1998, s. 104). Ohjelmistotuotannon kehitysmetodologiat ovat auttaneet kehittäjiä pysymään budjetissa ja aikataulussa. Ohjelmistotuotteen mittareina voidaan käyttää toiminnallisuutta, luotettavuutta, käytettävyyttä, tehokkuutta, ylläpidettävyyttä ja siirrettävyyttä (ISO 9126). Ohjelmistoprosesseissa määritetään halutut mittarit ja niille tavoitteet aina kuhunkin projektiin. Vain mitattavaa prosessia voi seurata, ohjata ja enustaa.

Kirjallisuudessa käytetään ohjelmistojen seurannan kuvaamiseen kolmijalkaa: toiminnallisuus, aika, resurssit (Atkinson, 1999). Näistä kolmesta vaatimuksesta kahden pystyy saavuttamaan ilman hyviä ohjelmistoprosesseja ja -käytäntöjä. Kaikkien kolmen saavuttaminen vaatii kuitenkin hyvää onnea, osaavia ihmisiä tai hyvän ohjelmistoprosessin.

Ohjelmistojen kehittämiseen on kehitetty suuri määrä erilaisia työkaluja, prosesseja ja menetelmiä. Yksimielisyyttä siitä, mitkä menetelmät olisivat kustannustehokkaita ei kuitenkaan ole. Yleisesti kuitenkin myönnetään, että käyttäjakeskeisillä tuotekehitysprosesseilla saadaan käyttöliittymästä paremmin käytettävä (Seffah et al, 2001).

## 3.2 Käyttäjakeskeisiä standardeja

Käyttäjakeskeiset standardit painottavat käyttäjän tuntemista. Käyttäjät ovat mukana koko kehitystyön aikana tavalla tai toisella, jolloin tuntuma käyttäjiin pysyy koko prosessin ajan. Joissakin prosessimalleissa käyttäjät otetaan mukaan suunnittelutyöhön. Toisissa malleissa käyttäjien havainnoinneilla, haastatteluilla ja testeillä haetaan käyttäjien näkökulma mukaan prosessiin.

Käyttäjakeskeisessä prosessissa tuotteen tavoiteltava käytettävyys määritellään mitattavasti. Näin voidaan todistaa, että tuote saavuttaa sille ennalta asetetut käytettävyysvaatimukset (Faulkner, 2000). Käyttäjakeskeisiin prosesseihin on koottu hyväksi havaittuja käytäntöjä ja menetelmiä. Käyttäjakeskeisillä prosesseilla pyritään saavuttamaan seuraavia tavoitteita verrattuna perinteisiin ohjelmistoprosesseihin:

- Tuotteesta löydetään käytettävyysongelmat jo prosessin alkuvaiheessa, jolloin ne on halvempaa korjata.
- Tuotteen käytön koulutukseen ei mene niin paljoa aikaa, koska tuote on helpompi ymmärtää ja käyttää.
- Tuotteen käyttö on tehokkaampaa.
- Käyttäjien muutosvastarinta uutta ohjelmistoa vastaan on pienempi.

Käyttäjakeskeiset prosessit ovat usein iteratiivisia. Tekemällä lyhyitä iteraatioita ja mittaamalla saavutettua käytettävyyttä löydetään käytettävyysongelmat jo prosessin alkuvaiheessa. Jos isot käyttöliittymämuutokset jäävät prosessin loppuvaiheeseen, ne ovat kalliimpia toteuttaa.

Kun ohjelmisto toimii käyttäjien olettamalla tavalla, ei tuotteen koulutukseen tarvitse varata yhtä paljon resursseja. Intuiitiivinen ja tehokas käyttöliittymä ovat käyttäjakeskeisen prosessin tavoitteita. Vaikka nämä tavoitteet voidaan luokitella laadullisiksi, ne vähentävät koko prosessin vaatimaa aikaa ja hintaa.

Ohjelmiston käyttö tehostuu, kun se on suunniteltu käyttäjakeskeisesti. Ohjelmiston suunnittelussa on otettu huomioon loppukäyttäjien tarpeet ja toimintatavat, jolloin ohjelma tukee käyttäjää mahdollisimman hyvin.

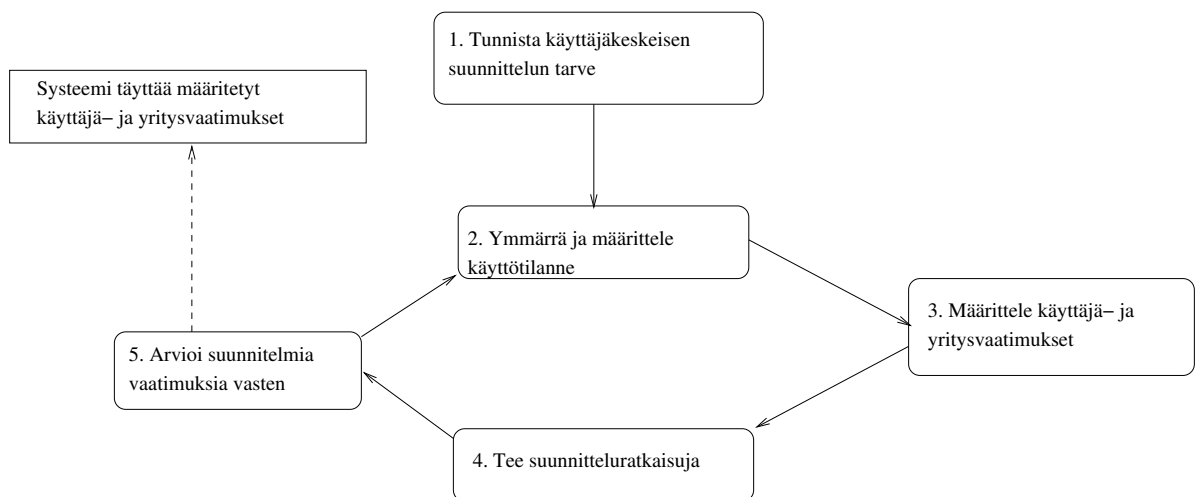
Kun käyttäjillä on tunne, että he ovat voineet vaikuttaa käyttämäänsä ohjelmistoon, on muutostarinta ohjelmistoa vastaan pienempi. He myös ymmärtävät, mikä suunnittelun taustalla on ollut vaikuttamassa tehtyihin suunnitteluratkaisuihin.

Käytettävyyteen liittyvissä standardeissa on kuvattu mahdollisia prosessimalleja. Käytettävyyteen liittyviä standardeja ovat muun muassa:

- ISO 13407 - käyttäjäkeskeiset prosessit interaktiivisille järjestelmille
- ISO DTR 16982 - käyttäjäkeskeistä suunnittelua tukevat käyttäjäkeskeisen menetit
- ISO 9241-11 Näyttöpäätteillä tehtävän toimistotyön ergonomiset vaatimukset. Osa 11: Käytettävyyden määrittely ja arviointi

### 3.2.1 ISO 13407

ISO 13407:n (ISO 13407) prosessimalli on standardien tarjoamista käyttäjäkeskeisistä prosessimalleista tunnetuin. Se tarjoaa neuvoja käyttäjäkeskeisten toimintojen tekemiseen koko vuorovaikutteisen ja tietokonepohjaisen tuotteen elinkaaren aikana. Standardi kuvaa tuotekehityksen iteratiiviseksi kuvan 3.1 mukaisesti.



Kuva 3.1: ISO 13407 -prosessikaavio



Käyttäjakeskeinen prosessi alkaa heti prosessin alusta, kun käyttäjien tarve uudelle ohjelmistolle on löydetty. Prosessissa ei edetä vesiputouksmallisesti aina vaiheesta seuraavaan, vaan suunnittelua jatketaan iteratiivisesti, kunnes ohjelmisto saavuttaa sille asetetut vaatimukset. Jokaisessa iteraatiossa tarkennetaan ymmärrystä käyttötilanteesta, tarkennetaan käyttäjien ja yrityksen vaatimuksia lopputuotteelle, tehdään suunnitteluratkaisuja ja arvioidaan tehtyjä suunnitelmia ja suunnitteluratkaisuja.

Ympäristöstä tulisi tunnistaa oletetut käyttäjät, käyttäjien tekemä työ ja fyysinen ja sosiaalinen käyttöympäristö. Kun tuotteen käyttötilanne ja -ympäristö tunnetaan, se ohjaa jo alusta alkaen suunnittelupäätöksiä. Mikäli ollaan laajentamassa jo olemassa olevaa tuotetta, tulee alkuperäiset määritelmät tarkistaa.

Monissa ohjelmistokehitysprosesseissa painotetaan toiminnallisten vaatimusten keräämistä. ISO 13407 laajentaa vaatimusten keräämistä myös käyttäjän ja yrityksen näkökulmaan. Vaatimukset tulisi myös kirjata selkeästi, jotta niitä voidaan myöhemmin mitata ja arvioida.

Suunnitteluratkaisujen tekemisessä tulee käyttää jo olemassa olevaa tietoa ja taitoa tuotteen kehittämiseksi. Mahdollisesti laajoista ja monipuolisista vaatimuksista tulee tehdä konkreettisia simulaatioita, malleja tai prototyyppejä ja kerätä näiden avulla edelleen lisätietoa käyttäjien tarpeista kunnes saavutetaan ratkaisu, joka saavuttaa asetetut tavoitteet. Eri tasoisia malleja voidaan käyttää koko prosessin ajan. Alussa mallit voivat olla hyvinkin suurpiirteisiä. Yksinkertaiset, nopeasti tehdyt mallit helpottavat keskustelua ja erilaisten ratkaisuvaihtoehtojen läpikäyntiä.

Arvioiminen on olennainen osa ISO 13407:n kuvaamaa prosessia. Arviointia tulisi tapahtua tuotekehityksen kaikissa vaiheissa, jotta saataisiin suunnitteluun vaikuttavaa palautetta ja tiedettäisiin, ollaanko asetetut tavoitteet saavutettu. Arviointia tulisi jatkaa vielä valmiin tuotteen seurantaankin, jotta tiedettäisiin tuotteen pidemmän aikavälin tavoitteiden toteutuminen.

### **3.2.2 ISO 9241-11**

ISO 9241-11 (ISO 9241-11) selittää käytettävyyden mittaamisen hyödyt käyttäjän suoriutumisen ja tyytyväisyyden kannalta. Nämä mitataan sillä, miten hyvin halutut tavoitteet saavutetaan, miten paljon työtä tarvitaan tavoitteiden saavuttamiseksi ja miten mukavaksi käyttäjä kokee tuotteen käytön. Standardi ei käsittele järjestelmän kehittämisprosessia. Sen sijaan se

- määrittelee käytettävyyden, jolloin eri käytettävyyden osa-alueet osataan tunnistaa ja ottaa huomioon
- antaa keinoja käytettävyyden ja sen kehityksen mittaamiseen
- mahdollistaa että tuloksien avulla voidaan saavutettu käytettävyys dokumentoida ja todentaa.

Käytettävyyden mittareina käytetään tuloksellisuutta, tehokkuutta ja tyytyväisyyttä tietyssä käyttötilanteessa ja tavoitteella. Tuotteen käyttötilanteeseen kuuluvat käyttäjä, tehtävä, laitteisto ja fyysinen ja sosiaalinen käyttöympäristö.

### **3.3 Käyttäjakeskeisiä menetelmiä**

Käyttäjakeskeiset prosessit koostuvat erilaisista menetelmistä. Tässä kappaleessa tutustutaan tämän työn aikana läpikäytyihin menetelmiin. Käytettävyyden menetelmiä on olemassa paljon. Ongelmana on, ettei ole tiedossa, minkälaisessa tilanteessa olisi järkevää käyttää mitään menetelmää (Riihiaho, 2000).

Luvussa 6 on pohdittu tässä työssä käytettyjen menetelmien soveltuvuutta, kun olemassa olevaa järjestelmää laajennetaan uusille käyttäjäryhmille.

#### **3.3.1 Käytettävyysarvio käyttäjien kanssa**

Tuotteelle tehtävää käytettävyysarviota käyttäjän kanssa nimitetään käytettävyystestiksi. Siinä tuotteen käytettävyyttä tulee testaamaan tuotteen todellisen käyttäjäryhmän edustajia. Testi suoritetaan usein käytettävyyslaboratoriossa. Laboratoriosta pyritään poistamaan ylimääräiset häiriötekijät ja sinne järjestetään testin edellyttämät olosuhteet. Testiä seuraa usein vähintään kaksi henkilöä: testin ohjaaja ja kirjuri. Testin ohjaaja on käyttäjän vieressä antamassa käyttäjälle testitehtäviä ja huolehtimassa, että testi sujuu suunnitelmien mukaisesti. Kirjuri kirjaa testin kulun. Usein testitilanne myös videokuvataan myöhempää läpikäyntiä varten. Usein testilaboratoriossa on puoliläpäisevä peili. Vain testin ohjaaja on käyttäjän kanssa samassa tilassa. Näin käyttäjä ei häiriinny ylimääräisistä henkilöistä tai kirjurin toimista. Käyttäjää

pyydetään usein ajattelemaan ääneen eli kertomaan ajatuksensa ääneen, jotta testitilanteessa tiedettäisiin, mitä käyttäjä milloinkin miettii.

Testiin osallistuu tuotteen oikean käyttäjäryhmän edustajia. Testiin kutsuttujen käyttäjien määrä riippuu testin tavoitteista. Mikäli tavoitteena on käytettävyysongelmien löytäminen, viisi käyttäjää on osoittautunut parhaaksi aika-hyöty -suhteeltaan (Riihiaho, 2000). Tilastollisesti merkittävien tuloksien tekemiseen tarvitaan kuitenkin tätä useampia käyttäjiä.

Perinteisessä käytettävyydestissä testitehtävät ja koko testin kulku on ennalta suunniteltu. Testitilanne pyritään vakioimaan niin, että eri testit olisivat keskenään vertailukelpoisia ja tulokset tilastollisesti päteviä. Tällöin puhutaan niin sanotusta summatiivisesta testauksesta. Testissä varmistetaan tiettyjen ohjelman osa-alueiden käytettävyys. Formatiivisessa testauksessa ideoidaan uusia toiminnallisuuksia ja sitä, miten toiminnallisuudet olisi hyvä toteuttaa. Tässä voidaan käyttää vapaata läpikäyntiä testausmenetelmänä (Nieminen, 1996). Siinä testikäyttäjälle ei anneta niin suoraan testitehtäviä, vaan häntä pyydetään yleisesti käyttämään järjestelmää. Ennalta on suunniteltu, mitkä järjestelmän osa-alueet halutaan käydä läpi ja mistä osioista halutaan kommentteja. Käyttäjä voi itse käydä järjestelmää siinä järjestyksessä läpi, jonka hän parhaaksi näkee. Usein on tarkoituksenmukaista käyttää testiä, joka on näiden kahden ääripään välillä (Shneiderman, 1998, s. 128).

Testien jälkeen löydetty käytettävyysongelmat luokitellaan sen mukaan, kuinka merkittävästi ne vaikeuttavat tuotteen käyttöä. Tulosten analysointiin vaikuttaa, miten testit on suunniteltu. Vakavimmat käytettävyysongelmat löytyvät ilman videointia (Nielsen, 1993, s. 203). Videointi on osoittautunut kuitenkin hyväksi tavaksi, jos kaikki testeissä ilmenneet ongelmat halutaan kuvata tarkasti. Videon avulla pystytään myös todistamaan virheiden olemassaolo, mikäli kehitystyössä on mukana tahoja, joille tämä halutaan vakuuttaa.

Mikäli tavoitteena on virheiden löytäminen niiden korjaamista varten, voidaan käyttää kevennettyjä käytettävyyden metodeja kuten käytettävyydesti *pikadata-analysoinnilla*<sup>1</sup>. Tutkimuksen mukaan menetelmällä löydetään lähes yhtä paljon käytettävyyso ongelmia kuin videodata-analysoinnilla, mutta siihen tarvitaan vain 10% videoanalysointiin tarvittavasta ajasta (Kjeldskov et al, 2004).

Testiin valitut käyttäjät kommentoivat tuotetta omasta näkökulmastaan. Tämän takia jo testiä suunniteltaessa ja tuloksia analysoidessa tulee ottaa huomioon, mihin käyttäjäryhmään kuulu-

---

<sup>1</sup>*Pikadata-analysointi*: Instant Data Analysis

via ihmisiä testiin kutsuu. Käytettävyydesteistä saadaan käytettävyyso Ongelmien lisäksi tietoa siitä, kuinka kyseisen käyttäjäryhmän edustajat toimivat tuotteen kanssa. Testin jälkeen suoritettavassa haastattelussa voidaan vielä syventää tietämystä kyseessä olevasta käyttäjäryhmästä, sen toiveista sekä toimintatavoista.

### 3.3.2 Asiantuntija-arviointi

Asiantuntija-arvioinnissa järjestelmää arvioidaan ilman tuotteen käyttäjiä. Asiantuntija-arviointia voidaan käyttää tuotekehityksen kaikissa vaiheissa. Se on nopea tapa selvittää järjestelmässä olevia ongelmakohtia. Asiantuntija-arvioinnit voidaan jakaa kahteen osaan, kognitiiviseen läpikäyntiin ja heuristiseen asiantuntija-arvioon.

#### 3.3.2.1 Kognitiivinen läpikäynti

Kognitiivisessa läpikäynnissä järjestelmää käydään läpi, niin kuin järjestelmää tuntematon ihminen sen näkee. Siinä käytetään apuna seuraavia kysymyksiä (Riihiaho, 2000):

- Onko tavoite oikea, ymmärtääkö käyttäjä kyseisen vaiheen kuuluvan tehtävään?
- Huomaako käyttäjä toiminnon ja onko se helposti löydettävissä?
- Yhdistääkö käyttäjä toiminnon omaan tavoitteeseensa?
- Huomaako käyttäjä toiminnon tehtyään, että hän on edennyt tavoitteessaan oikeaan suuntaan?

Kognitiivinen läpikäynti soveltuu parhaiten joidenkin melko pienien järjestelmän osakokonaisuuksien läpikäyntiin. Siinä valittu tehtäväkokonaisuus jaetaan pieniksi osatehtäviksi. Jokaisen osatehtävän kanssa tarkastetaan yllä olevat kysymykset, ja etsitään mahdolliset ongelmat. Menetelmä soveltuu hyvin, kun halutaan selvittää järjestelmän opittavuutta.

### 3.3.2 Heuristinen asiantuntija-arvio

Heuristinen asiantuntija-arvio on yksi käytetyimpiä testausmenetelmiä, sillä se on helppo oppia, sen tekeminen on halpaa eikä se vaadi suuria etukäteisvalmisteluja (Riihiaho, 2000). Heuristisessa asiantuntija-arviossa järjestelmää käydään läpi ennalta laadittujen sääntöjen eli heuristiikkojen avulla. Järjestelmästä pyritään löytämään kohtia, jotka rikkovat näitä heuristiikkoja. Heuristisen asiantuntija-arvion voi tehdä niin tuotteen varhaisimmille paperiprototyypeille kuin valmiille tuotteelle.

Eri tuotealueille on kehitetty erilaisia heuristiikkoja eli nyrkkisääntöjä. Kirjallisuudessa viitataan usein Nielsenin (1993) kymmeneen heuristiikkaan. Ohjelmistotaloilla on usein omia heuristiikkojaan omien ohjelmien rakentamiseen. Esimerkiksi Microsoft on tehnyt Windows -ohjelmille omat varsin yksityiskohtaiset heuristiikkansa (Microsoft, 1995). Nielsenin heuristiikka *“minimoi käyttäjän muistikuorma”* ja Microsoftin tarkka heuristiikka *“kaikilla ikkunoilla on hyvä otsikko ja sopiva kuvake vasemmassa yläkulmassa”* ovat esimerkkejä eri tyyllisistä heuristiikoista.

### 3.3.3 Haastattelu

Haastattelutyyppejä on useita strukturoidusta haastattelusta epämuodollisiin keskusteluihin. Strukturoidussa haastattelussa haastattelijalla on ennalta mietityt kysymykset ja niihin vastaukset, joista haastateltava valitsee sopivan. Strukturoimattomassa haastattelussa kysymykset ovat avoimia ja niihin kysytään haastateltavan mielipidettä. Kun tehtävälä on vielä epäselvä, strukturoimaton haastattelu on yleensä soveltuvampi tapa. On tärkeää, että haastattelijalla on valmiiksi mietittyjä kysymyksiä. Joskus haastateltava kertoo oma-aloitteisesti varsin kattavasti haastateltavasta aiheesta. Jos haastateltava on vähäsanainen tai keskustelua ei muuten saada sujumaan luontevasti on hyvä antaa muutamia helpompia kysymyksiä, jolloin jännittynyt tunnelma saattaa helpottaa. (Faulkner, 2000)

Haastatteluksi voidaan katsoa myös epämuodollinen keskustelu, jota tapahtuu työpaikalla muun vapaan keskustelun lomassa. Kahvitauolla tai työpaikan käytävällä tapahtuneista keskusteluista saa usein hyviä ideoita tuotteen kehittämiseen. Keskustelujen etuna on se, ettei niihin liity haastattelutilanteessa joskus ilmenevää jännitystä. Tällaiset sattumanvaraiset keskustelut, joita voi tuki ohjata haastattelun tyyliin, tuottavat usein hyviä ideoita.

### 3.3.4 Tilannesidonnainen läpikäynti<sup>2</sup>

Tilannesidonnainen läpikäynti (Beyer & Holtzblatt, 1998) on menetelmä, jossa seurataan asiakkaan työtä ja tehdään hänelle työtä koskevia kysymyksiä. Näin saadaan parempaa tietoa asiakkaan työskentelytavoista. Tilannesidonnaisen läpikäynnin tarkoituksena on kerätä tietoa, ei niinkään selittää ilmiöitä (Hom, 2004). Tavoitteena on, että koska haastattelija menee haastateltavan luokse, haastateltava pystyy työskentelemään lähes normaalisti. Näin haastattelija pääsee tutustumaan tehtävään työhön sen todellisessa ympäristössä. Tilannesidonnainen läpikäynti soveltuu sen tietoa keräävän muotonsa takia parhaiten tuotekehitysprosessin alkuun.

Tilannesidonnainen läpikäynti perustuu seuraavaan kolmeen periaatteeseen (Usor, 2004):

- Haastattelu tehdään työn oikeassa kontekstissa
- Haastateltava on oman työnsä asiantuntija, joten hänen mielipiteensä on hyvä ottaa mukaan tuotekehitykseen.
- Haastattelijan tavoite vaikuttaa saataviin tuloksiin, joten tavoitteen tulee olla selkeä.

Tilannesidonnaisessa läpikäynnissä haastateltava voi tehdä omaa työtänsä lähes häiriöttä, jolloin haastatteluun ei kulu haastateltavan näkökulmasta paljoa aikaa. Tämä helpottaa haastateltavien löytämistä. Tilannesidonnaisen läpikäynnin toteutus voidaan jakaa neljään osioon (UsabilityNet, 2004):

- lyhyt haastatteluosio
- vaihto haastattelusta havainnointiin
- havainnointi
- yhteenveto

Haastatteluosiossa käydään tavallisen haastattelun tyyliin muutama kysymys läpi ja selvitetään haastateltavalle tilannesidonnaisen läpikäynnin periaatteet. Mikäli nauhoitusta aiotaan käyttää, tulee käyttäjältä pyytää siitä lupa tässä vaiheessa.

---

<sup>2</sup>Suomenkielisissäkin teksteissä näkee usein tilannesidonnaisesta läpikäynnistä käytettävän nimeä *contextual inquiry*

Haastattelun vaihtuminen havainnoinniksi tulee tehdä selkeästi, jotta käyttäjä ei enää odota kysymyksiä, vaan keskittyy omaan työhönsä. Haastattelijan tulee ikään kuin häipyä taustalle tässä vaiheessa.

Havainnoinnin aikana haastattelija voi tehdä tarkentavia kysymyksiä haastateltavalle, kunhan se ei keskeytä mitään tärkeitä ja yhtenäistä työvaihetta.

Yhteenvedossa haastattelija kertoo haastattelemalleen henkilölle, mitä hän on oppinut havainnoinnin aikana. Näin varmistutaan, että haastattelija on ymmärtänyt oikein näkemänsä ja haastateltava voi vielä kommentoida ja oikaista haastattelijan näkemyksiä tapahtuneesta.

Kun haastattelija lähtee tekemään tilannesidonnaista läpikäyntiä, hänellä on oltava selvillä selkeä tavoite (Hom, 2004). Tavoite vaikuttaa huomattavasti siihen, mihin havainnoinnin aikana kiinnitetään huomiota. Kaikkea ei voi havainnoida samanaikaisesti.

Tilannesidonnaisessa läpikäynnissä on samoja piirteitä kuin etnografisessa tutkimuksessa, jossa tutkija menee osaksi jotakin yhteisöä. Tilannesidonnaisessa läpikäynnissä haastattelija voi kysellä tarvittaessa aktiivisesti havainnoinnin aikana. Tämä nopeuttaa tuloksien keräämistä etnografiseen tutkimukseen verrattuna.

### **3.3.5 Osallistuva suunnittelu**

Osallistuvan suunnittelun (Winograd, 1996; InfoDesign 2004) periaatteena on, että tuotteen rakentaja kehittää tuotetta yhdessä käyttäjien ja tilaajan kanssa. Se antaa tuotteen loppukäyttäjille mahdollisuuden keskustella tuotteen käyttötärpeestä ja siten kasvattaa mahdollisuutta, että tuotteesta tulee heille mieluinen. Osallistuvassa suunnittelussa tietotaidoltaan ja taustoiltaan erilaiset ihmiset voivat keskustella tasavertaisesti. Osallistuva suunnittelu antaa tuotteen kehittäjille mahdollisuuden tavata ja ymmärtää käyttäjiä sekä työskennellä heidän kanssaan.

Tarve osallistuvaan suunnitteluun tulee, kun tuotteen kehittäjillä ei ole riittävää tietoa-taitoa kehitettävän tuotteen aihealueesta. Suunnitteluun otetaan käyttäjiä mukaan, sillä he ovat aihealueen asiantuntijoita. Näin saadaan yhdessä parempi näkemys tuotteelle asetettaviin vaatimuksiin. Käyttäjillä ei ole useinkaan riittävää tietoa siitä, mitkä tekniset ratkaisut ovat mahdollisia, joten kehittäjien ja käyttäjien yhteistyö auttaa kumpaakin osapuolta.

### 3.3.6 Prototyypit

Prototyyppien rakentamisen on todettu vähentävän tarvetta muuttaa tuotetta tuotekehityksen myöhemmissä vaiheissa (ISO 13407). Prototyypit helpottavat kommunikointia, kun tuotteen ominaisuuksia voi konkreettisesti kokeilla ja nähdä. Niiden avulla voidaan nopeasti testata erilaisia ratkaisuvaihtoehtoja. Prototyyppien tarkkuus voi vaihdella prosessin edetessä. Alussa prototyypit voivat olla kynällä ja paperilla tehtyjä nopeita hahmotelmia.

Prototyyppien ei tarvitse aina olla pelkkiä pois heitettäviä keskustelun apuvälineitä, vaan niitä voidaan laajentaa pikkuhiljaa aina kohti lopullista tuotetta. Ensimmäiset yksinkertaiset prototyypit, joita ei ole toteutettu suurella tarkkuudella soveltuvat hyvin ajattelun, informaation ja käyttäjätarpeiden tutkimiseen. Yksityiskohtaiset prototyypit soveltuvat paremmin käyttäjän fyysisten tarpeiden tutkimiseen kuten näkymän intuitiivisuuden varmistamiseen (Hall, 2001). Erityisen hyödylliseksi prototyyppien rakentaminen on osoittautunut käyttöliittymien määrittelyn yhteydessä (Haikala & Märijärvi, 2001, s. 33).



## Luku 4

# Alkuperäisen järjestelmän esittely

Tässä luvussa käydään läpi, minkälainen simulaatio-ohjelma oli ennen kuin siihen ryhdyttiin tekemään parannuksia käyttäjäkeskeisen suunnittelun periaatteiden mukaisesti.

### 4.1 Yleistä

Matkapuhelinverkot ovat monimutkaisia järjestelmiä. Tukiasemissa on paljon säädettäviä ominaisuuksia ja verkko saattaa sijaita hyvinkin erilaisissa ympäristöissä. Nopeat tietokoneet ovat mahdollistaneet verkkojen mallintamisen tietokoneella. Mallit ovat raskaista matemaattisia mallinnuksia todellisista verkoista. Näitä malleja kehitetään koko ajan vastaamaan paremmin todellisia verkkoja. Verkkojen ominaisuudet kuitenkin kehittyvät, joten matemaattisten mallien on tämänkin takia kehityttävä. Laskentapuolen kehitys on itsessään yksi iso, koko ajan käynnissä oleva prosessi, mutta tässä työssä keskitytään simulaation käyttöliittymän kehittämiseen.

Verkossa toimii erilaisia päätelaitteita kuten matkapuhelimia. Matkapuhelimet voivat käyttää erilaisia verkon palveluja. Tällaisia palveluja ovat esimerkiksi tavallinen puhelu, GPRS-datan<sup>1</sup> siirto ja SIP-protokollaa<sup>2</sup> käyttävien ohjelmien käyttö. Kun verkon simulointi saatiin tehtyä, haluttiin sitä käyttää myös verkossa toimivien palveluiden testaamiseen. Simulaatiolla

---

<sup>1</sup>*GPRS*: General Packet Radio Service mahdollistaa matkapuhelimen nopeat datayhteydet.

<sup>2</sup>*SIP*: Session Initiation Protocol on matkapuhelimissa käytettävä kommunikaatioprotokolla, jonka avulla voidaan matkapuhelimissa toteuttaa verkon yli toimivia sovelluksia.

voidaan varmistaa, että matkapuhelimeen tehty sovellus toimii halutulla tavalla poikkeuksellisissakin verkko-olosuhteissa. Matkapuhelimeen tehdyn sovelluksen toimintaa voidaan näin testata juuri haluamassaan verkkoympäristössä.

Simulaattori laskee tehtyjen matemaattisten mallinnusten mukaisesti radiosignaalin etenemistä ja toteuttaa päätelaitteen oikeaa tilannetta vastaavan käyttäytymisen. Käyttäjälle annetaan hänen tarvitsemaansa tietoa muun muassa matkapuhelimen signaalin laadusta.

Simulaattorin laskentamalleja päivitetään, koska todellinen verkkoympäristö muuttuu koko ajan. Simulaattorin on kyettävä mukautumaan ja ennakoimaan näitä muutoksia. Simulaattorin käyttöliittymän halutaan pysyvän samannäköisenä, vaikka siinä käytetyt laskentamallit muuttuisivat. Matkapuhelinverkkojen toiminnassa on olemassa yleiskäsitteitä, jotka eivät muutu verkon ominaisuuksien mukana. Tällaisia muuttumattomia yleiskäsitteitä ovat esimerkiksi verkon kuormitus ja päätelaitteen liikkumisnopeus. Näyttämällä ja kysymällä simulaation käynnistäjältä näitä muuttumattomia asetuksia voidaan simulaatio käynnistää samanlaisella ohjelmalla vaikka itse verkon toiminnallisuus muuttuisikin.

## 4.2 Alkuperäinen käyttäjäryhmä

Alkuperäinen käyttäjäryhmä koostui käyttäjäkirjauksen mukaan yhdeksästätoista ihmisestä eri puolilta maailmaa. He olivat matkapuhelinjärjestelmien asiantuntijoita ja halusivat testata kehitettyjä asiakas-palvelin<sup>3</sup>-sovelluksia ja erilaisia puhelinohjelmistoja oikeaa verkkoa vastaavissa olosuhteissa. Käyttäjäryhmä koostui pelkästään talon sisäisistä käyttäjistä, koska ohjelman lisenssin takia sitä ei saanut levittää suuremmalle yleisölle. Käyttäjät säätivät verkon ominaisuuksia ja toimintaa samalla kun he seurasivat matkapuhelimessa olevan sovelluksen toimintaa.

Matkapuhelinverkkoalalla jo pitkään toimineena käyttäjille oli tullut tutuksi suuri määrä alaan liittyvää erikoistermistöä. He olivat asentaneet kyseessä olevan sovelluksen omatoimisesti helpottamaan simulaatiosysteemin käynnistämistä.

---

<sup>3</sup>*asiakas-palvelin eli client-server*: järjestelmä, jossa sovellus eli asiakas pyytää toista ohjelmaa eli palvelinta tekemään halutut toimenpiteet asiakkaalle

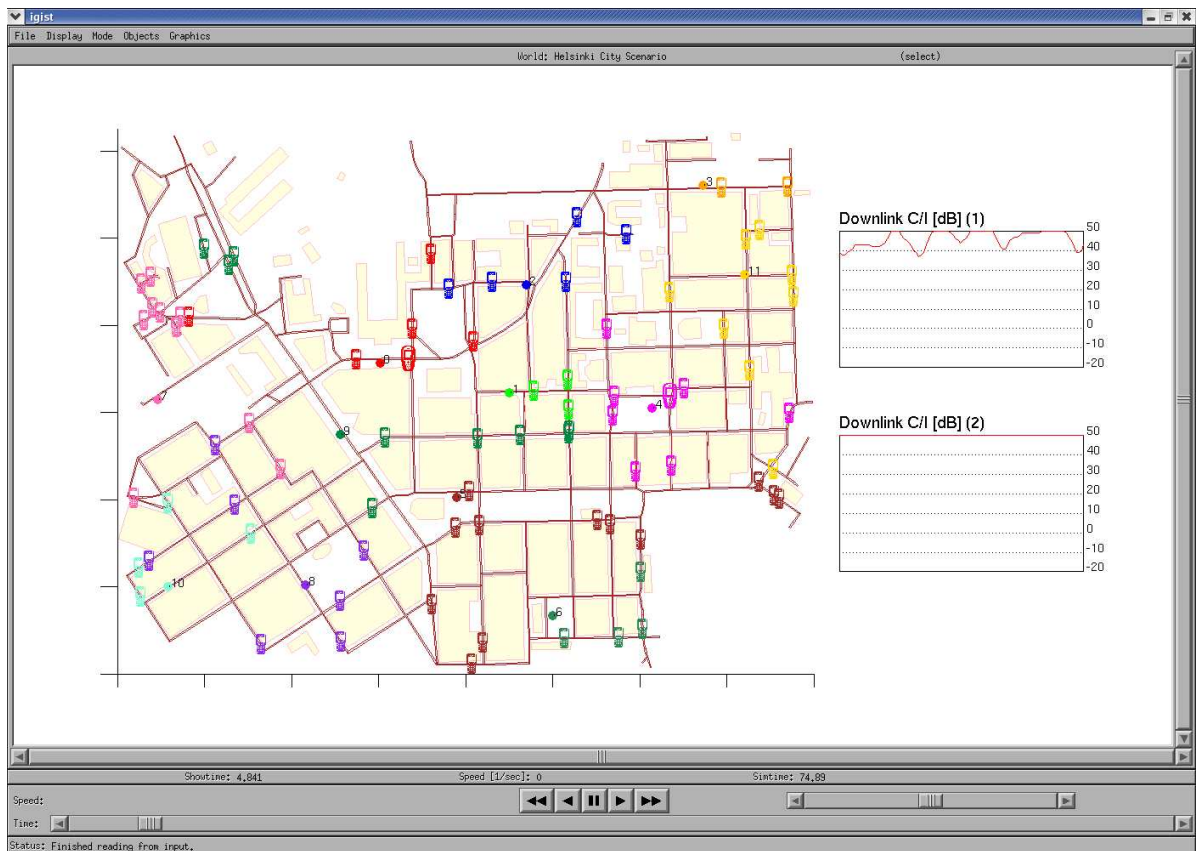
### 4.3 Alkuperäinen toiminnallisuus

Käyttöliittymällä käynnistettiin haluttu matkapuhelinverkon simulaatio. Tähän simulaatioon käynnistettiin myös testattava matkapuhelimen sovellus ja seurattiin sovelluksen toimintaa. Kun todellista verkkoympäristöä mallinnettiin maantieteellisesti pieneltä alueelta täytyi simulaatioon syöttää sen käynnistyessä noin 17 000 parametria. Simuloinnin käynnistys pyrittiin pitämään yksinkertaisena, joten valtaosa parametreista luettiin tiedostosta.

Simulaation pystyi käynnistämään tiedostoon kirjoitettujen oletusasetusten kanssa. Käyttäjän annettiin muuttaa muutamaa oleellisinta parametria. Nämä parametrit oli valittu sen mukaan, mitä alkuperäiset käyttäjät olivat pyytäneet. Parametreja oli alunperin vain yhdeksän kuten kuvasta 4.1 näkyy. Muihin parametreihin käyttäjä ei päässyt käsiksi käyttöliittymän avulla. Muutokset täytyi tehdä suoraan tiedostoissa oleviin oletusparametreihin. Kun simulaatio oli käynnistetty, voitiin verkko-olosuhteiden kehitystä ja sovelluksen toimintaa seurata kuvan 4.2 mukaisesta karttanäkymästä.



Kuva 4.1: Alkuperäinen simulaation käynnistyksen käyttöliittymä



Kuva 4.2: Alkuperäinen simulaation karttanäkymä, josta voi seurata simulaation etenemistä

## 4.4 Ongelmat alkuperäisessä järjestelmässä

Alkuperäinen käyttöliittymä oli toteutettu, kun simulaation käynnistämisen yhteydessä haluttiin tehdä muutoksia simulaation parametreihin. Käynnistämisen yhteydessä kysyttäviksi parametreiksi oltiin otettu ne, joita useimmiten haluttiin muuttaa ja joiden tekeminen onnistui helposti. Kun simulaation käynnistämiseen oli liitetty käyttöliittymä, huomattiin, kuinka paljon se helpottaa simulaation käyttöä. Käyttöliittymään haluttiin lisätä uusia parametreja. Sitä ei kuitenkaan oltu suunniteltu tulevia muutoksia varten, jolloin ohjelman muokkaaminen oli hankalaa. Alkuperäinen käyttöliittymä kuitenkin todisti sen, että matkapuhelimen sovellusten testaaminen simuloidussa verkkoympäristössä on tarpeellista.

Simulaation tuloksia näyttäväksi karttaikkunaksi valittu ohjelmistokirjasto ei antanut mah-

dollisuuksia kaikkien simulaation tulosten näyttämiseen. Ainoa tulosten esitystapa oli viiva-diagrammit kartan ohessa. Kartan ohjelmakirjaston lisenssi ei myöskään sallinut simulaation myymistä suuremmalle käyttäjäryhmälle.

Karttakirjastossa olevan ongelman takia ohjelman muistin tarve kasvoi simulaation pyöries-sä. Pitkässä simulaatiossa tämä muodostui ongelmaksi sillä muistin tarve rajoitti simulaation käyttöajan muutamaan tuntiin. Todelliseksi ongelmaksi tämä muodostuu senkin takia, että karttakirjastoa ei sen lisenssin takia voitu muuttaa, vaan sitä oli käytettävä sellaisenaan.

Karttanäkymä sai näytettävän tilan suoraan simulaation laskentaprosesseilta. Tiedon välittä-miseen tehtyä viestinvaihtoa ei oltu optimoitu ja alkuperäinen ratkaisu kulutti simulaatiota tekevän koneen tehoja huomattavan paljon, keskimäärin noin 50Mb minuutissa.

## 4.5 Alkuperäiset vaatimukset uudelle järjestelmälle

Alkuperäinen ohjelma oli todettu monelta osin puutteelliseksi. Ennen kuin uuden version te-keminen aloitettiin, kerättiin lista vaatimuksia, jotka uuden ohjelman tulisi täyttää. Lista si-sälsi toiminnallisia vaatimuksia sekä varsin yleisiä vaatimuksia. Toiminnalliset vaatimukset tuotetta kohtaan muuttuivat ohjelmaa kehitettäessä. Tämä on varsin tyypillistä ohjelmistoli-i-ketoiminnassa. Kuten monissa ohjelmistoprojekteissa niin myös tässä projektissa vaatimuksia tarkennettiin prosessin edetessä. Tämän takia uuden ohjelman kehitys tapahtui iteratiivisesti siten että iteraatioiden tavoitteita tarkennettiin ja muokattiin projektin edetessä. Vesiputous-mallista ohjelmistokehitystä ei tähän projektiin edes harkittu. Tällä tavalla muuttuvat vaa-timukset saatiin joustavasti mukaan ohjelmistokehitykseen. Alla on lueteltuna korkeamman tason vaatimuksia, jotka pysyivät muuttumattomina koko tuotekehityksen ajan.

1. Uusi järjestelmä korvaa aikaisemmin käytössä olleen ohjelman.
2. Asetusten tekemisen tulee olla helposti laajennettavissa ja muokattavissa koskemaan verkon uusia tai muuttuvia parametreja.
3. Simulaation tulokset kertova karttanäkymä korvataan kokonaan uudella karttakompo-nentilla.
4. Verkon ominaisuuksia pitää pystyä muokkaamaan ilman, että käyttöliittymään tulee nä-kyviä muutoksia.

5. Simulaation käynnistäminen tulee olla mahdollista ilman erillistä koulutusta.
6. Simulaation käyttö ei saa kuluttaa tarpeettoman paljon tietokoneen muistia tai prosessoritehoa.

## 4.6 Tavoiteltu uusi käyttäjäkunta

Ohjelmalle asetettiin uusia toiminnallisia vaatimuksia. Uudet toiminnallisuudet antoivat mahdollisuuden käyttäjryhmän laajentamisen matkapuhelimiin sovelluksia tekeviin ihmisiin ja verkkoon uusia ominaisuuksia kehittäviin ihmisiin. Uudessa ohjelmassa ei myöskään olisi lissenssiongelmia, jolloin sitä voidaan markkinoida aikaisempaa laajemmalle käyttäjäkunnalle.

Alkuperäinen käyttöliittymä ei soveltunut vaikeakäyttöisyytensä takia suurelle käyttäjäkunnalle. Siinä oletettiin käyttäjien tietävän puhelinverkkoalan erikoistermistöä.

Ohjelmaa helppokäyttöisemmäksi muokkaamalla ja lisäämällä siihen uusia toiminnallisuuksia saatiin ohjelmalle uusia potentiaalisia käyttäjryhmiä. Selventämällä käyttöliittymän toimintaa ja käytettyjä termejä uskottiin, että matkapuhelimiin sovelluksia tekevät ihmiset voisivat ottaa ohjelman käyttöön. Toisaalta lisäämällä käyttöliittymään uusia verkon ominaisuuksia oletettiin, että ihmiset, jotka kehittävät verkon toimintoja, haluaisivat katsoa näiden toiminnallisuuksien vaikutusta todellisessa verkossa.

Uusista käyttäjryhmistä matkapuhelimiin sovelluksia tekevät ihmiset eivät ole niinkään kiinnostuneita verkon toiminnasta, vaan testaamastaan sovelluksesta. He haluavat tietää, kuinka sovellus toimii erilaisissa verkkoympäristöissä. Verkkojen ominaisuuksien säätäminen ei ole niin tärkeätä kuin alkuperäisellä käyttäjryhmällä. Sovelluskehittäjät haluavat vain tietää syyn, miksi heidän tekemänsä ohjelmisto toimii tietyllä tavalla tietyissä verkkoympäristöissä. Sovellusta ja sen eri versioita halutaan testata juuri tietyntilanteessa.

Verkon kehittäjät haluavat seurata, miten matkapuhelinverkko toimii juuri tietyillä asetuksilla. He voivat seurata matkapuhelimessa olevan sovelluksen toimintaa eri verkon asetuksilla ja seurata tekemiensä verkon asetusten vaikutusta sovelluksen toimintaan.

Toteutettavalla prosessilla haluttiin varmistaa, että ohjelman käyttö ei vaatisi tarpeettoman suurta asiantuntemusta ja että sen käyttöönotto onnistuisi alkuperäistä laajemmalla käyttäjäkunnalla. Tehtävä ohjelma suunnattiin asiantuntijoille, joilla on valmiudet ymmärtää matkapuhelimessa olevan sovelluksen ja verkon yhteistoiminta.

Työn aikana tuli vastata kahteen tärkeään kysymykseen: kuinka varmistetaan, että tuote soveltuu alkuperäiselle käyttäjäryhmälle ja samanaikaisesti myös uusille käyttäjäryhmille. Alkuperäisen käyttäjäryhmän edustajia oli koko prosessin ajan lähistöllä, joten heidän mielipiteitä saatiin koko projektin aikana ilman suurempia ponnisteluja. Suurempana haasteena olikin saada uusien käyttäjäryhmien mielipide näkyviin tuotekehitykseen.



# Luku 5

## Toteutunut prosessi

Tässä luvussa kuvataan toteutunut prosessi. Käyttöliittymän toteutukseen osallistui vain yksi henkilö<sup>1</sup> ja aikaa projektille oli varattu yhdeksän kuukautta. Tässä ajassa piti saada uusi ohjelma tehtyä, testattua ja dokumentoitua.

Projekti kuvataan aikajärjestyksessä ja se on jaoteltu neljään osaan. Jako on tehty osittain projektin aikana ja osittain jälkikäteen. Jokaiseen osioon löytyi jokin yhdistävä tekijä, jonka mukaan osiot ovat nimetty. Osiot ovat itsenäisiä kokonaisuuksia, jotka voidaan nähdä ISO 13407:n mukaisina iteraatioina. Jokaisen osion aikana on standardin mukaisesti tarkistettu tuotteen käyttöympäristöä ja vaatimuksia, tehty joko jonkinlainen prototyyppi tai tuoteversio sekä arvioitu siihen mennessä saatuja saavutuksia. Näin projektin aikana muuttuvat vaatimukset saatiin joustavasti mukaan tuotekehitykseen. Kuvattavat osiot ovat ongelmakenttään tutustuminen, ensimmäinen prototyyppi ja vaatimusten keräys, käytettävyydesti ja prototyypin sekä viimeisenä osiona tuotteen julkaiseminen.

### 5.1 Ongelmakenttään tutustuminen

Ensimmäinen osio rajattiin kahden kuukauden mittaiseksi, jonka aikana tavoitteena oli tutustua tehtäväalaaan ja luodaan seuraavalle osiolle mielekkäät tavoitteet.

---

<sup>1</sup>joka oli siis myös diplomityön kirjoittaja

### 5.1.1 Lähtötilanne

Puhelinverkkoala sisältää paljon lyhenteitä ja termejä. Aihealueeseen tutustuminen eli ensimmäinen osio toteutettiin tekemällä alkuperäisen kaltainen sovellusohjelma uudella ohjelmointiympäristöllä eli Java-ympäristöllä. Näin sekä puhelinverkkoala että Java-sovelluskehitys tulivat tutuiksi heti prosessin alussa. Koska osion aikana toteutettiin vain alkuperäisen kaltainen järjestelmä, toteuttamisessa ei tarvinnut ottaa kantaa, kuka tuotetta käyttää tai kuinka helppoa sen käyttö tulee olemaan. Näin osiosta saatiin riittävän lyhyt, jonka aikana vain perehdyttiin tehtävälleen yleisesti.

### 5.1.2 Toteutus

Tavoitteena oli, että kahden kuukauden jälkeen seuraavalle osiolle olisi laadittu selkeät tavoitteet ja aikataulu. Tähän toteutusvaiheeseen ei ollut tarkoitus tehdä mitään selvitystöitä alkuperäisen käyttöliittymän ongelmista tai puutteista. Oli kuitenkin tiedossa lukuisia muutoksia, jotka tuotteeseen pitää vielä tehdä. Ensimmäiseen osioon ei kuitenkaan otettu vielä mukaan näitä muutosideoita.

#### 5.1.2.1 Ensimmäinen prototyyppi

Osion aikana toteutettiin käyttöliittymä samannäköisenä uudestaan, mutta aikaisemmasta poiketen Java-ympäristöllä. Vaikka työn tuloksena oli samanlainen ohjelma kuin aikaisemmin, tehty työ ei mennyt hukkaan, sillä suurin osa ensimmäisen osion työstä liittyi ohjelmiston osiin, jotka eivät ole riippuvaisia käyttöliittymän näkymästä kuten yleinen parametrien käsittely. Nämä osa-alueet eivät muuttuneet käyttöliittymän ulkoasun muuttuessa ja tehtyä ohjelmakoodia pystyttiin käyttämään sellaisenaan myöhemmissä osioissa.

Koko käyttöliittymää ei toteutettu uudelleen. Simulaation tuloksista kertova aiemmin käytetty karttanäkymä otettiin käyttöön sellaisenaan. Karttaosuuden uudelleenkirjoittaminen arvioitiin niin työlääksi, että se päätettiin siirtää seuraavaan osioon. Näin ensimmäisen osion tärkeimpään tehtävään eli tehtävälleen tutustumiseen jäi riittävästi aikaa. Näin saatiin nopeasti ensimmäinen prototyyppi. Se helpotti keskustelua, kun prototyypillä pystyi konkreettisesti näyttämään, miltä uusi Java-toteutus näyttäisi, minkälaisia rajoitteita siinä tulisi olemaan ja minkälaisia mahdollisuuksia se tarjoaisi alkuperäiseen käyttöliittymään verrattuna.

Tilaajan toiveesta alkuperäinen käyttöliittymä haluttiin korvata mahdollisimman nopeasti uudella käyttöliittymällä. Uusi järjestelmä tai sen prototyyppi piti saada muutamassa kuukaudessa niin hyvin toimivaksi, että sillä voitiin korvata aikaisemmin käytössä ollut ohjelma.

### 5.1.3 Kokemukset

Ensimmäinen kahden kuukauden pituinen osio toimi tutustumisjaksona, jonka aikana kerättiin prosessin aikana tarvittavia tietoja ja tutustuttiin sovelluksen käyttäjäkuntaan sekä suunniteltiin seuraava iteraatio. Tänä aikana myös tehtäväalan erikoistermistö tuli tutuksi ja alkuperäisen sovelluksen mukainen ohjelma toteutettiin uudelleen Java-ympäristöllä.

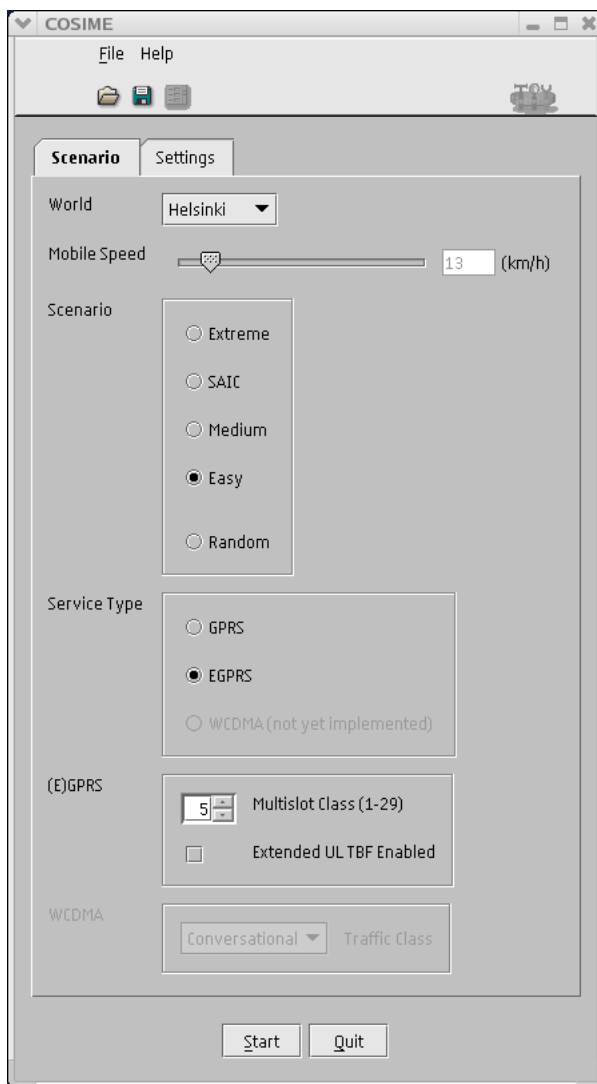
Shneidermanin mukaan suurin syy ohjelmistoprojektien epäonnistumiseen on suunnittelun puuttuminen prosessin alkuvaiheessa (Shneiderman, 1998, s. 104). Ensimmäinen osio, jossa ei kiirehditty lopullisen ulkonäön saamista, antoi hyvän mahdollisuuden prosessin suunnitteluun riittävän rauhallisesti. Koska vaatimukset tuotetta kohtaan muuttuivat koko projektin ajan, ei kaikkien seuraavien osioiden toteutusta lyöty lukkoon.

Ohjelmointitavoitteena ollut käyttöliittymän toteutus uudella ohjelmointiympäristöllä täyttyi ja uusi käyttöliittymä korvasi aikaisemman version. Parametrien ryhmittelyä ja nimiä vaihdettiin uutta käyttöliittymää tehdessä mutta toiminnallisuus pysyi samanlaisena. Näin saatiin ensimmäinen Javalla tehty toimiva prototyyppi toteutettavasta käyttöliittymästä.

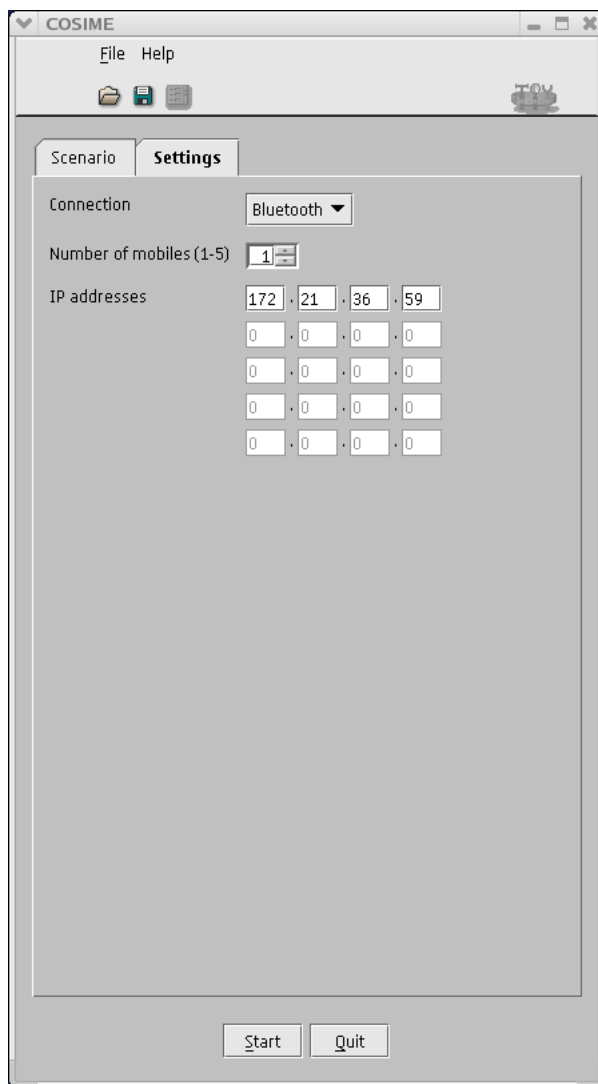
Tehtäväälaan tutustuminen sujui muita tavoitteita täyttäessä huomaamatta. Mitään erillistä koulutusta ei tarvittu. Alussa edes sovelluksen käynnistäminen ei onnistunut, sillä se sisälsi niin paljon outoja termejä. Iteraation lopussa oli jo selvää, kuinka ohjelma toimii ja mitä siinä olevat termit tarkoittavat.

## 5.2 Toinen prototyyppi ja vaatimusten keräys

Toinen osio tai iteraatio rajattiin myös kahden kuukauden mittaiseksi. Prototyypin valmistus ja vaatimusten keräykseen kerääminen aloitettiin, kun käytettävät työkalut ja tehtäväala olivat tulleet jo tutuiksi. Osion alkupuolella tehtiin nykyisten käyttäjien haastattelu sähköpostin välityksellä tavoitteena kerätä heidän vaatimuksiaan toteutettavalle tuotteelle. Näitä ideoita oli ke-



Kuva 5.1: Simulaation käynnistyksen parametrit, jotka liittyvät simulaation toimintaan. Käyttöliittymän runsaasti tilaa vievä rakenne ei sallinut sen laajentamista seuraavissa iteraatioissa.



Kuva 5.2: Simulaation käynnistyksen parametrit. Toisen välilehden tiedot koskivat yhteyden muodostamista testattavaan matkapuhelimeen.

rätty jo tehtävään tutustumisen aikana alkuperäisen käyttäjäryhmän ja kehitysryhmän sisällä. Haastattelulla pyrittiin laajentamaan ymmärrystä nykyisestä käyttäjäkunnasta sekä saamaan nykyisten käyttäjien kehitysehdotuksia mukaan tuotekehityksen seuraaviin vaiheisiin.

### 5.2.1 Lähtötilanne

Työn tilaajalla oli paljon uusia ideoita, jotka tuotteeseen haluttiin toteuttaa. Projektin alussa haluttiin painottaa erityisesti alkuperäisen käyttäjäryhmän vaatimuksia, koska ne olivat työn tilaajalla jo tiedossa. Kaikkien tiedossa olevien ideoiden ja toiminnallisuuksien tekemiseen kuluisi paljon pidempikin aika kuin osiolla varattu kaksi kuukautta. Siinä ajassa saatiin kuitenkin toteutettua niin iso käyttöliittymän laajennus ja muutos, että sillä voidaan korvata aiemmin käytössä ollut versio kokonaan.

Osion alussa haluttiin tutustua ohjelman alkuperäiseen käyttäjäryhmään, jotta voitiin varmistua, että työn tilaajan näkemys tuotteeseen tarvittavista ominaisuuksista vastaisi myös sen käyttäjien toiveita. Suunniteltiin sähköpostin välityksellä tehtävä kysely nykyisille käyttäjille. Jotta kahden kuukauden pituisen osion aikana ehdittäisiin toteuttamaan mahdollisimman suuri osa ideoista, päätettiin, että pääpaino on kuitenkin työn tilaajan jo aikaisemmin keräämien ideoiden toteuttamisessa. Kyselyn tavoitteeksi ei näin ollen asetettu uusien ideoiden keräämistä vaan käyttäjiin tutustuminen.

Osion aikana simulaation käynnistysosuus ja karttaosuus oli toteutettava uudestaan. Käynnistysosuus laajeni uusien toiminnallisten vaatimusten myötä niin paljon, että sen toimintaperiaate piti suunnitella uudestaan. Aiemmin käytetty käyttöliittymäosuus ei soveltunut toimintaperiaatteeltaan laajempaan käyttöön. Karttaosuuden lisenssi alkuperäisessä versiossa ei antanut mahdollisuutta sen laajentamiseen tai muokkaamiseen, joten myös kartta toteutettiin alusta asti uudelleen.

Koska sovelluksen tekemiseen ei otettu käyttäjiä mukaan osallistuvan suunnittelun mukaisesti, oli prosessin myöhemmissä vaiheissa valmistauduttava muokkaamaan käyttöliittymää, mikäli suunnittelijan malli ei vastannutkaan käyttäjän mallia. Vaikka käyttöliittymän ulkonäkö myöhemmissä vaiheissa vaihtuisikin, ei osion aikana tehty työ menisi hukkaan, sillä suurin työpanos liittyi sovelluksen yleiseen toimintaan ja karttanäkymään. Suuri osa toteutetusta työstä oli käyttöliittymäriippumatonta. Vaikka käyttöliittymä myöhemmin vaihtuisi ei esimerkiksi karttaosuuden ja simulaattorin väliseen kommunikointiin tarvitsisi koskea.

Tavoitteeksi osiolla asetettiin, että

- simulaation käynnistysnäkyä sekä karttanäkymä toteutetaan täysin Java-kielellä, niin että vanhan järjestelmän voi korvata uudella ohjelmalla
- mahdollisimman moni uusi idea ja toiminnallisuus toteutetaan
- saadaan jonkinlainen yleisnäkemys, tuotteen alkuperäisestä käyttäjäkunnasta
- varmistaa, että tilaajan ideat ja toiveet ovat oikeasti hyödyllisiä tuotteen käyttäjäkunnalle

## 5.2.2 Toteutus

Käyttöliittymän nykyiseen käyttäjäkuntaan tutustuttiin sähköpostitse tehtävällä kyselyllä, koska käyttäjät olivat sijoittuneet maantieteellisesti laajalle alueelle. Ilman laajamittaista matkustelua vain jonkinlaiset etähaastattelut ja -kyselyt, kuten videoneuvottelu, puhelinhaastattelu tai sähköpostikyselyt olivat mahdollisia.

### 5.2.2.1 Käyttäjäkysely

Käyttäjäkirjauksen mukaan alkuperäisellä sovelluksella oli 19 käyttäjää. Käyttäjät olivat jakaantuneet maantieteellisesti hyvin laajalle alueelle. Suomessa oli yhdeksän käyttäjää, USA:ssa seitsemän käyttäjää ja Kiinassa kolme käyttäjää. Sähköposti nähtiin parhaaksi tavaksi toteuttaa kysely tutustua käyttäjiin. Kysely päätettiin lähettää sähköpostilla suoraan sen viestiosassa, jotta käyttäjien olisi mahdollisimman helppoa ja nopeata vastata kyselyyn. Näin vastausprosentti oletettiin saatavan paremmaksi kuin liitetiedostona lähetettävään kyselyyn. Sähköpostin tekstiosan muotoiluun on vähemmän vaihtoehtoja kuin erillisessä liitetiedostossa mutta viestiin vastaaminen on näin vaivattomampaa. Kyselyn tarkoituksena oli kartoittaa tuotteen nykyisiä käyttäjiä, käyttötapoja ja -tottumuksia sekä kerätä kehitysideoita seuraavaa versiota varten. Samalla kyseltiin käyttäjien innokkuutta olla ylipäätään mukana tuotteen kehitystyössä ja antaa palautetta kehitettävästä tuotteesta. Käyttäjien käyttöaktiivisuudesta ei ollut aiempaa tietoa, joten se oli myös yhtenä kysymyksenä.

Kaksitoista käyttäjää eli 63% vastasi sähköpostikyselyyn. Näistä vastauksista neljä oli tyhjiä, koska vastaajat eivät olleet käyttäneet tuotetta. He olivat vain asentaneet sen itselleen. Vaikka täytettyjä vastauksia tuli vain kahdeksan, kysely tuotti runsaasti uusia ideoita sekä tietoa tuotteen käyttäjistä.

Simulaation käynnistämistä ja siihen liittyviä termejä käyttäjät pitivät melko helppona. Tämän kerrottiin johtuvan enemmänkin omasta perehtyneisyydestä tehtäväälaan kuin käyttöliittymän selkeydestä. Käyttäjät kertoivat myös katsovansa melko usein erillisiin tiedostoihin kertyviä lokitietoja selvittääkseen, miksi simulaatio toimi jollain tietyllä tavalla. Alkuperäinen käyttöliittymä ei tukenut tätä toimintatapaa. Käyttäjät joutuivat itse etsimään simulaatioon liittyvät lokitiedostot ja arvaamaan, mihin niistä olisi mahdollisesti tallentunut ongelmaan liittyvää tietoa. Useat käyttäjät toivoivat myös simulaation käynnistämiseen enemmän muunneltavia parametreja, mitä alkuperäisessä käyttöliittymässä oli mahdollista tehdä.

Yksikään käyttäjä ei ilmoittanut, että tuotteen karttaosuudessa olisi ollut ongelmia. Lisenssiongelma pakotti kuitenkin tekemään karttakirjaston uudelleen. Uuden karttakirjaston toiminnallisuus voitiin siis melko pitkälle kopioida vanhasta käyttöliittymästä, jolloin sen tulisi olla vähintään yhtä selkeä kuin se on nykyisellään.

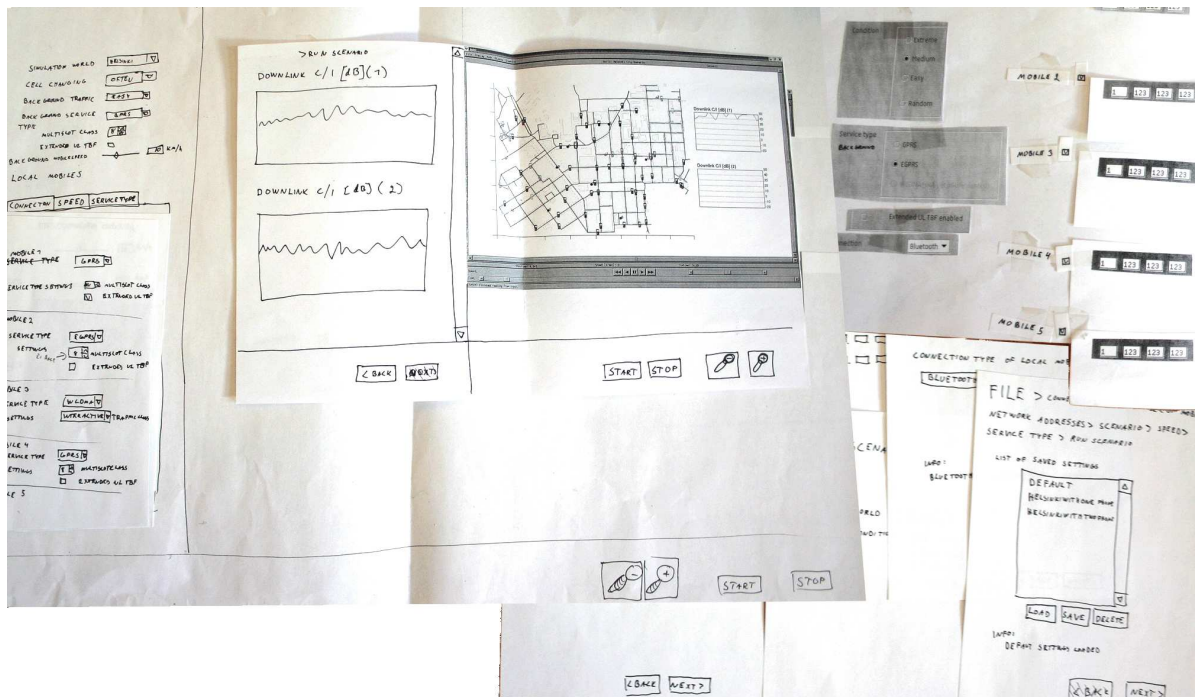
Kaikki käyttäjät olivat hyvin perillä verkon eri parametreista, joita käyttöliittymässä säädetään. Osa heistä olisi halunnut säätää parametreja huomattavasti laajemminkin kuin mitä käyttöliittymä antaisi mahdollisuuksia.

Käyttäjille lähetetty kysely on kokonaisuudessaan nähtävissä liitteessä A.

### 5.2.2.2 Prototyypin rakentaminen

Osion aikana toteutettiin simulaation käynnistyksen käyttöliittymä sekä uusi karttakirjasto. Simulaation käynnistyksen yhteydessä annettavien parametrien sijoitteluun oli runsaasti vaihtoehtoja. Näitä vaihtoehtoja käytiin läpi muun muassa paperiprototyypeillä (ks. kuva 5.3), jossa käyttöliittymäkomponentteja leikattiin ja teipattiin haluttuihin kohtiin. Vaihtoehtoja punnittiin käyttäen Nielsenin käytettävyyden määritelmää. Käytön tehokkuudelle haluttiin antaa suurin painoarvo, koska versio oli suunnattu erityisesti käyttäjäkunnalle, joka voitiin luokitella asiantuntijakäyttäjiksi.



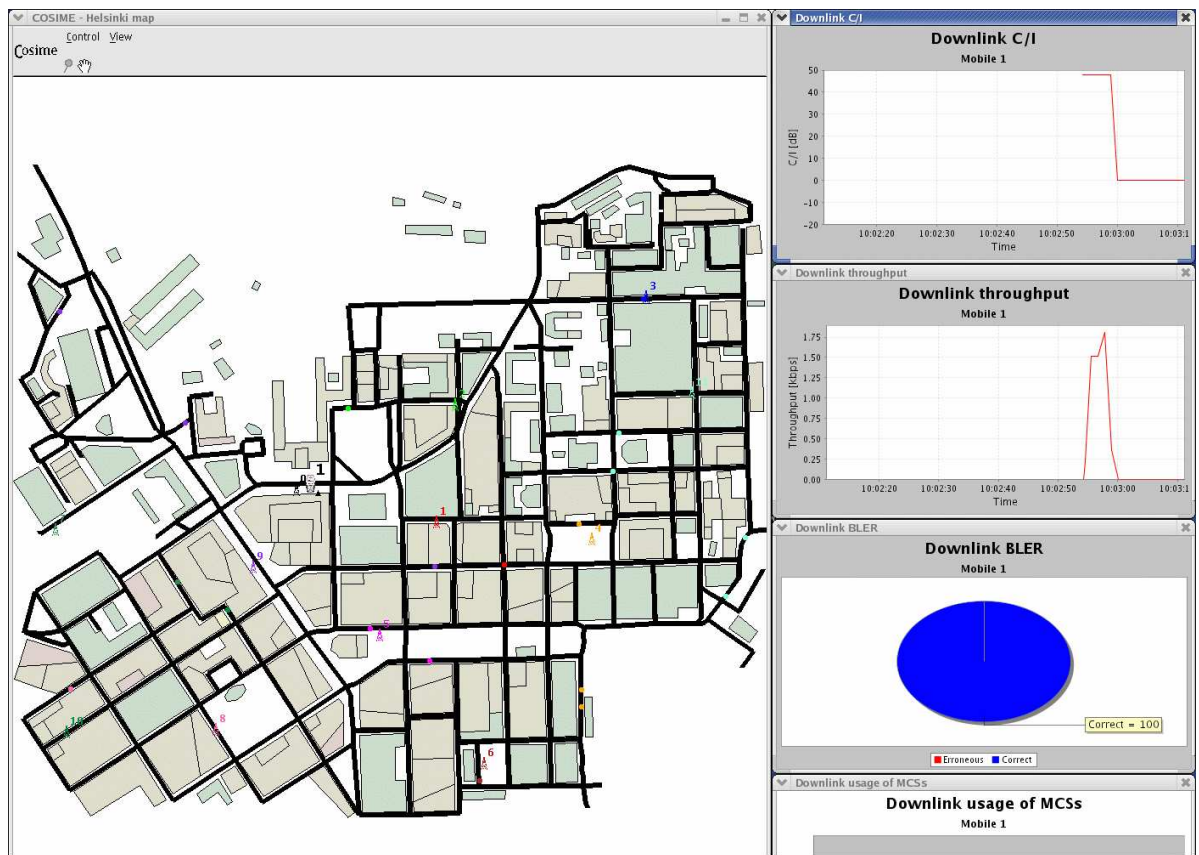


Kuva 5.3: Iteraation aikana tehtyjä paperiprototyyppejä

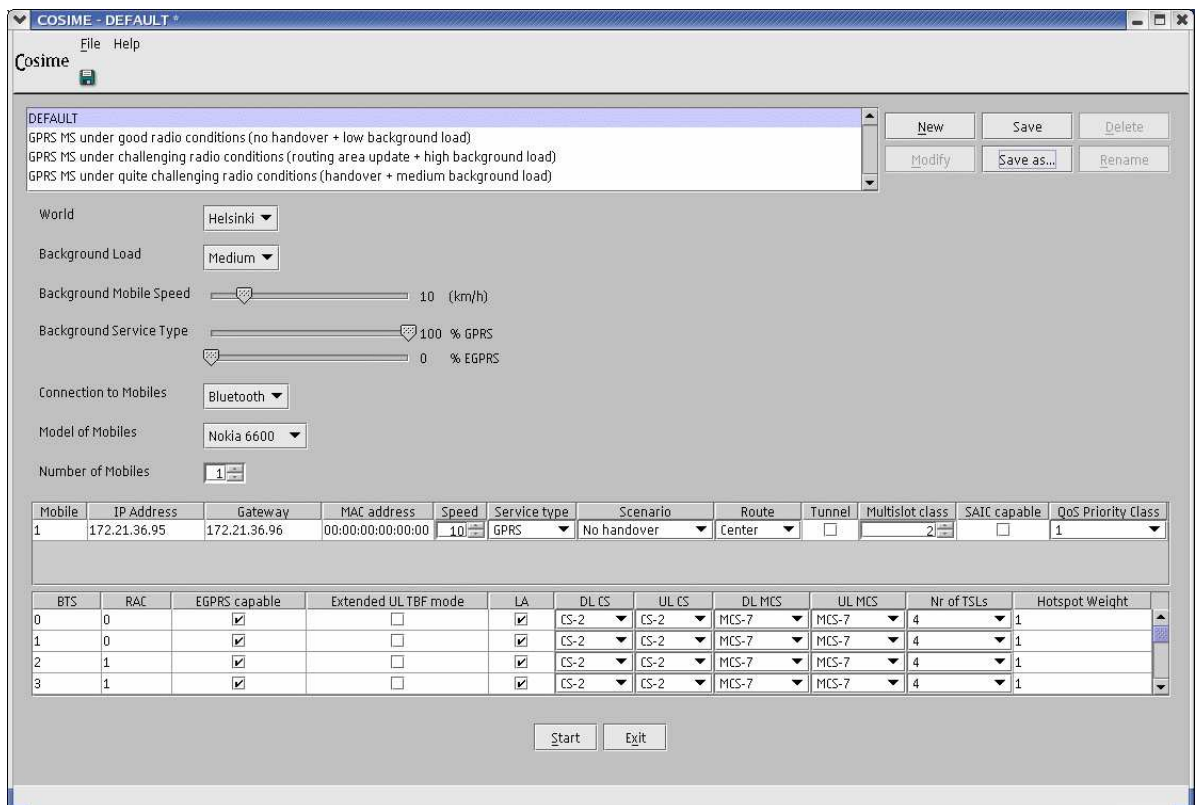
Aikaisemman version kartta oli osoittautunut toimivaksi, eikä käyttäjillä tehdyn kyselyn mukaan ollut ongelmia kartan kanssa. Osion aikana toteutettaviin kartan perustoiminnallisuuksiin kuuluvat kartan näyttäminen, zoomaus, kartan liikuttelu sekä kartalla liikkuvien matkapuhelimien sijainnin näyttäminen (kuva 5.4). Näiden toteutuksessa käytettiin aikaisemmin toimivaksi osoittautunutta tapaa. Matkapuhelimiin liittyy myös huomattava määrä simuloinnin aikana kertyvää tietoa. Se, mitä tästä tiedosta ja missä muodossa näytettäisiin käyttäjälle, jätettiin mietittäväksi prosessin myöhempisiin vaiheisiin. Tätä varten tarvitaan tarkempaa käyttäjien tuntemusta kuin mitä tällä hetkellä oli olemassa.

### 5.2.3 Yhteenveto

Alkuperäinen käyttäjäkunta voitiin luokitella asiantuntijakäyttäjiksi, joten heille käytön tehokkuus oli erittäin tärkeää. Kaikki muokattavat parametrit päätettiin sijoittaa yhteen ruutuun, vaikka ruutu tuli hieman ahtaaksi (kuva 5.5). Simulaation käynnistyksen yhteydessä muokattavien parametrien määrä yli kolminkertaistui alkuperäisestä yhdeksästä 29:ään.



Kuva 5.4: Karttanäkymä toisen iteraation jälkeen.



Kuva 5.5: Käyttöliittymä toisen iteraation jälkeen. Suurimmat ongelmat käytettävyydestissä liittyivät ruudun hahmottamiseen.

Toteutettu sähköpostikysely selkeytti kuvaa alkuperäisen käyttäjäryhmän luonteesta, vaikka sähköpostin välityksellä saatava kuva ei ollut täysin kattava. Mikäli nykyisiin käyttäjiin haluttaisiin tutustua paremmin, käyttäjien nykyisen työskentelyn havainnointi tai osallistuva suunnittelu voisi olla hyvä tapa. Osion aikana keskityttiin kuitenkin varmistamaan jo kerättyjen ideoiden tarpeellisuus alkuperäisellä käyttäjäkunnalla, ei niinkään keräämään uusia ideoita. Tässä sähköpostikysely toimi hyvin. Koska se lähetettiin vain tuotteen alkuperäisille käyttäjille, ei ollut varmuutta, kuinka ohjelma soveltuisi laajemmalle käyttäjäkunnalle. Keskittymällä aluksi vain alkuperäiseen käyttäjäkuntaan saatiin heille toteutettua mahdollisimman nopeasti alkuperäisen ohjelman korvaava parempi tuote, joka oli yksi ennalta asetetuista vaatimuksista.

## 5.3 Käytettävyydesti ja prototyypin parantelu

Kolmas osio kesti kolme kuukautta. Sen aikana suunniteltiin ja toteutettiin käytettävyydesti sekä jatkettiin prototyypin kehittämistä. Käytettävyydesti osoitti, että tehty käyttöliittymä oli kehityskelpoinen, ja että prototyypin laajentamista voitiin jatkaa.

### 5.3.1 Lähtötilanne

Projektissa oli tähän mennessä toteutettu käyttöliittymä, jota oikean käyttäjäryhmän edustajat eivät olleet kommentoineet. Tämän osion tavoitteena oli varmistaa, että tuotteesta tulee sellainen, jota sen käyttäjät todella tarvitsevat. Tähän menetelmänä oli käytettävyydesti, jonka tavoitteena oli saada käyttäjien mielipide käyttöliittymän kehityssuunnasta.

Tuotteen käyttäjäkuntaa oli laajennettu ohjelmaan tehtyjen muutosten mukana, joten käytettävyydestiin otettiin mukaan kolme tuotteen alkuperäisen käyttäjäryhmän ja kaksi uuden käyttäjäryhmän edustajaa.

Tässä osiossa myös laajennettiin prototyyppiä. Työn tilaajalla oli vielä toteuttamattomia ideoita, joita ei aiemmissa osioissa oltu tehty. Näitä ominaisuuksia lisättiin käyttöliittymään testien ja sen analysoinnin lomassa.

Käytettävyydestistä saadut ideat ja kommentit otettiin mukaan tuotekehitykseen, joten testien jälkeen oli tarkistettava tuotekehityksen suunta ja seuraavan iteraation tavoitteet.

### 5.3.2 Käytettävyystesti

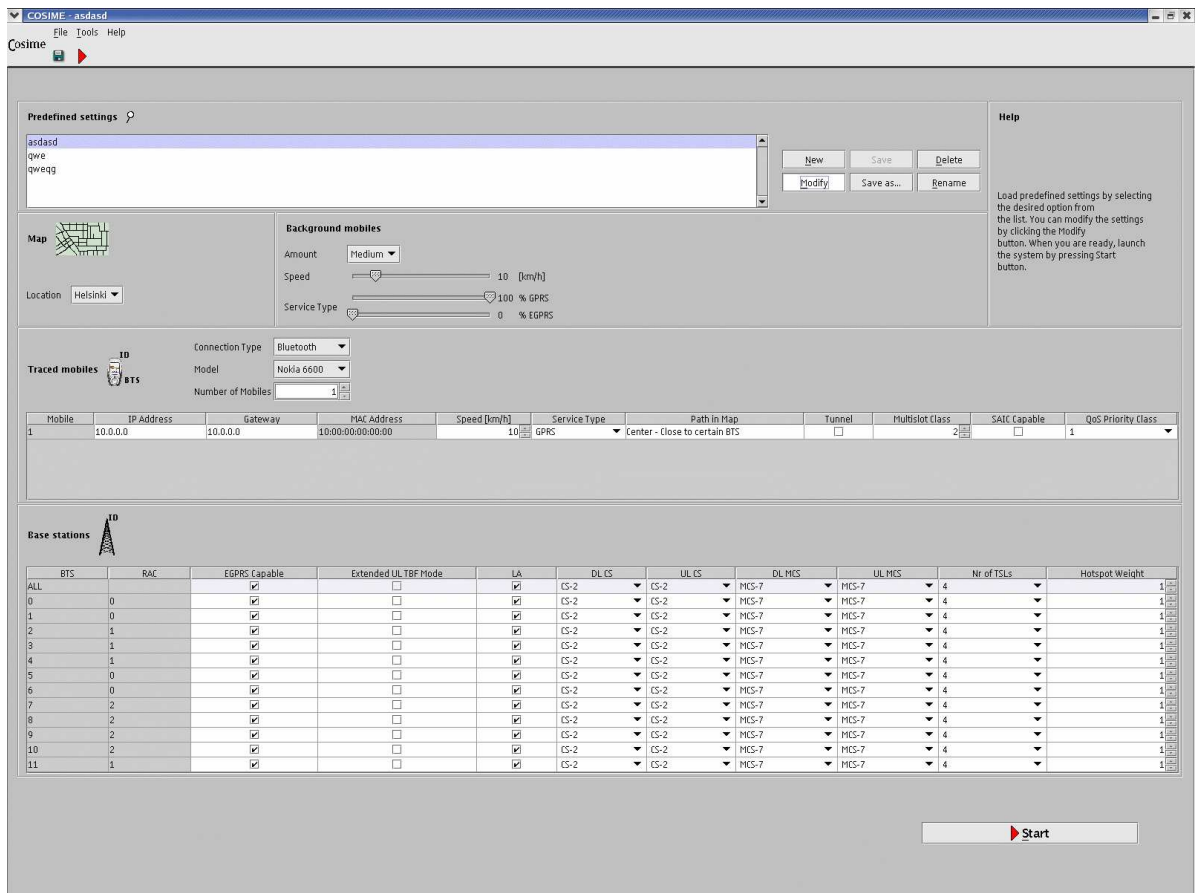
Käytettävyystesti suoritettiin työhuoneessa, johon oli viritetty videokamera. Kamera kuvasi testaajan takaa koneen näyttöruutua, jolloin nauhalta oli mahdollista seurata, mitä testikoneen ruudulla tapahtui. Ruutukuvan ja äänen perusteella analysoitiin testidata. Pilottitestin jälkeen kameraan lisättiin vielä erillinen mikrofoni lähemmäs käyttäjää, jolloin äänen laatu parani. Käyttäjä istui tietokoneen ääressä, johon testattava ohjelma oli asennettuna valmiiksi. Testin ohjaaja istui käyttäjän vieressä, jossa hän pystyi seuraamaan käyttäjän toimia ja antamaan käyttäjälle tarvittavia testitehtäviä. Hieman taaempana samassa huoneessa istui kirjuri, joka kirjasi testin aikana esiintyneet ongelmat, parannusehdotuksen ja jatkokehitysideat.

Testausmenetelmänä oli vapaan läpikäynnin tyylinen testi, jossa käyttäjä sai vapaasti käyttää ohjelmaa. Ennen testiä käyttäjälle kerrottiin testin kulusta ja käytettävyystestin periaatteet. Käyttäjän tuli testin aikana käydä läpi kaikki tärkeät ohjelman osiot. Mikäli käyttäjä ei oma-aloitteisesti löytänyt haluttuja ohjelman osioita, häntä pyydettiin tekemään tehtävä, joka edellytti puuttuvan osion läpikäyntiä. Testin lopuksi käytiin vielä läpi kaikki käyttöliittymän elementit ja tarkennettiin, miten käyttäjä ymmärsi ne käytettyään ohjelmaa jo jonkin aikaa.

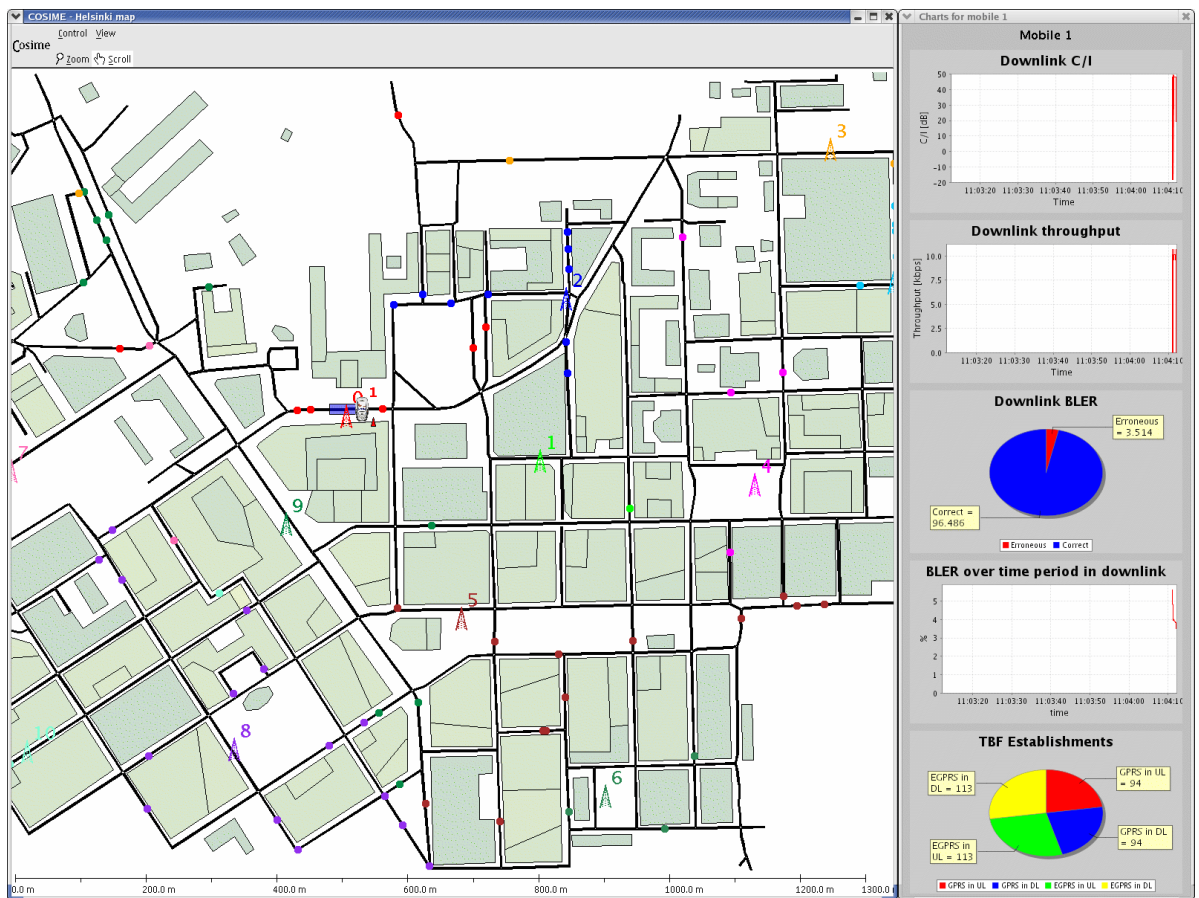
Testit analysoitiin videoiden avulla, jotta kaikki testeissä ilmenneet ongelmat löytyvät ja tulevat käytyä läpi. Ongelmista tehtiin lista ja niille annettiin vakavuusluokka, korjausehdotus sekä arvioitu korjauksen kesto. Tähän ongelmalistaan kerättiin ohjelmaan tulleet uudet ideat sekä jo entuudestaan tiedossa olleet vaatimukset. Näin saatiin työlista, jossa oli kaikki ohjelmaan tiedossa olevat muutostarpeet.

### 5.3.3 Prototyypin parantelu

Testien tuloksena oli työlista ohjelman muutostarpeista. Tähän mennessä tehtyä prototyyppiä kehitettiin edelleen kohti tulevaa tuotetta. Kun tärkeimmiksi luokitellut muutokset oli tehty, saatiin kuvien 5.6 ja 5.7 mukaiset käyttöliittymänäkymät. Asetusnäkyman suurin muutos edelliseen versioon verrattuna oli, että ruutu jaoteltiin selvemmin eri osioihin. Ruutuun lisättiin myös osio, josta käyttäjät löytävät apua käyttöliittymän käyttöön ja käyttöliittymässä esiintyviin parametreihin.



Kuva 5.6: Käyttöliittymän asetusnäkökulma kolmannessa osiossa tulleiden muutosten jälkeen. Käyttöliittymän rakennetta on parannettu ympäröimällä yhtein liittyvät parametrit.



Kuva 5.7: Karttanäkymä kolmannessa osiossa tulleiden muutosten jälkeen.

### 5.3.4 Tilanne osion jälkeen

Noin kaksi kuukautta ennen projektin päättymistä tuotteen käyttöliittymään oli toteutettu tärkeimmiksi katsotut ominaisuudet. Projektille asetetut toiminnalliset vaatimukset oli saatu täytettyä. Käyttöliittymä sisälsi nyt ominaisuudet, joita sen alkuperäinen käyttäjäryhmä ja työn tilaaja olivat halunneet. Projektin aikana oli myös kerätty vaatimuksia tuotteen uusilta käyttäjäryhmiltä. Käyttöliittymällä pystyi tekemään kaikki sille annetut toiminnallisuudet. Tuotteessa oli kuitenkin vielä käytettävyysoongelmia.

## 5.4 Tuotteen julkaiseminen

Ennen tuotteen julkaisemista se piti testata virhetilanteiden varalta. Viimeiselle osiolle oli varattu kuukausi aikaa. Se käytettiin käytettävyyden hiomiseen. Ohjelmaan oli tehty muutoksia edellisten käytettävyydestien jälkeen, joten ennen julkaisemista haluttiin tehdä uusi käytettävyydesti. Osion aikana korjattiin esiin tulleet ohjelmalliset virheet ja käytettävyydesteissä ilmenneet ongelmat. Osion lopuksi ohjelman ensimmäinen virallinen versio julkaistiin.

### 5.4.1 Käytettävyydesti

Kun tuotteelle asetetut toiminnalliset vaatimukset oli saatu valmiiksi, tuotetta voitiin kutsua ominaisuuksiensa puolesta valmiiksi. Ennen tuotteen julkaisemista ohjelman toimivuus ja hyvä käytettävyys haluttiin varmistaa. Samalla haluttiin varmistaa, että tuote todella soveltuu niin alkuperäisen kuin uudenkin käyttäjäryhmän edustajille. Näiden seikkojen selvittämiseksi tuotteelle tehtiin uusi käytettävyydesti. Testissä pyrittiin vastaamaan seuraaviin kysymyksiin:

1. Onko käyttöliittymä alkuperäisen käyttäjäryhmän mielestä selkeytynyt?
2. Onko käyttöliittymässä ominaisuuksia, joita alkuperäinen käyttäjäryhmä ei tarvitse?
3. Onko käyttöliittymässä piirteitä, jotka ovat häiritseviä?
4. Kuinka uudet käyttäjäryhmän edustajat suhtautuvat tuotteeseen?
5. Soveltuuko ohjelma matkapuhelimeen sovelluksia tekeville?



Kohtia 1-3 selvitettiin ottamalla testiin kaksi alkuperäisen käyttäjäryhmän edustajaa ja kohtia 4 ja 5 selvitettiin kolmen uuden käyttäjäryhmän edustajan avulla. Uuden käyttäjäryhmän edustajiksi valittiin matkapuhelimen Symbian Series 60 alustalle ohjelmoivia työntekijöitä. Kaikista testeistä etsittiin esitettyjen kysymysten lisäksi myös käytettävyysoongelmia. Testitieto kerättiin pikadata-analysointimenetelmällä (ks. kappale 3.3.1).

Testin suorittamista varten tietokoneelle asennettiin testattava ohjelma. Kaikki tietokoneen asetukset jätettiin sellaisiksi, kuin ne oletusarvoisesti ovat asennuksen jälkeen. Käyttäjä joutui näin tekemään ohjelmaan sen käyttönoton niin kuin se tapahtuu todellisessa tilanteessa asennuksen jälkeen. Käyttäjä istui tietokoneen edessä ja hänellä oli käytössään ohjelman testaamiseen tarvittava matkapuhelin (ks. kuva 5.8). Testin ohjaaja istui käyttäjän vieressä ja kirjuri käyttäjästä katsottuna takavasemmalla.

Testit varmistivat, että tuote oli selkeytynyt aikaisempiin versioihin verrattuna. Kukaan ohjelman käyttäjistä ei tarvinnut kaikkia ohjelman ominaisuuksia. Ylimääräisiksi katsotut osiot eivät kuitenkaan häirinneet käyttöä ja nämä ominaisuudet osoittautuivat tarpeellisiksi joillekin käyttäjille. Ohjelman rakenne nähtiin selkeänä eikä siinä ollut häiritseviä ominaisuuksia. Uuden käyttäjäryhmän edustajat saivat suoritettua hyvin kaikki heille annetut testitehtävät ja he pitivät ohjelmaa hyödyllisenä. Ohjelmasta löydettyjen käytettävyysongelmien takia he eivät kuitenkaan pitäneet sitä vielä julkaisukelpoisena. Simulaation käynnistämässä oli muutamia vakaviksi luokiteltuja käytettävyysoongelmia, etenkin ohjelman käynnistyksen yhteydessä antamassa alkupalautteessa ja mahdollisissa virheilmoituksissa. Kaiken kaikkiaan testeissä ilmeni 33 käytettävyysongelmaa. Niistä 10 oli vakavia ja 9 merkittäviä ongelmia. Loput ongelmista olivat vähäisiä tai kosmeettisia. Käytettävyysongelmien määrä oli puoliintunut ensimmäiseen käytettävyydestään verrattuna.

### 5.4.2 Lopullinen tuote

Ennen ohjelman julkaisua kaikki vakavat ja merkittävät ohjelmasta löydetty käytettävyysongelmat korjattiin. Muutamia vähäisiksi katsottuja ongelmia, jotka ovat työmäärällisesti suuria, jätettiin korjaamatta. Käyttöliittymän tarjoamaa palautetta parannettiin käyttämällä selvästi erottuvaa punaista taustaväriä sellaisten asetusten kohdalla, jotka olivat virheellisiä. Käyttöliittymässä olevia ohjeita ja virheilmoituksia parannettiin myös niin, että ne kuvasivat kyseistä tilannetta paremmin.

Simulaation käynnistysnäkyvässä (kuva 5.9) oli yhteen kuuluvat asiat lokeroitu hahmottamisen helpottamiseksi. Oikeassa ylänurkassa oli ohje, joka auttoi kunkin käyttöliittymäelementin ymmärtämisessä. Karttanäkyvässä (kuva 5.10) oli erillinen ikkuna graafisille kuvaajille, joita pystyi poistamaan ja lisäämään tarpeiden mukaan.

Ohjelma toteutettiin modulaarisesti niin, että karttaa oli mahdollista käyttää erillisenä kirjastona. Käyttöliittymän muokkaaminen tehtiin mahdollisimman helpoksi, sillä projektin päättyessä oli jo tiedossa, että ohjelman kehitys ei tule loppumaan tehtyyn 1.0 versioon. Simulaation käynnistämiseen liittyvien parametrien muokkaamista ja lisäämistä pystyi tarvittaessa tekemään ilman että käyttöliittymään tarvitsee koskea ollenkaan. Toteutettu käyttöliittymä sisälsi yli 17 000 riviä ohjelmakoodia.

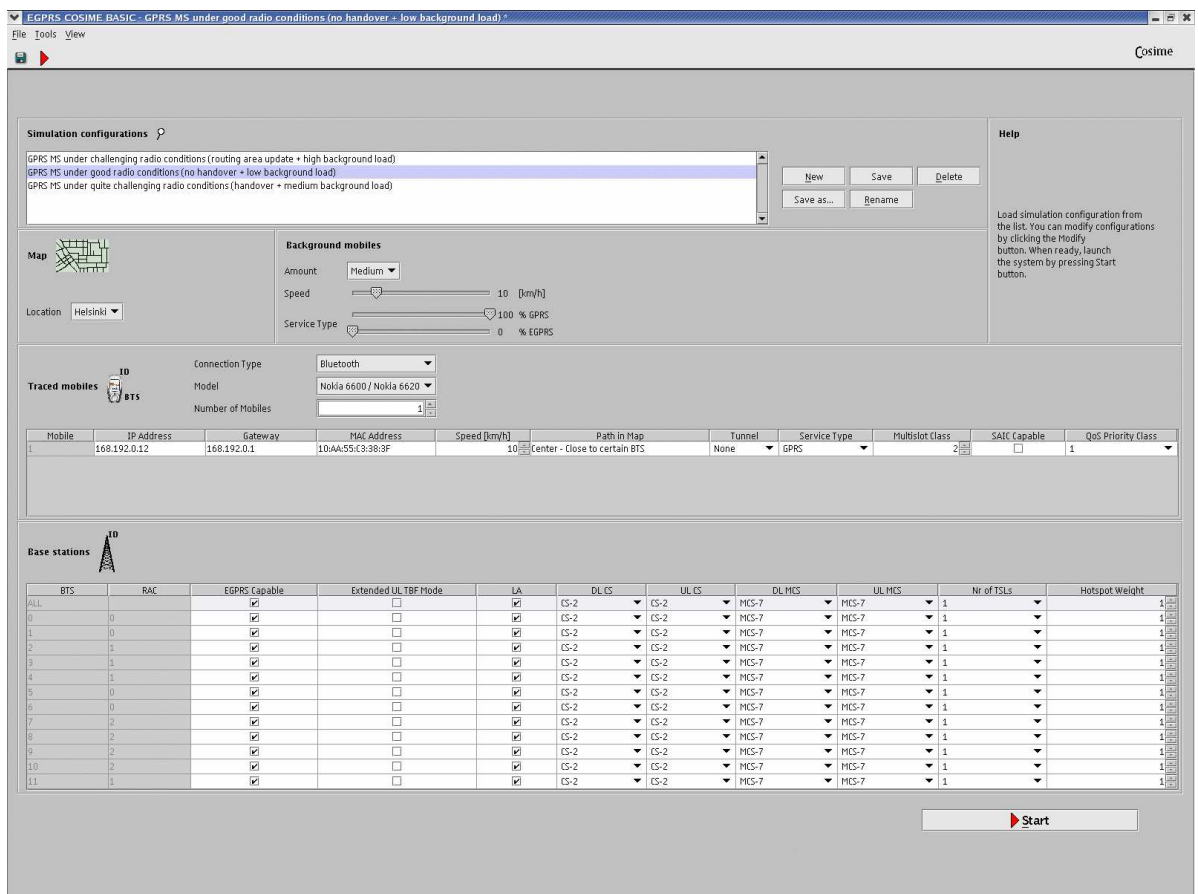
### 5.4.3 Yhteenveto

Tuote saatiin julkaistua aikataulun mukaisesti ja se todettiin käytettävyydeltään riittävän hyväksi. Tuote toteutti sille asetetut toiminnalliset vaatimukset. Käyttöliittymälle suoritettiin kaksi käytettävyydestä ennen sen julkaisemista ja lähes kaikki löydetyt käytettävyysongelmat korjattiin.

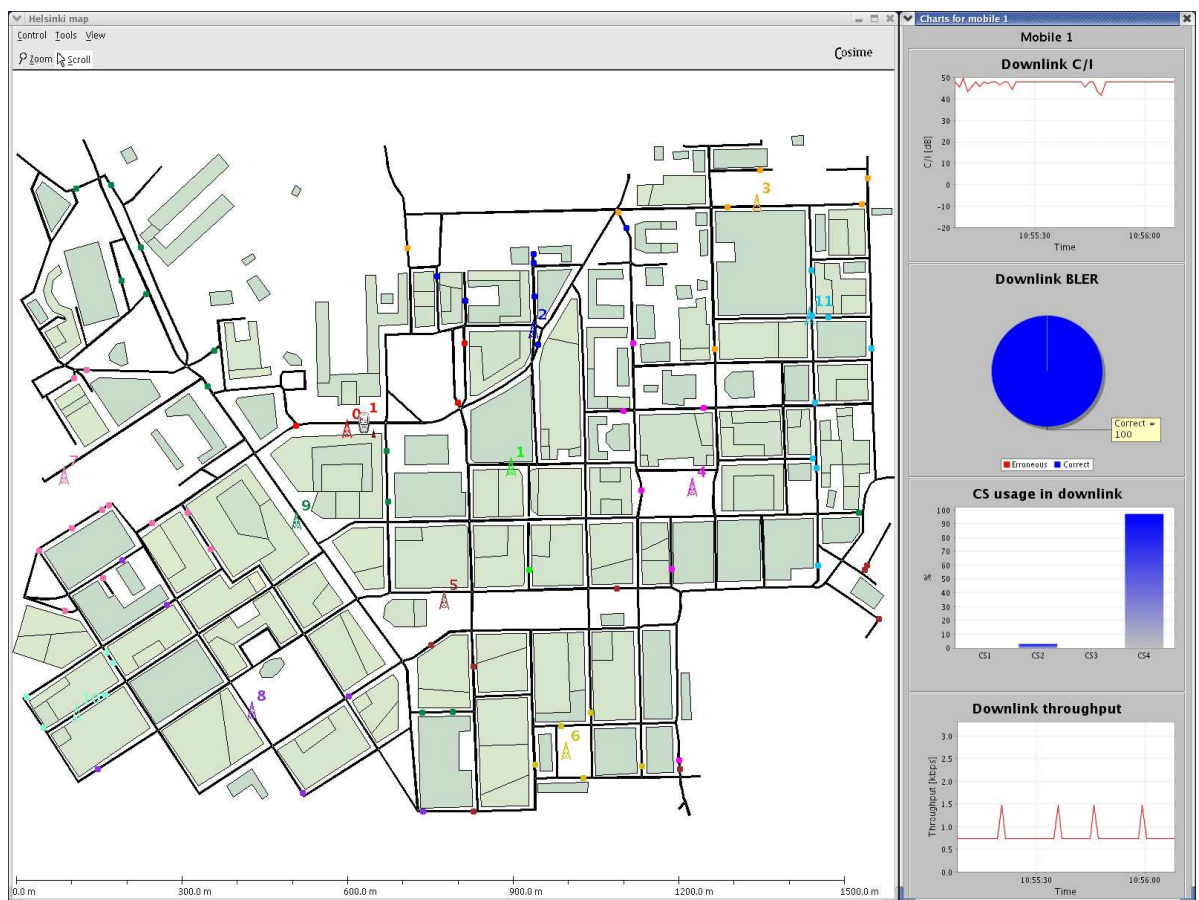
Verkon muuttuvista ominaisuuksista sekä simulaation käyttötarpeista johtuen sovellusta on kuitenkin kehitettävä jatkuvasti. Tämän projektin päättyessä sovellus oli kuitenkin valmis alkuperäistä käyttäjäryhmää laajemman käyttäjäryhmän käyttöön. Suoritettujen testien perusteella ohjelma tukee sekä uusien että vanhojen käyttäjäryhmien tiedossa olevia työskentelytapoja.



Kuva 5.8: Käytettävyydestiin osallistunut käyttäjä suorittamassa käytettävyydestiä. Testin ohjaaja istui käyttäjän oikealla puolella ja kirjuri testaajasta katsottuna takavasemmalla.



Kuva 5.9: Valmis käyttöliittymä, joka julkaistiin 1.0 versiona.



Kuva 5.10: Valmiin ohjelman karttanäkymä

# Luku 6

## Johtopäätöksiä ja pohdintaa

Työn aikana laajennettiin asiantuntijajärjestelmän käyttäjäkuntaa. Tiivistelmänä toteutuneesta prosessista voidaan kuvata, että se tehtiin neljässä iteraatiossa. Ensimmäisessä vaiheessa tutustuttiin tehtävälleen. Toisessa vaiheessa toteutettiin tuote, jossa oli huomioitu alkuperäisen käyttäjäryhmän vaatimuksia. Kolmannessa vaiheessa käyttöliittymää muokattiin niin, että se soveltuu myös uuden käyttäjäryhmän edustajille. Neljännessä ja viimeisessä vaiheessa tehdyn ohjelman käytettävyyttä parannettiin sekä uusien että alkuperäisten käyttäjien näkökulmasta.

Työn aikana tutkittiin, mitä tekijöitä on otettava huomioon, kun asiantuntijajärjestelmän käyttäjäkuntaa lähdettiin laajentamaan.

### 6.1 Eri käyttäjäryhmien huomioiminen

Tehdyssä ohjelmistoprojektissa työn tilaaja kuului tuotteen alkuperäiseen käyttäjäryhmään. Työn tilaajan toiveilla oli erittäin suuri painoarvo, koska kyseessä oli suoraan tilaajan kanssa tehtävä yhteistyö. Oletus oli, että tilaaja on omien tarpeidensa asiantuntija, tilaajalta tulevia vaatimuksia ei arvioitu kriittisesti. Tämä ei välttämättä johda käytettävyydeltään parhaaseen mahdolliseen lopputulokseen, mikäli tuotteella on myös muita käyttäjäkuntia. Koska tuotteen haluttiin myös muiden käyttäjäkuntien näkökulma, tilaajan edustama käyttäjäryhmä sai suhteettoman paljon huomiota muiden käyttäjäryhmien kärsiessä tilanteesta. Vaatimusmäärittelyvaiheessa muista käyttäjäryhmistä tulevia ideoita ei kerätty riittävästi, sillä tilaajan puo-

lelta vaatimuksia tuli jo niin paljon kuin projektin aikana ehdittiin toteuttaa. Lisävaatimuksia toisista käyttäjäryhmistä ei ollut enää tarvetta etsiä.

Työ tehtiin työn tilaajan tiloissa ja keskuudessa osallistuvan suunnittelun oloisesti. Tämä osoittautui erittäin toimivaksi ratkaisuksi. Työn tilaajan ja siis myös yhden käyttäjäryhmän vaatimukset tulivat ohjelmistokehitykseen erittäin luontevasti mukaan. Koko projektin aikana pidettiin vain yksi muodollinen palaveri. Muuten kaikki ajatusten vaihto tapahtui varsin vapaamuotoisesti ja luontevasti käytäväkeskusteluina tai prototyyppejä katsottaessa. Alkuperäisen käyttäjäryhmän ideat saivat riittävästi huomioarvoa, ja tehty ohjelma tukee heidän työskentelytapojaan hyvin. Tilanteessa, jossa työn tilaaja kuuluu tuotteen käyttäjäryhmään ja tuotteella ei ole muita käyttäjäryhmiä, järjestely olisi ollut paras mahdollinen. Kyseessä olevalle tuotteelle haluttiin kuitenkin käyttäjäryhmää laajentaa.

Tehdyillä käytettävyydestestauksilla ja haastatteluilla saatiin uusien käyttäjäryhmien näkökulma prosessiin. Vapaamuotoisten käytettävyydestien perusteella lopputuote tukee heidän käyttötarpeitaan. Käyttöliittymään lisätyt ominaisuudet sekä uuden käyttöliittymän helppokäyttöisyys antaa mahdollisuuden että tutkittujen uusien käyttäjäryhmien edustajat voisivat ottaa ohjelman käyttöönsä.

Tuotteen markkinointi laajemmalle käyttäjäkunnalle alkoi toden teolla vasta tämän projektin päätyttyä. Olisikin mielenkiintoista seurata, muodostuuko käyttäjäkunta tehtyjen oletusten mukaiseksi. Käyttöliittymässä ei ole ominaisuuksia, jotka varsinaisesti sulkisivat mitään käyttäjäryhmää pois, vaikka prosessin aikana tehdyt oletuksen uusista käyttäjäryhmistä eivät osuisikaan täysin oikeaan.

Eri käyttäjäryhmät löysivät käytettävyydesteissä samoja ongelmia. Vaikka näkökulma koko prosessin aikana oli keskittynyt alkuperäiseen käyttäjäryhmään, voidaan olettaa, että ohjelman käyttö onnistuu helposti myös muiden käyttäjäryhmien edustajilta. Mielenkiintoiseksi jatkotutkimukseksi jää, minkälaiseksi tuotteen käyttäjäkunta muodostuu ja kuinka hyvin tehty ohjelma palvelee uusia käyttäjäryhmiä.

## 6.2 Käytetyt menetelmät

### 6.2.1 Prototyypit ja osallistuva suunnittelu

Prototyypit yhdistettynä osallistuvaan suunnitteluun oli työn keskeisimpiä menetelmiä, joilla käyttäjien näkökulma saatiin mukaan ohjelmistokehitykseen. Prototyypit olivat osallistuvassa suunnittelussa keskustelun innoittajana.

Alkuperäinen käyttäjäryhmä tuli mukaan projektiin osallistuvan suunnittelun kaltaisesti, koska ohjelmistokehitys tapahtui heidän keskuudessaan. Tämä käytäntö oli erittäin toimiva. Ideoita ohjelmalle asetetuista vaatimuksista tuli muodollisesti palaverissa mutta pääosin epämuodollisesti käytävällä keskustellessa tai muuten ajatuksia vaihtaessa. Koska työn tilaaja kuului alkuperäiseen käyttäjäryhmään, käytetty menetelmä oli varsin hyvä.

Työn alkuvaiheilla käytettiin nopeasti tehtyjä paperiprototyyppejä. Niillä saatiin kokeiltua erilaisia käyttöliittymäversioita ennen tuotteen rakentamista. Tuotteen rakentaminen tehtiin iteraatiivisesti jatkaen aina siitä, mihin oli jääty. Tuote oli aluksi vain yksinkertainen prototyyppi. Sitä paranneltiin kunnes projektin lopussa prototyypistä oli tullut virallinen lopputuote eli toimiva ohjelmisto.

### 6.2.2 Kyselyt

Projektin alkupuolella tehtiin sähköpostikysely. Kyselyn tarkoituksena oli hahmottaa tuotteen alkuperäistä käyttäjäryhmää laajemmin. 63% kyselyn saaneista vastasi kyselyyn. Ennen kyselyä tuotteen käyttäjistä ei ollut selkeää kuvaa. Vastauksien perusteella selvisi ohjelman käyttötapa ja tottumuksia. Käyttäjät eivät käyttäneet ohjelmaa päivittäin, vaan viikoittain tai jopa kuukauden välein ja aloitettuaan tekivät kerralla useampia simulointeja.

Jotta kyselystä olisi saanut enemmän tietoa, siinä olisi pitänyt olla enemmän tilaa vapaalle tekstille. Vastauksista esimerkiksi selvisi, kuinka helppoa ohjelman käyttö alkuperäisistä käyttäjistä oli asteikolla 1-5. Tämä ei kuitenkaan vielä anna tietoa siitä, mikä käytössä on helppoa ja mikä vaikeata. Käytännössä vain kysymyksistä, joihin vastattiin kirjallisesti, saatiin ohjelman kehitystyön kannalta merkittävää tietoa. Projektista saadun kokemuksen perusteella voidaan todeta, että pelkkä sähköpostin välityksellä tehtävä kysely ei riitä kattavan kuvan saamiseksi käyttäjistä.



### 6.2.3 Asiantuntija-arvio

Tuotteen kehittäjä pyrki pitämään mielessään Nielsenin heuristiikat sekä muut hyvät käytettävyyden periaatteet. Tämän lisäksi yksi ulkopuolinen arvioija katsoi ja kommentoi tehtyjä paperiprototyyppejä ja ruutukaappauskuvia. Näin saatiin arvokasta ulkopuolista näkemystä ohjelman toiminnallisuudesta ja sen ulkonäöstä. Varsinaisia usean asiantuntijan asiantuntija-arviota tuotteelle ei tehty missään vaiheessa. Projektin aikana käytettiin käytettävyydestä käytettävyyden parantamiseksi. Käytettävyydesteissä pystyi selvittämään oikeiden käyttäjäryhmän edustajien toimintatapoja ja työmalleja. Asiantuntija-arvioita ei nähty tarpeellisena menetelmänä käytettävyydestien lisäksi.

### 6.2.4 Käytettävyyden arviointi käyttäjien kanssa

Käytettävyydestit osoittautuivat erittäin toimiviksi tavoiksi parantaa ohjelman käytettävyyttä. Ohjelmistokehityksessä mukana olleet tulivat monelta kohdin sokeiksi ohjelmassa oleville ongelmille, koska he käyttivät ohjelman prototyyppiä lähes päivittäin. Tehdyt kaksi käytettävyydestiä olivat kummatkin melko vapaamuotoisia. Testeissä oli mukana sekä alkuperäisen käyttäjäryhmän että uuden käyttäjäryhmän edustajia. Vapaammalla läpikäynnillä pyrittiin saamaan parempaa kuvaa ohjelman käyttötarpeesta. Tärkeänä tavoitteena oli löytää ohjelmassa olevat käytettävyysongelmat ja varmistaa, että tuote todella soveltuu eri käyttäjäryhmille. Tähän tarkoitukseen vapaamuotoiset käytettävyydestit soveltuivat erittäin hyvin. Käytettävyydestit paljastivat tuotteen käytettävyysonget ja tarkensivat kuvaa tuotteen käyttäjäryhmistä.

## 6.3 Aikataulu, resurssit ja laajuus

Tutkittavaksi valitussa ohjelmistoprojektissa oli etukäteen tiukasti määritelty aikataulu sekä resurssit. Ohjelma julkaistiin aikataulunsa mukaisesti annetuilla resursseilla. Kaikki sille asetetut toiminnalliset vaatimukset tulivat myös täytettyä. Ohjelmistokehityksen “kolmijalka” *resurssit, aika ja laajuus* pysyi tasapainossa. Tämä osoitti tavoitteiden määrittelyn onnistuneen. Ohjelmistokehitystä teki yksi henkilö yhdeksän kuukautta. Tänä aikana syntynyt käyttöliittymä sisälsi yli 17 000 riviä ohjelmointikoodia. Syntynyt ohjelma oli modulaarinen ja se oli helposti muokattavissa tulevia muutoksia silmälläpitäen.

## 6.4 Yhteenveto

Projekti valmistui onnistuneesti. Tehty ohjelma soveltuu hyvin alkuperäiselle käyttäjäryhmälle sekä tehtyjen tutkimusten mukaan myös laajemmalle käyttäjäkunnalle. Tuotteen käyttäjäkunnan laajentaminen jää nyt työn tilaajan harteille. Käyttöliittymältään tuote soveltuisi alkuperäisten 19 asiantuntijakäyttäjän lisäksi matkapuhelimeen sovelluksia tekeville sekä verkko-tekniologioita suunnitteleville ihmisille. Heitä on Suomessakin jo satoja, joten maailmanlaajuinen käyttäjäpotentiaali tulee olemaan moninkertainen alkuperäiseen käyttäjäkuntaan verrattuna.

Osallistuva suunnittelu yhdessä prototyyppien kanssa osoittautui hyväksi tavaksi varmistaa, että tuote soveltuu sen tiedossa olevalle käyttäjäryhmälle. Vapaamuotoiset käytettävyysettit laajensivat kuvaa uusista käyttäjäryhmistä samalla, kun testit paljastivat tuotteessa olevia käytettävyyssongelmia.

Tilaaaja oli tyytyväinen lopputulokseen. Ainakaan simulaation käyttöliittymästä johtuen matkapuhelinteollisuudella ei ole syytä huoleen. Siksi myös tekijä voi olla tyytyväinen.

# Lähteet

- (Atkinson, 1999) Atkinson R., 1999, Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria, International Journal of Project Management Vol. 17, Elsevier Science Ltd & IPMA
- (Bevan, 2001) Bevan N., 2001, International Standards for HCI and usability, Human-Computer Studies, Academic Press
- (Beyer & Holtzblatt, 1998) Beyer H., Holtzblatt K., 1998, Contextual Design: Defining Customer-Centered Systems, Morgan Kaufmann Publisher
- (Faulkner, 2000) Faulkner X., 2000, Usability Engineering, Palgrave
- (Haikala & Märijärvi, 2001) Haikala I., Märijärvi J., 2001, Ohjelmistotuotanto, Talentum Media
- (Hall, 2001) Hall R. R., 2001, Prototyping fir usability of new technology, Human-Computer Studies, Academic Press
- (Hom, 2004) Hom J., The Usability Method Toolbox - Contextual Inquiry, <http://jthom.best.vwh.net/usability/context.htm>, viitattu 7.12.2004
- (ISO 9126) ISO/IEC 9126, Information technology - Software product evaluation - Quality characteristics and guidelines for their use, 1991[E]
- (ISO 9241-11) SFS-EN ISO 9241-11 Näyttöpäätteillä tehtävän toimistotyön ergonomiset vaatimukset. Osa 11: Käytettävyyden määrittely ja arviointi, 1998
- (ISO 13407) ISO 13407, Human-centered design processes for interactive systems, 1999[E]

- (Kjeldskov et al, 2004) Kjeldskov J, Skov M. B., Stage J., 2004, Instant Data Analysis: Conducting Usability Evaluations in a Day, NordiCHI '04, October 23-27
- (Microsoft, 1995) Microsoft Corporation, 1995, The Windows interface guidelines for software design, Microsoft Press
- (Miller, 2000) Miller R. C, Myers B. A., 2000, Integrating a Command Shell Into Web Browser, the USENIX Association
- (Nielsen, 1993) Nielsen J., 1993, Usability Engineering, Academic Press
- (Nieminen, 1996) Nieminen M. P., 1996, Designing user interface concepts for multimedia services, Master's thesis, Teknillinen korkeakoulu
- (Riihiaho, 2000) Riihiaho S., 2000, Experiences with Usability Evaluation Methods, Licentiate's thesis, Teknillinen korkeakoulu
- (Seffah et al, 2001) Seffah A., Djouab R, Antunes H, 2001, Comparing a Recinciling Usability-Centered and Use Case-Driven Requirements Engineering Processes, IEEE Computer Society
- (Shneiderman, 1998) Shneiderman B., 1998, Designing the User Interface: Strategies for Effective Human-Computer Interaction, Addison-Wesley
- (Sinkkonen et al, 2002) Sinkkonen I., Kuoppala H., Parkkinen J., Vastamäki R., 2002, Käytettävyyden psykologia, IT Press
- (UsabilityNet, 2004) UsabilityNet - Contextual Inquiry, <http://www.usabilitynet.org/tools/contextualinquiry.htm>, viitattu 7.12.2004
- (Usor, 2004) Usor - A Collection of User Oriented Methods - Contextual Inquiry, <http://sunrise.nada.kth.se/usor/jml.cgi/Methods/context.jml?graphics=true>, viitattu 7.12.2004
- (Winograd, 1996) Terry Winograd, 1996, Bringing Design to Software, Addison-Wesley, lukujen tiivistelmät osoitteessa <http://hci.stanford.edu/bds/14-p-patric.html>, viitattu 10.12.2004

# Liitteet

- Liite A: Ensimmäisen osion aikana lähetetty sähköpostikysely

# Liite A

## Ensimmäisen osion aikana lähetetty sähköpostikysely

Subject: questionnaire about the simulation program

We are making new version for simulation user interface. To make the new version well usable for you we need your opinions about the current version. We would also like to hear your insight how the new version should work. Please modify this email and send it back to us.

Answering the question should not take more than five minutes.

You can answer to multiple-choice questions by adding few X-marks in front of your choice or by bolding your choice.

How often you use the program?

- Daily
- Once a week
- Few times a month
- Once a month
- More seldom
- I have never used it

How easy was the installation of the program?

- X (I didn't install it)
- 1 (easy)
- 2
- 3
- 4
- 5 (hard)

How easy is it to use the program?

- 1 (easy)
- 2
- 3
- 4
- 5 (hard)

How often you change the settings in the program when you start using it?

- Every time I use it
- Almost every time
- Now and then
- Seldom
- Never

It is easy to know what the setting mean in the program and how they change the behavior of the system?

- 1 (yes, I know every of them)
- 2
- 3
- 4
- 5 (no, I don't know what they mean)

When you press start-button the program open terminals. How often you check the opening terminals?

- Always
- Almost every time
- Now and then
- Seldom
- Never

How often you check the log files when running the program?

- Always
- Almost every time
- Now and then
- Seldom
- Never

There are now two test scenarios (Helsinki and Manhattan). How many test scenarios would be best for you? What parameters would you like to change (like condition etc)?

(write here)

How many mobiles would you like to connect and test with the program?

(write here)

The map in the program is clear?

- 1 (excellent or very clear)
- 2
- 3
- 4
- 5 (bad or confusing)

I use following features in the map,



- zoom
- showing or hiding some details
- time scale
- stop, pause, rewind, forward, play
- what else (write here)

Would you like to start and change parameters of the program with Windows or Mac? System will however run on Linux.

- 1 (Yes, it would be great)
- 2
- 3 (All the same)
- 4
- 5 (No, Linux version is better)

There is now downlink C/I chart in the map. What kind of charts would you like to see in the map?

(write here)

If you are using the log files or opening terminals what you are checking from them?

(write here)

What kind of problems you experience while using current version of the program?

(write here)

What kind of problems you experience with the current version of the map in the program?

(write here)

What features there are missing in the program now or what features you would like to include in the next version?

(write here)

What else comes into your mind when speaking about the program?

(write here)

Are you interested to see and test alpha or beta versions of the graphical user interface? This way you will get better possibility to say how the new version should work and what features there should be. Alfa version is expected to be ready on June and first beta on August.

Yes/No

Place for free comments:

(write here)

Thank you for taking the time to answer. Our aim is to have new and improved version of the program user interface ready on November 2004.

Petteri Suontama