



**TAMPEREEN  
AMMATTIKORKEAKOULU**

**OPINNÄYTETYÖ**

**WWW-POHJAISEN ASIAKASREKISTERIN  
TOTEUTUSSUUNNITELMA  
Case: eTupa**

**Anna-Kaisa Hannula**

Tietojenkäsittelyn koulutusohjelma  
toukokuu 2007  
Työn ohjaaja: Petri Heliniemi

**TAMPERE 2007**



---

Tekijä(t)	Anna-Kaisa Hannula	
Koulutusohjelma(t)	Tietojenkäsittely	
Opinnäytetyön nimi	WWW-pohjaisen asiakasrekisterin toteutussuunnitelma, Case: eTupa	
Työn valmistumis- kuukausi ja -vuosi	toukokuu 2007	
Työn ohjaaja	Petri Heliniemi	Sivumäärä: 47

---

## TIIVISTELMÄ

Opinnäytetyön tavoitteena oli suunnitella nykyaikainen ja helppokäyttöinen WWW-pohjainen asiakasrekisteri sekä sen ulkoasu ja lomakepohjat.

Yrityksellä oli käytössään vanha asiakasrekisteri Microsoft Accessissa, joka ei enää palvellut kasvavan yrityksen tarpeita. Asiakasmäärän kasvaessa haluttiin kehittää sovellus, jonka avulla erilaisia käsin tehtäviä toimistotöitä voidaan keskittää tehtäväksi yhdellä sovelluksella. Kun kaikki asiakkaisiin liittyvät toimistotyöt saadaan yhteen sovellukseen, päivittäinen toimistotyöhön keskittyvä työaika vähenee.

Työn toimeksiantajana oli eTupa, joka on perustettu vuonna 2005 palvelemaan asiakkaita erilaisissa tietoteknisissä ongelmissa. Tämän työn perusteella laadittu suunnitelma uudesta asiakasrekisteristä tullaan toiminnallisesti toteuttamaan kesän 2007 aikana.

Lähdin kehittämään suunnitelmaa oman osaamiseni ja erilaisten kirjallisten lähteiden perusteella. WWW-sivujen käytettävyydestä ja rakentamisesta löytyy runsas määrä kirjallisuutta. Kirjallisuuden lisäksi tutustuin erilaisiin Internet-sivuihin, joiden avulla löytyy kirjallisuutta ajankohtaisempaa tietoa.

Opinnäytetyön tuloksena saavutettiin helppokäyttöinen pohja tulevalle asiakasrekisterille. Lukuisten yrityksen sisäisten palaverien perusteella jokainen yrityksen työntekijä sai esittää mielipiteensä asiakasrekisterin ominaisuuksista. Näiden palaverien perusteella kehitin sivupohjan, josta löytyvät ne ominaisuudet, jotka todettiin tärkeimmiksi.



Author(s)	Anna-Kaisa Hannula	
Degree Programme(s)	Business Information Systems	
Title	Designing plan for a customers managing website. Case: eTupa	
Month and year	May 2007	
Supervisor	Petri Heliniemi	Pages: 47

---

## ABSTRACT

The purpose of this thesis was to design a modern and easy-to-use website to manage customers. In this thesis I will design the look of the website and forms. This customers managing website will be implemented in summer 2007.

Company had an old customers managing program which didn't work properly anymore. When the amount of customers grew, company wanted to design an application whereby different kind of office work can be centralized in one application. By implementing the new managing program daily office work will be reduced.

I made this website design to eTupa. eTupa was established in 2005. eTupa helps its customers in all kinds of data technical problems.

In this thesis there has been used information from books and the Internet. Usability of websites is popular subject so I found a lot of literature. From webpages I got more fresh information than I got from books.

On the basis of this thesis, eTupa got easy-to-use managing program. With input from all employees from eTupa we designed a good base to the new customers managing program.

# Sisällysluettelo

<b>Käsitteet ja lyhenteet .....</b>	<b>6</b>
<b>1 Johdanto .....</b>	<b>7</b>
<b>2 Yrityksen esittely.....</b>	<b>9</b>
<b>3 Käytettävyys.....</b>	<b>11</b>
3.1 Mitä on käytettävyys?.....	11
3.2 Käytettävyyden merkitys yritykselle.....	11
3.3 Ihminen käyttäjänä .....	12
3.3.1 Hahmolait.....	12
3.3.2 Käyttäjän muisti .....	13
3.3.3 Sovelluksen antama palaute .....	13
3.4 Visuaalinen suunnittelu .....	14
3.4.1 Sommittelu.....	14
3.4.2 Sivuston ulkoasu.....	14
3.4.3 Typografia.....	15
<b>4 Sivunkuvauskielet .....</b>	<b>16</b>
4.1 HTML.....	16
4.1.1 HTML-dokumentin rakenne .....	17
4.2 XHTML.....	17
4.2.1 XHTML-dokumentin rakenne .....	17
4.2.2 Strict ja Transitional-määrittelyjen eroja.....	18
4.2.3 XHTML:n edut.....	19
<b>5 PHP ja MySQL .....</b>	<b>20</b>
5.1 PHP.....	20
5.2 MySQL-tietokanta.....	21
<b>6 Tyyllisäännöt .....</b>	<b>22</b>
<b>7 Lomakkeet WWW-sovelluksessa.....</b>	<b>25</b>
7.1 Lomakkeen rakenne .....	25

7.2 Lomakkeen ulkoasu.....	26
<b>8 Nykyisen asiakasrekisterin arviointi .....</b>	<b>27</b>
8.1 Visuaalinen ilme.....	27
8.2 Rakenne .....	27
8.2.1 Lomake-kentät.....	28
8.3 Yhteenveto .....	30
<b>9 Uuden sovelluksen tuotanto .....</b>	<b>31</b>
9.1 Tarvemäärittely .....	31
9.1.1 Tietoturva.....	31
9.1.2 Lomakekentät.....	32
9.1.3 Asiakashaku .....	32
9.1.4 Uuden asiakkaan lisääminen .....	33
9.1.5 Raportit .....	33
9.1.6 Asetukset.....	33
9.1.7 Reskontra-toiminnot .....	34
9.1.8 Helppokäyttöisyys ja käytännöllisyys .....	34
9.2 Järjestelmän toteutus .....	35
9.2.1 Rakenne .....	35
9.2.2 Visuaalinen ulkoasu.....	36
9.2.3 Kuvat.....	36
9.2.4 Navigaatio .....	36
9.2.5 Typografia.....	36
9.2.6 Sivuston osat .....	37
<b>10 Yhteenveto.....</b>	<b>44</b>
<b>11 Lähteet.....</b>	<b>46</b>

## Käsitteet ja lyhenteet

CSS	Cascading Style Sheets CSS-tyyliohje antaa suosituksen sivuston ulkonäölle.
HTML	HyperText Markup Language Kieli, jonka avulla voidaan rakentaa WWW-sivustoja.
PHP	Hypertext Preprocessor Käytetään yleisesti luotaessa dynaamisia WWW-sivustoja.
XHTML	Extensible HyperText Markup Language XHTML korvaa HTML:n. XHTML perustuu XML- kieleen.
XML	Extensible Markup Language XML:n avulla luodaan merkkäuskieliä.
Validointi	Dokumentin kieliasun tarkistaminen määritysten mukaiseksi.
WWW	World Wide Web

# 1 Johdanto

## Tausta

Yrityksen toiminnan laajentuessa ja asiakaskunnan kasvaessa, eTuvassa todettiin, ettei vanha asiakasrekisteri enää palvele yrityksen tarpeita. Vanha asiakasrekisteri oli vaikeakäyttöinen ja sen käyttäminen vei työaikaa. Tämän ongelman pohjalta yrityksessä alettiin miettiä uuden ja mahdollisesti opinnäytetyönä toteutettavan asiakasrekisterin toteuttamista.

Vanha asiakasrekisteri rakennettiin yritystoiminnan alkaessa Microsoft Accessiin. Accessin omat muokkaus- ja ulkonäkömahdollisuudet olivat rajalliset. Asiakasrekisterin pohjan muokkaaminen oli erittäin hankalaa, eikä henkilökunnalta löytynyt motivaatiota ryhtyä opettelemaan Accessin tarkempia ominaisuuksia, jotta vanhasta asiakasrekisteristä olisi saatu toimintaa palveleva.

Suoritin työharjoitteluni keväällä ja kesällä 2006 eTuvassa, jonka aikana sain toimeksiannon, jonka tavoitteena on suunnitella uusi asiakasrekisteri. Asiakasrekisteristä haluttiin WWW-pohjainen ja siitä tulisi löytyä myös muita ominaisuuksia, kuin vain asiakkaiden perustietojen hallitseminen. Muina ominaisuuksina haluttiin esimerkiksi huoltoselosteiden, käteiskuittien ja laskujen tulostamista suoraan sovelluksesta. Vanha asiakasrekisteri ei antanut mahdollisuutta edellä mainittujen ominaisuuksien olemassaoloon.

Uusi sovellus siis helpottaa yrityksen arkea huomattavasti, kun kaikki tiedot asiakkaista löytyvät samasta paikasta. Tämän myötä myös virheet vähenevät, kun kirjauksia ei tarvitse tehdä moneen eri paikkaan.

## Tavoitteet

Tämän raportin tavoitteena on kertoa lukijalle, että mitä helppokäyttöisen WWW-sivuston suunnittelussa tulee ottaa huomioon. Koska sivustoa tullaan käyttämään joka päivä, on sen oltava juuri yrityksen omia tarpeita palveleva ja erittäin helppokäyttöinen. Olen siis tutustunut erilaisiin käytettävyyttä käsitteleviin teoksiin ja niiden pohjalta rakentanut yritykselle sopivan ratkaisun.

## Lähdeaineisto

Käytin opinnäytetyön perusteena useita XHTML- ja WWW-suunnittelua koskevia teoksia. Teoksien tekijöinä on alan parhaimmistoa, joten heidän tietämyksensä WWW-suunnittelusta on huippuluokkaa. Yritin löytää mahdollisimman tuoretta tietoutta ja luotettavista lähteistä.

Internet on täynnä vinkkejä ja ohjeita kunnollisen WWW-sivuston tekemiseen, kaikkiin Internet-lähteisiin en kuitenkaan halunnut luottaa. Internetistä löydetyistä lähteistä pidin luotettavana esimerkiksi W3 Schoolsin sivustoa, joka tarjosi hienoja vinkkejä ja ohjeita. Internetin ehdoton etu on tietojen nopea päivittyminen, Internetin avulla voi löytää uusimman tiedon nopeammin, kuin varsinaista kirjallisuutta lukemalla.



---

## 2 Yrityksen esittely

eTupa on perustettu vuonna 2005 palvelemaan asiakkaita erilaisissa tietoteknisissä ongelmissa. eTuvassa on lisäksi kaksi Tampereen kaupungin tukemaa maksutonta nettipistettä, jotka ovat asiakkaiden vapaassa käytössä. eTuvasta asiakas saa apua niin tietoteknisten laitteiden kuin digisovittimenkin hankintaan ja käyttöön. eTuvan asiakaskunta koostuu pääasiassa kotitalouksista Pirkanmaalla.

Idea eTuvan perustamisesta syntyi jo vuonna 2001, kun eTampere-tietoyhteiskuntaohjelma kutsui tamperelaisia atk-kouluttajia koolle. Kokoontumisen tuloksena syntyi kouluttajaverkosto, joka koostuu tamperelaisista yrityksistä, julkisista koulutusorganisaatioista sekä kolmannen sektorin atk-kouluttajista. Kouluttajaverkoston palaverissa tuotiin esiin tarve, jonka mukaan Tampereelle tulisi perustaa yksityishenkilöitä palveleva neuvontapiste. Idealle ei kuitenkaan löytynyt heti toteuttajaa. Lopulta vuonna 2004 Tampereen ammattikorkeakoulun opiskelija ja eTampere-toimistolla työskennellyt Emma Littunen teki päätöksen ryhtyä eTuvan yrittäjäksi. Emma Littunen teki eTuvan perustamisesta opinnäytetyön Tampereen ammattikorkeakoulun liiketalouden linjalle vuonna 2005. (Littunen 2005: 10-11.)

eTupa työllistää tällä hetkellä viisi henkilöä. Henkilökunnan on tärkeää omata hyvät asiakaspalvelutaidot ja olla tietoteknisesti taitavaa, jotta asiakkaalle jää hyvä mielikuva eTuvan kyvystä hoitaa asiakkaan tietoteknisiä laitteita.

Asiakkaalla on mahdollisuus tuoda tietokone tai muu tietotekninen laite toimistolle huollettavaksi tai vaihtoehtoisesti eTuvan henkilökunta tekee asiakkaan luokse kotikäynnin. Toimistolla tehdään pidempiaikaisia huoltotoimenpiteitä, kuten käyttöjärjestelmän uudelleenasetuksia ja kotikäynneillä esimerkiksi WLAN-verkon asennuksia. Kotikäynnit ja muut huoltotoimenpiteet ovat maksullista palvelua, toimistolla tapahtuva tietotekninen perusneuvonta on maksutonta.

Koska elämme vuotta 2007, digisovittimien myynti on lähtenyt nousuun. Analogiset tv-lähetykset loppuvat elokuussa 2007, joten ihmiset ovat päivä päivältä kiinnostuneempia hankkimaan digisovittimen. Digi-aika herättää kansalaisissa paljon kysymyksiä ja niihin eTupa vastaa mielellään. Digisovittimien myynnin lisäksi eTuvasta saa maksullisia kanavakortteja kaapeliin, antenniin ja satelliittiin. Asiakas voi siis hankkia koko digipaketinsä suoraan eTuvasta ja näin siirtyä helposti digilähetysten aikakauteen.

eTupa tekee tiivistä koulutusyhteistyötä Tampereen kaupungin kanssa. Yhteistyön piiriin kuuluvat Nettikinkerit, sähköinen koulutusopas ja kouluttajaverkosto.

Nettikinkerit ([www.etupa.fi/nettikinkerit](http://www.etupa.fi/nettikinkerit)) ovat kansalaisille ilmaisia koulutusiltoja, joissa neuvotaan esimerkiksi oman tietokoneen ostossa tai Internet-liittymän hankinnassa. Sähköiseen koulutusoppaaseen ([www.etupa.fi/koulutusopas](http://www.etupa.fi/koulutusopas)) atk-koulutusta järjestävät tahot voivat ilmoittaa omat koulutuksensa. Sähköisestä koulutusoppaasta käyttäjä voi hakea erilaisilla hakuehdoilla omiin tarpeisiinsa sopivia atk-koulutuksia.

#### Yhteystiedot

eTupa  
Itsenäisyydenkatu 25  
33500 TAMPERE  
puhelin (03) 212 5050  
[www.etupa.fi](http://www.etupa.fi)  
sähköposti: [info@etupa.fi](mailto:info@etupa.fi)

---

## 3 Käytettävyys

Suurin osa maailman ihmistä törmää jokapäiväisessä elämässään käytettävyysongelmiin. Käytettävyys on laaja käsite, joka muokautuu myös ihmisen omista valinnoista ja ymmärryksestä eri asioita kohtaan. Sovellus on suunniteltava käyttäjää varten, ei suunnittelijaa varten. Ikävän usein sovelluksen todelliset hyödyt saadaan irti ainoastaan suunnittelijoiden keskuudessa.

### 3.1 Mitä on käytettävyys?

Käytettävyys luo vahvoja mielikuvia, kuitenkin mikään yksittäinen termi ei voi aukottomasti sitä selittää. Wiio (2004: 28.) määrittelee käytettävyyden sanoilla ymmärrettävä, vaivaton, kattava sekä esteettisesti miellyttävä.

Ymmärrettävän sovelluksen avulla käyttäjän on helppo päätellä, miten hän pääsee haluamaansa lopputulokseen. Vaivattoman sovelluksen avulla käyttäjä suoriutuu tehtävistään yksinkertaisella tavalla. Kattava sovellus tarjoaa puolestaan niitä toimintoja, joita käyttäjä tarvitsee hoitaakseen sen tarpeen, johon sovellus on tarkoitettu. Esteettinen sovellus viestittää käyttäjälle laadusta. (Wiio 2004: 28-31.)

### 3.2 Käytettävyyden merkitys yritykselle

Sovelluksien huono käytettävyys maksaa yritykselle työtunteina ja suorana rahana. Wiion (2004: 32-34) mukaan ongelmilla voi olla seuraavia vaikutuksia: käyttäjä ei hyödynnä kaikkia ominaisuuksia, käyttäjän työaika kuluu tarpeettoman kauan tai käyttäjä tekee virheitä.

Sovellusprojekteja saatetaan nopeuttaa tai niiden budjetti on rajallinen. Tämä taas karsii väistämättä sovelluksien ominaisuuksia tai niiden testausta. Kuitenkin panostuksella nimenomaan sovelluksen käytettävyyteen voitettaisiin lukuisia työtunteja ja madallettaisiin työntekijöiden kynnystä sovelluksen omaksumiseen (Wiio 2004: 34-35).

Käytettävyydellä on merkitystä myös fyysisen työturvallisuuden kannalta. Käytettävyysongelmat aiheuttavat turhautumista ja turhautuminen voi aiheuttaa virhealttiutta ja työtapaturmariskiä. (Kuutti 2003: 16.)

### 3.3 Ihminen käyttäjänä

Sovelluksia suunniteltaessa suunnittelijoiden on otettava huomioon ihmisten erilaisuus ja jopa sukupuoli. On tutkittu, että naiset ja miehet reagoivat erilaisilla sovelluksen ulkonäköön ja sen antamiin palautteisiin. Naiset voivat jopa kieltäytyä käyttämästä sovellusta, koska he eivät usko osaavansa. Sovellusten tulee olla niin käyttäjäystävällisiä, että käyttäjien ei tarvitse pelätä epäonnistumista. Aiemmat epäonnistumiset vastaavan tuotteen käytössä lisäävät pelkoa. (Sinkkonen, Kuoppala, Parkkinen ja Vastamäki 2002: 290.)

#### 3.3.1 Hahmolait

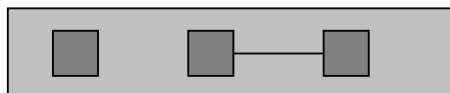
Käyttöliittymän suunnittelun kannalta oleellisimpia asioita ovat hahmolait. Hahmolait kertovat siitä, miten ihminen mieltää yhteenkuuluviksi näkemiään asioita. Lapsesta asti ihminen on ryhmitellyt asioita. Käyttöliittymä ei siis saa rikkoa luonnollisia hahmolakeja. Jos asiat kuuluvat yhteen, ne pitää ryhmitellä hahmolakien mukaan. (Kuutti 2003: 27.)

Erilaisia hahmolakeja ovat esimerkiksi läheisyys, samanlaisuus ja yhteenliittyminen. Läheisyyden hahmolaissa ihminen mieltää lähellä olevat kohteet ja yksityiskohdat yhteenkuuluviksi. Samanlaisuuden hahmolaissa ihminen mieltää samankaltaiset hahmot yhteenkuuluviksi. Hahmolakeja voidaan tuoda esiin erottelemalla yhteenkuuluvat asiat väreillä tai muodoilla. (Sinkkonen, Kuoppala, Parkkinen ja Vastamäki 2002: 102-103.)



*Kuva 1 Esimerkki läheisyyden hahmolaista*

Yksi voimakkaimmista hahmolajeista on yhteenliittyminen. Yhteenliittymisen hahmolaissa toisissaan kiinni olevat kohteet kuuluvat yhteen (Sinkkonen, Kuoppala, Parkkinen ja Vastamäki 2002: 104).



*Kuva 2 Esimerkki yhteenliittymisen hahmolaista*

---

### 3.3.2 Käyttäjän muisti

Ihmisen muisti voidaan jakaa kahteen osaan, lyhytkestoiseen ja pitkäkestoiseen muistiin. Lyhytkestoisen muistin kapasiteetti on pieni, yleensä viidestä yhdeksään asiaa. Asiat pysyvät lyhytkestoisessa muistissa vain muutamia sekunteja, jos niitä ei tarkoituksella pidetä muistissa, esimerkiksi toistamalla. Lyhytkestoista muistia ei saisi käyttöliittymäsuunnittelussa kuormittaa yli viidellä asialla. (Kuutti 2003: 53.)

Käyttöliittymäsuunnittelussa tulee ottaa huomioon tietokoneen valtavasti suuri muistikapasiteetti. Kaikki tarvittava tieto tulee pitää tietokoneen muistissa ja tieto esitetään käyttäjälle aina kun sitä tarvitaan. Päivämääriä kysyessä tulee aina esittää syötteen formaatti. (Kuutti 2003: 54.)

### 3.3.3 Sovelluksen antama palaute

Sovelluksen tulee antaa käyttäjälle kunnollista palautetta reaaliajassa. Esimerkiksi lomaketta täytettäessä tulisi kertoa virheellisestä arvosta heti kun se on annettu, eikä vasta silloin kun lomaketta lähetetään. Tämä on suuri ongelma WWW-pohjaisissa lomakkeissa. HTML-sivunkuvauskieli on tehty yksisuuntaiseen kommunikointiin, joten siinä ei ole keinoja tarkastaa lomaketta ennen kuin koko lomake lähetetään. (Kuutti 2003: 56-57.)

Mikäli joku sovelluksen tehtävä kestää yli 10 sekuntia, on suotavaa, että käyttäjälle kerrotaan jäljellä oleva odotusaika. Käyttäjä huomaa jo yhden sekunnin viiveen. Mikäli sovellus ei anna käyttäjälle mitään ohjetta odottamiseen, käyttäjä saattaa sulkea sovelluksen luullen sen kaatuneen. Jäljellä olevaa aikaa voidaan esittää esimerkiksi tilanpalkilla, joka on varmasti tuttu näky kaikille tietokoneen käyttäjille. (Kuutti 2003: 57-58.)

---

## 3.4 Visuaalinen suunnittelu

Yksi olennainen osa käytettävyyttä on visuaalinen suunnittelu. Käyttäjä joutuu katselemaan sovelluksen ulkonäköä koko ajan sovellusta käyttäessään. Visuaalisessa suunnittelussa on tärkeää olla johdonmukainen ja valittua linjaa tulee käyttää koko sovelluksessa. (Kuutti 2003: 90.)

### 3.4.1 Sommittelu

Sommittelulla tarkoitetaan sivuston osien järjestämistä rajattuun tilaan. Sivuston osia ovat esimerkiksi kuvat ja tekstit. Hyvin suunnittelulla sommittelulla voidaan herättää mielenkiintoa ja välittää käyttäjälle sivuston tunnelmaa. Sivustolla olevilla viivoilla ja väreillä voidaan rytmittää sivuston ulkoasua ja ohjata käyttäjän mielenkiintoa tiettyyn paikkaan. (Korpela ja Linjama 2005: 363.)

Mikäli sivustolta halutaan nostaa jokin asia esiin, kannattaa se tehdä hillitysti väreillä tai sisennyksillä. Visuaalisia ärsykeitä kannattaa kuitenkin käyttää harkiten ja hyvästä syystä (Kuutti 2003: 91). Käyttäjän katse kiinnittyy väreihin ja kontrasteihin, näin sommittelu ohjaa käyttäjän katsetta. Kohteen ympärille kannattaa myös jättää tyhjää tilaa, koska se erottaa kohteen ympäristöstä. (Korpela ja Linjama 2005: 364-365.)

### 3.4.2 Sivuston ulkoasu

WWW-sivujen ulkoasun suunnittelua voidaan pitää samankaltaisena, kuin television ja painotuotteiden suunnittelua. Suurin ero löytyy typografiasta. WWW-sivut muuntuvat selaimen, näytön ja käyttäjän toimenpiteiden vaikutuksesta. Muut mediat ovat WWW-sivuihin nähden vähemmän alttiita edellä mainituille vaikutuksille, eihän lehden fontin koko voi enää sen jälkeen vaihtua, kun lehti on painettu. (Korpela ja Linjama 2005: 356.)

Yhtenäistä ulkoasua tulee noudattaa jokaisella sivulla. Yhteinäinen ulkoasu luodaan helpoiten CSS-tyylitiedostojen avulla. Tärkeintä on, että sivuston jokaisella sivulla on samanlainen taustakuva sekä tekstisisältö ja kuvat ovat sijoitettu samaan paikkaan. Lisäksi on tärkeää käyttää samaa fonttia ja värejä. (Korpela ja Linjama 2005: 356.)

### 3.4.3 Typografia

Jarno Lukkarila (2001: 7) määrittelee typografian seuraavasti:  
*”Typografiaa voidaan sanoa puhutun kielen visuaaliseksi muokoksi. Se antaa tekstille ulkoasun, jolla voi puheen tapaan ilmaista sävyjä, painotuksia ja voimakkuuksia.”*

Sivustolle tulee valita yksi fonttilaji, jota käytetään läpi koko sivuston. Erilaiset fontit ja fonttikoot tuovat sivustolle levottoman vaikutelman. Leipätekstinä käytetään yhtä fonttikokoa ja otsikot määritellään oman tahdon mukaisesti hieman suuremmiksi tai muuten poikkeaviksi. Usein sivustolla käytetään kolmea erilaista fonttilajia: otsikoissa, leipäteksteissä ja kuvateksteissä. (Korpela ja Linjama 2005: 373-374.)

Fonttien käyttöä vaikeuttavat erilaiset selaimet. Sivuston ulkoasua ei kannata suunnitella pelkkien fonttien varaan, koska osa selaimista ei näytä suunniteltua fonttia ollenkaan. Tämän takia sivustolle kannattaa määritellä vaihtoehtoisia fontteja. Olemassa ei ole mitään standardia siitä, mitä fontteja käyttäjien koneista löytyy, koska tietokoneiden valmistajat eivät toimita kiinteitä fonttijoukkoja. Mikäli tarkastellaan kaikista Windows-koneista löytyviä fontteja, on lista lyhyt: Arial, Courier New ja Times New Roman. (Korpela ja Linjama 2005: 375-377.)

Fonttityyppejä ovat esimerkiksi päätteellinen (serif) ja päätteetön (sans-serif). Pääteellistä fonttia käytetään yleensä sanomalehdissä ja painetussa tekstissä. Pääteellisen fontin pienet koukerot helpottavat lukemista. Pääteellisillä fonteilla on taipumus muuttua sumeiksi, mikäli ne esitetään näytöllä liian pieninä. (Laak 2005.)

Päätteetön fontti, kuten Verdana, on välitykseltään avara ja helpolukuinen. Mikäli näytöllä on alhainen resoluutio, on suoravii-vainen fontti selkeämpi vaihtoehto. (Laak 2005.)

---

## 4 Sivunkuvauskielet

### 4.1 HTML

World Wide Webin (WWW) kehityksen yksi suurimmista peruskivistä on ollut HTML (*Hypertext Markup Language*). HTML luotiin 1990-luvun alussa CERNin tutkimuslaitoksella Sveitsissä. Kielen avulla pystytään muotoilemaan ja kuvailemaan tekstiä sekä esittämään kuvia ja ääntä. (Ek ja Eriksson 2001: 5, 7)

WWW:n alkuaikoina modeemien hitaus ja hankalasti muodostettavat yhteydet eivät houkuttelleet käyttäjiä. Kuitenkin 1990-luvun puolivälissä modeemien siirtonopeudet olivat kasvaneet siinä määrin, että WWW:n suosio räjähti nousuun. Käyttäjät kiinnostuivat WWW:n tuomista palveluista ja verkkoon tuotettiin tuhansia ja tuhansia uusia sivuja virhesietoisen ja helpon HTML-kielen avulla. (Linjama 2001: 20-21.)

HTML:n kehitystä ovat sekä hyvässä että pahassa, ohjanneet web-selaimet. Selaimet tulkitsevat koodia erilaisilla tavoilla, tavoilla joita selainten valmistajat pitävät tärkeinä. Lisäksi selainten valmistajat ovat urakoineet saadakseen selaimet ymmärtämään jopa virheellistä koodia. (Ek ja Eriksson 2001: 5, 7.) Selainten määrä kasvoi alkuaikoina hurjalla vauhdilla ja niihin lisättiin mitä erilaisimpia tunnisteita ja tekniikoita. Tästä johtuen selaimet näyttävät kukin hieman eri tavalla HTML-dokumentin sisällön. (Veen 2002, 142.)

HTML-dokumentteja voidaan tuottaa erilaisilla ohjelmilla, mikä on johtanut siihen, että käyttäjät luottavat ohjelman avulla syntyneen koodin oikeellisuuteen, eikä sitä ei vaivauduta tarkistamaan. Tosin ensimmäisissä HTML-versioissa ei ollut syntaksia, eli kielioppia, jota WWW-sivujen tekijä olisi voinut noudattaa. Nykyaikaiset standardoidut HTML-kieltä kuvaavat spesifikaatiot ovat W3C:n (World Wide Web Consortium) kehittämiä. Spesifikaatiot esitetään DTD:ssä, joka esittää sallitut rakenteet, elementit ja määreet. Mikäli käyttäjä tekee HTML-dokumentin tarkasti kieliopin mukaisesti, selaimet eivät voi tulkita sitä väärin. (Linjama 2001: 21, 23, Ek ja Eriksson 2001: 7.)



### 4.1.1 HTML-dokumentin rakenne

W3Schoolsin (2007) esimerkkiä mukaillen HTML-dokumentin rakenne näyttää seuraavalta:

```
<HTML>
<BODY>
<P>Hello World.</P>
</BODY>
</HTML>
```

Esimerkki tulostaa yksinkertaisen Hello World. -tekstin näytölle. Pienen HTML-dokumentin tekeminen ei kaikessa yksinkertaisuudessaan ole tämän vaikeampaa.

Linjaman (2001: 66) mukaan HTML-dokumentin yksi tyypillinen virhe on päättävän tagin puuttuminen

```
<P>Hello World.
```

tai että, tagi-parit on kirjoitettu ristiin:

```
<P><B>Hello World</P></B>
```

## 4.2 XHTML

Vuosien saatossa HTML-kieltä on kehitetty ja sen viimeisin versio on nimeltään XHTML (*eXtensible Hypertext Markup Language*), joka julkaistiin vuonna 2000. XHTML:n tarkoituksena on luoda puhtaampi ja vaativampi rakenne. XHTML-dokumentti sisältää perinteisen HTML-dokumentin komennot ilman muotoilumääreitä. (Ek ja Eriksson 2001: 7-9, Linjama 2001: 9) Lisäksi XHTML on kehitetty toimimaan yhdessä muiden web-standardien kanssa, kuten CSS, josta kerrotaan myöhemmin tässä työssä. (Zeldman 2003: 149).

### 4.2.1 XHTML-dokumentin rakenne

XHTML-dokumentin alussa määritellään dokumentin DTD, eli dokumentin tyyppi. DTD määrittelee dokumentin kieliopin, joka tarkistetaan sivujen validoinnin yhteydessä. Validointi tarkoittaa syntaksin oikeellisuuden tarkistusta. Suositeltuja DTD-määrittelyksiä ovat Transitional tai Strict ja ne merkitään !DOCTYPE-merkinnällä. Näistä kahdesta Transitional antaa enemmän vapauksia XHTML-dokumentteja luotaessa. (Ek ja Eriksson 2001: 7-9.)

Esimerkki XHTML-dokumentista mukailleen W3C:n (2007) esimerkkiä:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html>
<head>
<title>XHTML-dokumentti</title>
</head>
<body>
<p>Hello World!</p>
</body>
</html>
```

XHTML-kieli on case-sensitiivinen, eli kirjainten koko merkitsee. Kaikki XHTML-dokumentin kirjaimet tulee kirjoittaa pienillä kirjaimilla tai muuten kirjoitettu koodi ei ole kieliopin mukaista, eli validia. (Zeldman 2003: 159).

Tyypillinen HTML-dokumentin osa näyttäisi tältä:

```
<P>Hello World!</P>
```

XHTML-kielessä saman tulee näyttää seuraavalta:

```
<p>Hello World</p>
```

#### 4.2.2 Strict ja Transitional-määrittelyjen eroja

Zeldman (2003: 69, 71) löytää seuraavia eroja Strict ja Transitional määrittelyistä:

Strict:

- Tyylimäärityksien on oltava ulkoisessa tiedostossa.
- Ulkonäön tekemiseen käytetään vain CSS:ää.
- Taulukkoja saa käyttää vain niiden alkuperäiseen tarkoitukseensa, eli tiedon näyttämiseen taulukkomuodossa.
- Sivun rakenteen on oltava loogisesti nimetty, esimerkiksi header, menu, content, footer.
- Sivuston käyttäjäkunta käyttää uusia selaimia.

Transitional:

- CSS-tyylitiedosto määrittelee esimerkiksi dokumentin värit ja fontit.
- Taulukkotaitto on sallittua jos sillä saadaan parempi käytettävyys aikaiseksi.

- Suositellaan sivustoille, joiden käyttäjäkunta käyttää vanhempia selaimia.

### 4.2.3 XHTML:n edut

Zeldman (2003: 150) listaa kirjassaan tärkeimmät syyt siirtyä käyttämään XHTML:ää:

- XHTML korvaa HTML 4:n.
- XHTML aiheuttaa oikein tehtynä vähemmän ongelmia, kuin HTML.
- Uudet selaimet rakastavat XHTML:ää.
- XHTML toimii uusissa langattomissa laitteissa, kuten kämmenmikroissa ja matkapuhelimissa.
- XHTML on osa uusia WWW-standardeja yhdessä CSS:n kanssa.

---

## 5 PHP ja MySQL

### 5.1 PHP

PHP (*Hypertext Preprocessor*) on suoraan HTML-koodiin kirjoitettava ohjelmointikieli. PHP-koodi ajetaan palvelimella aina ennen, kuin sivu lähetetään selaimelle. Kun sivulla oleva koodi suoritetaan, näkyy selaimella ainoastaan ohjelman tulostus. Tämän takia käyttäjät eivät näe PHP-koodia katsoessaan sivun HTML-koodia. PHP-koodia ei pystytä käsittelemään suoraan tietokoneen levyltä, vaan se on suoritettava WWW-palvelimella. (Heinisuo 2003: 16-17.)

Esimerkki PHP:n käytöstä mukailien Heinisuo esimerkkiä, joka tulostaa näytölle tekstin Hello World (2003: 17):

```
<?php
print "Hello World";
?>
```

PHP-kielillä on mahdollista toteuttaa kaikki yleisimmät ohjelmointikielien rakenteet, kuten for- ja while-silmukat, if-lauseet sekä muuttujat ja funktiot. Koska PHP on oliopohjainen kieli, voidaan sovelluksen toteuttaa myös aliluokkina. PHP on myös laajennettavissa erilaisten ohjelmakirjastojen avulla. (Heinisuo 2003: 17.)

Koska harvalla normaalilla käyttäjällä on käytössään oma WWW-palvelin, ovat Internet-operaattorit rientäneet kansalaisten avuksi. Näin PHP-sovellukset voidaan helposti ajaa operaattorin palvelimella. Jos operaattori päivittää PHP:n uudempaan versioon, on sivujen tekijän oltava valppaana, jotta oma sovellus toimii päivityksen jälkeen. PHP:n ominaisuuksia kehitetään jatkuvasti ja vanhojen ominaisuuksien toiminta saattaa näin muuttua. (Heinisuo 2003: 19.)

PHP toimii WWW-palvelimessa, jollainen on esimerkiksi Apache. Apache on monipuolinen palvelinohjelmisto, jota käytetään pääasiassa Unix-käyttöjärjestelmissä, mutta se toimii myös Windowsissa. Mikäli Apache-palvelin asennetaan omalle tietokoneelle, on PHP-sovellusten ajaminen mahdollista myös omalla koneella, ilman tiedostojen siirtämistä operaattorin WWW-palvelimelle. (Heinisuo 2003: 20.)

---

## 5.2 MySQL-tietokanta

MySQL on monipuolinen relaatiotietokanta. MySQL:n avulla voidaan toteuttaa niin lomakepalveluita kuin sähköisen kaupan järjestelmiäkin. MySQL toimii sekä Windows- että Unix-pohjaisissa käyttöjärjestelmissä. MySQL:n on kehittänyt ruotsalainen MySQL Ab alun perin kyseisen konsultointiyrityksen omaan käyttöön. (Heinisuo 2003: 11, 36).

Koska MySQL-sovellukset eivät käsittele tietokantaa suoraan, vaan palvelinohjelmiston kautta, noudattaa MySQL palvelinarkkitehtuuria. MySQL-palvelimeen voidaan ottaa yhteys esimerkiksi edellä esitellyn PHP-kielen avulla. (Heinisuo 2003: 36.)

MySQL:n ehdoton etu on helppo asennettavuus, kevyt ylläpito ja pienet ylläpitokustannukset. Monet kalliit kaupalliset tietokantaohjelmistot vaativat täysipäiväistä huolenpitoa. Vaikka MySQL ei sisällä kaikkia ominaisuuksia, joita kaupalliset tietokannat pitää sisällään, löytyy MySQL:stä paljon sellaisia ominaisuuksia joiden takia se on erinomainen valinta useimpien WWW-palvelujen taustalle. Lisäksi MySQL:n on mahdollisuus ostaa kaupallisia tukipalveluja. (Heinisuo 2003: 36-37.)

## 6 Tyylisäännöt

Tyylisäännöstö on ehdotus dokumentin esitysasusta. Tyylisäännöstö kirjoitetaan tarkoitukseen kehitellyllä kielellä. Tyylisäännöstöjen käytetyin ja toistaiseksi lähes ainoa käytettävissä oleva on CSS (*Cascading Style Sheets*). (Korpela ja Linjama 2005: 300.)

Jotta sivuston tekijä voi tehdä nykystandardien mukaisen Internet-sivuston, on sivustolla käytettävä CSS-tyylitiedostoja. CSS on standardoitu tapa hallita esimerkiksi sivuston värejä, fontteja sekä niiden sijainteja ja kokoja. CSS-tyylitiedostojen ehdoton etu on niiden tuoma yksinkertaisuus sivustoa hallitessa. Kun kaikki sivuston ulkonäkömääritykset on kerätty yhteen dokumenttiin, voidaan sivuston ulkonäön muutokset päivittää helposti vain yhteen paikkaan. Puhtaalla CSS-layoutilla sivuston tekijä voi kirjoittaa XHTML-dokumenttiin ainoastaan sivuston rakenteen ja kirjoittaa sen ulkonäön määritykset CSS-tiedostoon. CSS nopeuttaa myös sivuston latausaikoja, kun pitkistä HTML-dokumenteista luovutaan. (Zeldman 2003: 211-212.)

CSS-tyylisäännöstö on kuitenkin vain ehdotus sivun esitysasusta. Selaimet näyttävät ehdotuksen erilalla tai voivat jättää sen jopa kokonaan näyttämättä. Korpelan ja Linjaman (2005: 310.) mukaan suurin osa tyylisäännöstöillä aiheutetuista ongelmista johtuu siitä, ettei selaimien erilaisia ominaisuuksia tiedetä tai niitä ei muisteta ottaa huomioon.

CSS-tyylisäännöt koostuvat säännöistä. Säännöt asettavat elementtien ominaisuuksille arvoja. Ominaisuuksia ovat esimerkiksi väri, koko tai fonttilaji. CSS-sääntö koostuu selektorista ja aaltosulkeissa olevista deklaraatioista. Deklaraatio koostuu ominaisuuden nimestä, kaksoispisteestä ja ominaisuuden arvosta. Kuvassa 3 oleva CSS-sääntö määrittelee kaikille <p> alkuisille kappaleille fontin väriksi sinisen. (Korpela ja Linjama 2005: 302.)

sääntö					
	deklaraatio				
selektori		ominaisuus	:	arvo	
p	{	color	:	blue	; }

Kuva 3 CSS-sääntö

Tyylejä voidaan myös ryhmitellä, mikäli tyyleillä on samoja ominaisuuksia. Tällainen ryhmä on esimerkiksi otsikkotasot, jotka merkitään h1, h2 ja niin edelleen. Esimerkissä tyyleille h1 ja h2 annetaan fontin väriksi sininen ja se lihavoidaan. (Linjama 2001: 514.)

```
h1, h2 {
color: blue;
font-weight: bold;
}
```

HTML-elementeille voidaan määritellä alaluokkia, jonka avulla yhdelle HTML-elementille saadaan useita erilaisia esitystapoja. Aliluokka määritetään antamalla alkuperäisen tyylin nimestä ja aliluokasta muodostuva yhdistelmänimi, Linjaman (2001: 517) mukaan esimerkiksi näin: tyyli.luokka, jossa tyyli tarkoittaa alkuperäisen tyylin nimeä ja luokka aliluokan nimeä. Sanat erotetaan pisteellä.

```
h1 {
color: blue;
font-weight: bold;
}

h1.punainen {
color: red;
}
```

Aliluokkaa ei kuitenkaan ole pakko yhdistää tiettyyn tyyliin, vaan tyyli voidaan tuoda esiin myös jonkun muun elementin yhteydessä (Linjama 2001: 518).

```
.punainen {
color: red;
}
```

CSS-tyyli kirjoitetaan yleensä ulkoiseen tiedostoon, joka linkitetään HTML-dokumenttiin. Tämä yksi tyylitiedosto voi toimia usean eri HTML-lomakkeen tyylin pohjana. (Linjama 2005: 507.) CSS-tyylitiedosto linkitetään HTML-dokumenttiin seuraavalla tavalla:

```
<link href="tyyli.css" rel="stylesheet" type="text/css" />
```

CSS-tyylejä voidaan kirjoittaa myös suoraan HTML-dokumenttiin, mutta silloin menetetään ulkoisten CSS-tiedostojen tuoma etu helposta päivitettävyydestä. Zeldman (2003: 228) kertoo kirjassaan syitä myös siihen, miksi CSS-tyylit on välillä kirjoitettu suoraan HTML-dokumenttiin: sivusto sisältää vain yhden sivun tai CSS-tyylimäärittäviä tarvitaan poikkeuksellisesti vain yhdelle tietylle sivulle. Sisäinen tyylitiedosto kirjoitetaan HTML-dokumentin head-osaan.

---

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html>
<head>
<style type="text/css">
<!--
p {
color: green;
font-weight: bold;
}
-->
</style>
<title>XHTML-dokumentti</title>
</head>
<body>
<p>Hello World!</p>
</body>
</html>
```



## 7 Lomakkeet WWW-sovelluksessa

Lomake antaa selaimelle ohjeen muodostaa käyttöliittymä, jonka kautta käyttäjä voi kirjoittaa tekstiä tai antaa muuta dataa. HTML-kielen lomakerakenne on alkeellinen ja rajoittunut, mutta toiminnaltaan yksinkertainen: se ottaa vastaan käyttäjän syötteitä ja lähettää ne eteenpäin. (Korpela ja Linjama 2005: 268.)

Sähköisistä lomakkeista on käyttäjilleen paljon hyötyä mikäli lomakkeet on osattu toteuttaa toimiviksi niin esteettisestä kuin toiminnallisesta näkökulmasta. Yhdeksi suureksi hyödyksi voidaan nähdä lomakkeiden sähköinen toiminta, lomakkeiden tuomia tietoja voidaan jatkokäsitellä helposti. Vuorovaikutusta käyttäjän kanssa lisää käyttäjän valintojen mukaan muuttuvat sivustot. (Mäntylä, Lahtonen ja Ekonoja: 2007.)

Sähköisistä lomakkeista voi tulla myös ongelmia, mikäli ne on toteutettu virheellisesti tai käyttäjän tiedot hukkuvat jostain syystä bittiavaruuteen. Lisäksi lomakkeiden käsittelyyn vaaditaan usein erillinen ohjelma, joka saattaa itse tehtynä olla monimutkainen tehtävä. Lomakkeen käsittelyssä tulee myös ottaa huomioon erikoistilanteet, esimerkiksi skandinaaviset merkit. (Mäntylä ym: 2007)

### 7.1 Lomakkeen rakenne

Lomakkeen sisälle on mahdollista rakentaa taulukoita tai muita rakenteita. Ennen jokaista syöttökenttää on hyvä olla selitys, joka kertoo kentän merkityksen. Lomakkeen sisään on mahdollista kirjoittaa myös pidempiä selityksiä, kuten täyttöohjeita. (Korpela ja Linjama 2005: 279)

Seuraavassa esitellään erilaisia lomakkeiden muotoja (Heinonen ja Lahtonen: 2004.)

```
<input type="text" />
```

Yksirivinen syöttökenttä, johon käyttäjä voi syöttää tekstiä. Kentän pituuden voi määrittellä.

```
<textarea />
```

Laajempi tekstikenttä, johon käyttäjän on mahdollista kirjoittaa tekstiä usealle riville.

```
<input type="password" />
```

Salasanakenttään käyttäjä voi kirjoittaa tekstin, joka näytetään tähtimerkkeinä.

```
<input type="checkbox" />
```

Valintaruutu on neliön muotoinen kenttä, joka voidaan valita joko päälle tai pois. Valintaruutuja voidaan valita samasta ryhmästä monta.

```
<input type="radio" />
```

Radiopainike on ympyrän muotoinen kenttä, josta käyttäjä voi valita vain yhden samasta ryhmästä.

```
<input type="submit" />
```

Lähetyspainikkeen avulla lähetetään lomakkeen tiedot eteenpäin.

## 7.2 Lomakkeen ulkoasu

Lomakkeiden ulkoasuun voidaan vaikuttaa CSS-tyylimäärittelyillä. On kuitenkin suotavaa ottaa huomioon, että lomakkeen kenttiä ei kannata tyylitellä kovin voimakkaasti. Käyttäjät ovat tottuneet normaalin näköisiin lomakkeisiin, joten lomakkeet jotka ovat ratkaisevasti erilaisia, voivat sekoittaa. Huonoimmassa tapauksessa käyttäjä ei edes enää tunnista lomaketta lomakkeeksi. (Korpela ja Linjama 2005: 278, 292.)

Lomakkeiden toiminnallisuuden kannalta on aiheellista tehdä lomakkeesta kokonaisuutena helpommin havaittava ja hahmotettava. Mikäli sivulla on myös muuta sisältöä, voidaan lomakkeen alku kertoa esimerkiksi erilaisen taustavärin kautta. (Korpela ja Linjama 2005: 292.)

## 8 Nykyisen asiakasrekisterin arviointi

Tässä luvussa arvioidaan eTuvan nykyisen asiakasrekisterin hyviä ja huonoja puolia sekä tehdään parannusehdotuksia.

### 8.1 Visuaalinen ilme

Asiakasrekisterin ulkoasu perustuu Microsoft Accessin omiin tyyliin, joten ulkoasuun ei ole päästy vaikuttamaan kovinkaan paljon. Tyylistä on kuitenkin pyritty tekemään mahdollisimman selkeä ja osittain asiakasrekisterin värimaailma vastaakin eTuvan keltaisen ja sinisen johtamaa värimaailmaa.

Asiakasrekisteri esitetään kapealla kaistaleella keskellä näyttöä, joten käyttäjän on vieritettävä ruutua pystysuorassa saadakseen kaikki tiedot esiin. Useat käyttäjät ovat tottuneet käyttämään hiiren rullauspyörää siirtyäkseen näytöllä pystysuunnassa. Tässä tapauksessa hiiren rullauspyörän käyttäminen tuo esiin seuraavan asiakkaan. Vaihtumista ei välttämättä edes huomaa ja näin käyttäjä ryhtyy täyttämään väärän asiakkaan tietoja. Mikäli lomakeruutu olisi leveämpi, mahtuisi suurin osa täytettävistä tiedoista yhdelle ruudulle ja näin välttyttäisiin turhalta ruudun vierittämiseltä.

### 8.2 Rakenne

Asiakasrekisterin lomakepohjaan syötetään jokaisen asiakkaan tiedot nimestä ja osoitteesta lähtien. Lomakepohjassa on kuitenkin heikkouksia. Lomakepohja on tehty toimintaa aloitettaessa, joten kaikkia kohtia lomakkeesta ei enää tarvitse täyttää ja ne häiritsevät kokonaisuutta.

Kenttien yhteydessä ei mainita, miten kenttä tulisi täyttää. Esimerkiksi päivämäärän kohdalla syöttömuotoja on yhtä paljon kuin käyttäjiä. Päivämäärä tulee merkitä muodossa *kuukausi-päivä-vuosi*. Suomalaiseen kulttuuriin kuuluu merkitä päivämäärä muodossa *päivä.kuukausi.vuosi*, joten maininta kentän syöttömuodosta olisi paikallaan.

**Perustiedot**

<b>Número</b> (Laskuri)	<b>Etunimi</b>	<b>Sukunimi</b>	<b>Firma</b>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<b>Osoite</b>		<b>Postinumero</b>	<b>Kaupunginosa</b>
<input type="text"/>		<input type="text"/>	<input type="text"/>
<b>Puhelinnumero</b>		<b>Sijainti</b>	<b>Sähköposti:</b>
<input type="text"/>		<input type="text"/>	<input type="text"/>
<b>Ikä</b>	<b>Ikäkoodi</b>	<b>Osaaminen</b>	<b>Paljonko olisit valmis maksamaan vastaavanlaisesta palvelusta?</b>
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text"/>	<input type="text"/>
<b>Erityistä:</b>			<b>Käyntejä (kpl)</b>
<input type="text"/>			<input type="text" value="1"/>
			<b>Matka (km)</b>
			<input type="text"/>
<b>Asiakkaan kommentit kotikäynnistä</b>			
<input type="text"/>			
<b>Mistä saanut tiedon eTuvasta?</b>			
<input type="text"/>			

*Kuva 4 Perustieto-kenttä*

### 8.2.1 Lomake-kentät

#### Perustieto-kenttä

Perustieto-kentät ovat hyvin laadittuja muutamia poikkeuksia lukuun ottamatta. Suurin puute on kaupunki, jossa asiakas asuu. Hyvin usein se on Tampere, mutta asiakkaita löytyy myös naapurikunnista. Puute haittaa erityisesti yhteenvetojen tekemistä postitusta varten.

Vanhaan asiakasrekisteriin tulee käsin lisätä käyntien lukumäärä. Ikävän usein käyntien määrä on väärin suhteessa käyntien todelliseen lukumäärään, joten kohta jossa kerrotaan käyntien lukumäärä, tulee saada automaattiseksi tai toteuttaa toisella tavalla.

## Laitteistotiedot-kenttä



**Laitteistotiedot**

Käyttöjärjestelmä  Virusohjelmisto  Palomuri

Laitteisto  Tulostin  Skanneri

Internet  Mistä hankkinut koneet

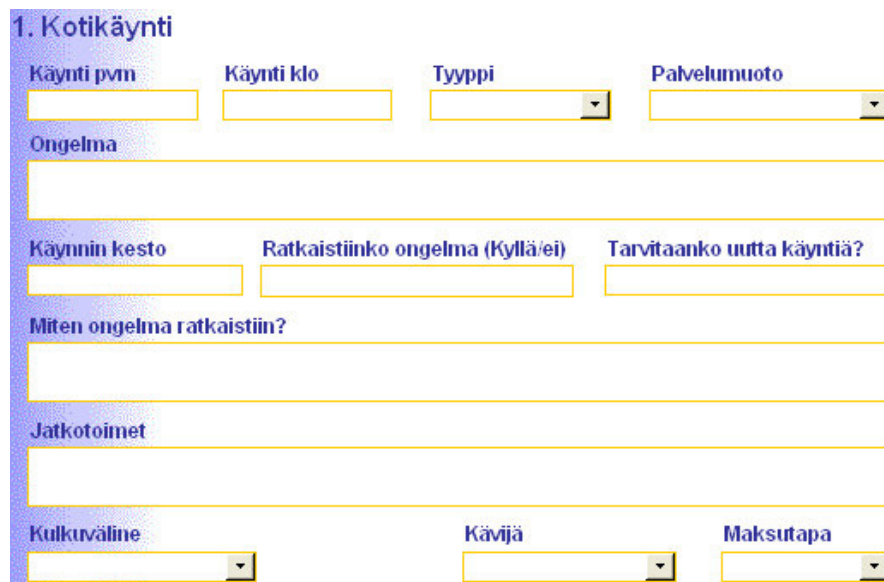
Selain

*Kuva 5 Laitteistotiedot-kenttä*

Ikävän usein laitteistotiedot jäävät täyttämättä. Tarkan mallin tietäminen helpottaisi jatkossa asiakkaan kanssa asiointia. Laitteistotietojen täyttämisen voisi tehdä pakolliseksi, jotta tiedot tulisi katsottua ja täytettyä asiakasrekisteriin. Tosin täyttämisen pakollisuus saattaa tuottaa myös ongelmia, jos tietoja ei ole muistettua katsoa.

## Kotikäynti-kenttä

Kentän otsikko on virheellinen. Kenttiin kirjataan myös toimistolla tapahtuvat työt.



**1. Kotikäynti**

Käynti pvm  Käynti klo  Tyyppi  Palvelumuoto

Ongelma

Käynnin kesto  Ratkaistiinko ongelma (Kyllä/ei)  Tarvitaanko uutta käyntiä?

Miten ongelma ratkaistiin?

Jatkotoimet

Kulkuväline  Kävijä  Maksutapa

*Kuva 6 Kotikäynti-kenttä*

Heti ensimmäinen kohta käynnin päivämäärä tuottaa työntekijöille ongelmia, koska jokaisella täyttökerralla käyttäjä joutuu miettimään kuinka kenttä täytetään oikein. Missään ei lue, että päivämäärä tulee syöttää muodossa *kuukausi-päivä-vuosi*. Harva suomalainen täyttää päivämääräkenttiä kyseisessä järjestyksessä, höystettynä vielä väliviivalla.

Kellonaikakin tuottaa käyttäjälle kysymyksiä: laitetaanko väliin piste? Riittääkö vain 13? Vai pitääkö muodon olla 13.00? Käynnin kesto tuo saman ongelmakohdan esiin. Käyttäjä ei saa mistään selvälle pitääkö kentässä eritellä puolet tunnit pisteellä vai pilkulla, vai merkitäänkö kenttään vain täysiä tunteja.

### 8.3 Yhteenveto

Asiakasrekisterissä on selviä puutteita ja turhia kenttiä, jotka hidastavat täyttämistä ja aiheuttavat epäselvyyksiä. Asiakasrekisteristä tehtävät yhteenvedot eivät ole totuuden mukaisia, koska kenttiä on täytetty väärin tai niitä ei ole täytetty ollenkaan.

Suurin puute vanhassa asiakasrekisterissä on se, ettei se anna kuin alle kymmenen merkintämahdollisuutta koti- tai toimistokäynneille, tämän jälkeen käynnit joudutaan merkitsemään tyhjiin mahdollisiin kenttiin. Kyseiset käynnit eivät täten näy missään, koska ne ovat upotettu niille kuulumattomille paikoille. Uuden asiakassovelluksen yksi tärkeimmistä ominaisuuksista tulee olla sen helppo muokattavuus. Kenttien otsikoita tulee voida vaihtaa helposti ja niiden pitää olla kenttää kuvaavia. Lisäksi jokaisesta käynnistä tulee olla merkintä niitä vastaavilla paikoilla, vastaavaa tilan loppumista ei saa uudessa asiakasrekisterissä olla.

Asiakasrekisteriä on ollut mahdollisuus päivittää. Microsoft Accessin käyttö on kuitenkin erittäin hankalaa ja monimutkaista, joten päivitykseen ei ole kukaan halunnut lähteä, koska se vie huomattavasti työaikaa. On ollut helpompaa tyytyä vanhaan ratkaisuun ja odotella uuden asiakasrekisterin valmistumista. Yrityksessä ollaan erittäin innostuneita mahdollisuudesta rakentaa oma ratkaisu, jossa jokainen yksityiskohta on mietitty tarkkaan. Turhia ominaisuuksia ei haluta. Jos jokin ratkaisu todetaan käytännössä puuttuvaksi, voidaan se rakentaa jälkikäteen.

## 9 Uuden sovelluksen tuotanto

Tässä luvussa tutustutaan tarkemmin eTuvan uuden asiakasrekisterin tarvemäärittelyyn ja käyttöliittymän lopulliseen toteutukseen. Sivuston tuotanto on jaettu kahteen pääosaan: käyttöliittymän tarvemäärittelyyn ja toteutukseen.

Omaan työnkuvaani ei kuulunut sivuston varsinaista toiminnallista toteuttamista, vaan sen hoitaa eTuvan toinen työntekijä kevään ja kesän 2007 aikana.

### 9.1 Tarvemäärittely

Uuden sovelluksen tarvemäärittely laadittiin yhteisissä palaverissa, joihin osallistuivat kaikki eTuvan työntekijät. Palaverissa jokainen sai esittää omat mielipiteensä uutta sovellusta koskien. Jokaisella työntekijällä on kokemusta eTuvan rutiineista, joten kaikki esille tulleet ideat ja parannusehdotukset otettiin huomioon.

Uuden järjestelmän perustarve on asiakkaiden tietojen helppo hallitseminen. Kaikki asiakkaan tiedot halutaan löytää helposti yhdestä paikasta, aikaisemmin harrastetun mappien selauksen sijaan. Jos ennen käyttäjä on joutunut tekemään asiakkaan koneen huollon yhteydessä huoltoselosteen erikseen ja tämän jälkeen vielä kirjoittamaan tiedot asiakasrekisteriin, saa käyttäjä nyt uudesta sovelluksesta tulostettua suoraan huoltoselosteen.

Sen jälkeen, kun asiakas on syötetty järjestelmään, kaikki käyttäjät pääsevät katsomaan asiakkaan tapahtumia. Mikäli asiakas kysyy myöhemmässä vaiheessa apua esimerkiksi puhelimella, ajan tasalla olevasta sovelluksesta on helppo saada tukea puhelulle ja näin asiakkaan neuvominen helpottuu. Nopean neuvonnan ehtona on myös sovelluksen nopea toiminta. Käyttäjällä pitää olla mahdollisuus hakea asiakkaan tietoja nimen tai muiden hakuehtojen perusteella mielellään heti sovelluksen etusivulta. Nopea toiminta tuo vaatimuksia uuden asiakasrekisterin ulkonäölle, turhia graafisia tehosteita vältetään, jotta sivuston latausajat pysyvät kurissa.

#### 9.1.1 Tietoturva

Sovellus pitää sisällään asiakkaiden henkilö- ja osoitetietoja, joten sivustolle pitää kirjautua salasanan avulla. Asiakkaiden tiedot eivät missään nimessä saa päästä muiden kuin henkilökunnan tietoon. Kun tieto on WWW-palvelussa, on otettava huomioon sen tuomat riskitekijät, tietomurto on aina mahdollinen.

Koska asiakasrekisteriä pidetään auki työpäivän ajan, on henkilökunnan oltava varovainen, ettei pienen poissaolohetken aikana kukaan pääse luvottomasti tietoihin käsiksi.

### 9.1.2 Lomakekentät

Uusi asiakasrekisteri tulee pitämään sisällään samankaltaisia lomakepohjia, kuin vanha. Vanhan asiakasrekisterin ongelma oli esittäytyminen erittäin kapealla kaistaleella keskellä näyttöä, joten uudessa asiakasrekisterissä lomakepohjat tullaan levittämään leveämmälle osalle näyttöä, jotta sivustoa ei tarvitse vierittää pystysuunnassa. Lomakepohjat asetellaan järkevästi, leveällä kaistalla ei saa olla liian montaa kenttää vierekkäin. Samankaltaiset kentät, kuten etu- ja sukunimi sijoitetaan samalle riville, kuten kaikki puhelinnumerot ja sähköpostiosoitteet. Näin luonnollinen jaottelu säilyy vaikka käyttöliittymä levenee vanhasta huomattavasti.

### 9.1.3 Asiakashaku

Asiakashaussa on oltava tarpeeksi monta hakuehtoa, kuten nimi, osoite, asiakasnumero ja vapaahaku. Haun yhteydessä tullaan mainitsemaan kenttien täyttömuodot. Täyttömuodot olivat vanhassa asiakasrekisterissä suurena ongelmana. Koska kenttien yhteydessä ei kerrottu kentän täyttöohjetta, täytettiin kenttiä väärin.

Etusivulle sijoitetaan hakukentät, tulokset löydetyistä asiakkaista ja luettelo viimeisimmistä asiakastapahtumien muokkauksista. Sovelluksen tulee ymmärtää kääntää pienet ja isot kirjaimet, niin etteivät ne hämää hakutulosta. Hakutulokseen halutaan päästä käsiksi sovelluksen tuottamaa linkkiä painamalla, eli ainoastaan yhdellä klikkauksella.

Asiakkaan tietoja pitää päästä muuttamaan helposti, mikäli esimerkiksi puhelinnumero muuttuu. Muokkaus-painike pitää löytyä jokaiselta sivulta, jossa esitetään kyseisen asiakkaan tiedot. Muokkaus-napin lisäksi halutaan mahdollisuus asiakkaan poistamiseen. Helpoiten puhelinnumeroa pääsisi muuttamaan seuraavasti: asiakkaan haku etusivulta löytyvästä hakukentästä, hakutuloksen löydyttyä muokkaa-napin painallus ja tarvittavien tietojen muuttaminen avautuvasta lomakepohjasta ja tallennus.



---

#### 9.1.4 Uuden asiakkaan lisääminen

Uuden asiakkaan lisäys oli tehty vanhassa asiakasrekisterissä hankalaksi. Uuden asiakkaan lisääminen halutaan tehdä uudessa asiakasrekisterissä helpoksi ja lisäys voitaisiin tehdä esimerkiksi päänavigaatiosta tai muusta helposti löytyvästä kohdasta. Toiveena on, että yhtä linkkiä klikkaamalla käyttäjä saa suoraan eteensä uuden asiakkaan lomakentän.

Uudelle asiakkaalle määritellään nimi- ja osoitetietojen lisäksi puhelinnumero ja sähköpostiosoite. Vanhasta poiketen asiakkaan perustiedot halutaan tallentaa ensin ja vasta tämän jälkeen käyttäjällä olisi mahdollisuus lisätä asiakkaalle käyntitapahtuma. Asiakkaan tallennetut perustiedot halutaan pitää esillä koko ajan, jotta käyttäjä tietää varmasti olevansa oikean asiakkaan kohdalla.

#### 9.1.5 Raportit

eTuvassa ollaan kiinnostuneita toteutuneiden käyntien ja huoltojen jakaumista. Sovelluksesta pitää löytyä raporttiosio, josta saa tulostettua yhteenvedon toteutuneista tapahtumista. Jotta raporttien tekeminen onnistuu, pitää uudessa sovelluksessa olla käyntien tarkemmat tyyppimäärittelyt, esimerkiksi digikäynti tai tietokonekäynti. Tyyppimäärittelyn avulla saadaan myöhemmin tulostettua raportti, jossa nähdään digi- ja tietokonekäyntien jakauma. Toiseksi tärkeäksi raportiksi muodostuu se, miten toimistolla tehdyt huollot jakautuvat. Mahdollisuuksina on esimerkiksi virusturvan päivitys, käyttöjärjestelmän uudelleenasetus, yksityisopetus tai laitteistojen asennus. Työntekijöiden on oltava tarkkana tyyppiä valitessa, väärin valittu tyyppi vääristää raporteja.

#### 9.1.6 Asetukset

Yhteisissä neuvotteluissa esille nousi asetukset, joista käyttäjät voisivat helposti käydä vaihtamassa esimerkiksi vaihtuneen työntekijän tiedot. Vanhassa asiakasrekisterissä kummitteli vanhoja työntekijöitä, koska kukaan ei ole osannut käydä vaihtamassa tietoja.

### 9.1.7 Reskontra-toiminnot

Koska eTupa on pitänyt jokapäiväistä kirjanpitoaan tähän asti käsin usean eri Excel-taulukon avulla, uuden asiakasrekisterin toivottiin pitävän sisällään reskontra-ominaisuuksia, jotka poistaisivat tarpeen erillisten Excel-taulukkojen ylläpitämisestä. Esiin nousivat käteiskuitit ja näin automaattinen kassakirja, laskut ja tietyn henkilön autolla ajetut kilometrit. Erillisten taulukkojen ajantasainen ylläpitäminen on ollut hankalaa ja virheeltistä.

Uusi sovellus osaisi siis tehdä automaattisesti lisäyksen kassakirjaan, mikäli käyttäjä valitsee käynnin tiedoista käteismaksun ja laittaa hinta-kohtaan käynnin hinnan. Mikäli käyttäjä valitsee maksutavaksi laskun, olisi käyttäjän helppo tallennuksen jälkeen tulostaa asiakkaalle lasku käteiskuitin sijaan. Sovelluksen on siis tiedettävä minkä maksutavan käyttäjä on valinnut, jotta laskun tulostaminen ei ole silloin mahdollista, kun maksutavaksi on valittu käteismaksu. Tiedot kirjoitetuista laskuista ja käteiskuiteista halutaan löytyvän raportti-osiosta.

Kotikäyntien yhteydessä ajetut kilometrit lisätään oikealle henkilölle valitsemalla oikea henkilö pudotusvalikosta samaan tapaan, kuin laskun tai käteiskuitin valitseminen tapahtuu. Raportti-osiosta voidaan esimerkiksi kerran kuussa käydä tulostamassa yhteenveto ajetuista kilometreistä palkkahallintoa varten.

### 9.1.8 Helppokäyttöisyys ja käytännöllisyys

Uuden asiakasrekisterin myötä työntekijän arki tulee helpottumaan. Lisäksi yhdestä paikasta tapahtuva tietojen hallinta vähentää työntekijöiden virheitä.

Seuraavassa on esimerkki työntekijän vanhasta rutiinista toimistolla tapahtuneen huoltokäynnin jälkeen:

1. huoltoselosteen puhtaaksikirjoitus Wordillä ja tulostus
2. kuitin tai laskun kirjoitus Excelillä ja tulostus
3. mahdollisen käteissuorituksen kirjaus Excelin kassakirjaan
4. asiakasrekisterin kirjoitus

Uuden sovelluksen myötä eTuvan työntekijälle riittää seuraava työsuoritus:

1. huollon kirjaus sovellukseen ja tallentaminen
2. huoltoselosteen, laskun tai käteiskuitin tulostaminen

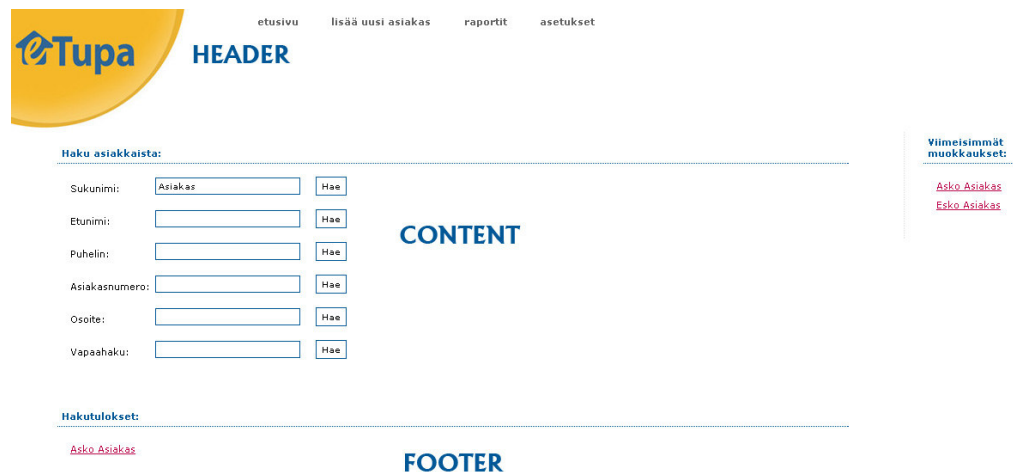
Myöhemmässä vaiheessa laskujen hallinnasta saatetaan tehdä automaattista. Kuitenkin vielä laskut tulostetaan ja niiden maksamista hallitaan käsin verkkopankin kautta.

## 9.2 Järjestelmän toteutus

Tässä osiossa käydään läpi uuden asiakassovelluksen toteutus. Koska asiakasrekisteri sijaitsi ennen Microsoft Access-ohjelmassa, on uusi sovellus täysin uusi ja se on suunniteltu vastaamaan eTuvan omia tarpeita. Sovellusta tullaan mahdollisesti laajentamaan myöhemmässä vaiheessa. Seuraavassa esitellään sovelluksen tulevaa toiminta-ajatusta. Itse olen toteuttanut kaikki sivuston ulkoasumääritykset. Yhdessä eTuvan muun henkilöstön kanssa olemme miettineet lomakekenttien sisällön.

### 9.2.1 Rakenne

Sivuston rakenne koostuu kolmesta osasta. Header-osasta löytyy eTuvan logo ja navigaatio. Content-osassa käyttäjälle näytetään varsinainen sivun sisältö, esimerkiksi hakulomakkeet tai hakutulokset. Oikealla oleva pystysuuntainen palkki sisältyy myös content-osuuteen. Kolmantena osana sivustolla on footer. Footeriin on mahdollista sijoittaa tekstiä, mutta tällä hetkellä footer-osalle ei ole varsinaista käyttöä.



Kuva 7 Sivuston rakenne

## 9.2.2 Visuaalinen ulkoasu

Visuaalisen ulkonäön perusteena käytin eTuvan omaa graafista ohjeistoa, jossa määritellään kaikki värit ja fontit, joita yrityksessä käytetään. Tämän perusteella suunnittelin sivuston ulkoasun. Ulkoasun on tärkeää olla rauhallinen, koska asiakasrekisteri on ahkerassa käytössä jokaisena työpäivänä. Se ei siis saa rasittaa työntekijöiden silmiä. Ulkoasuun oltiin eTuvassa erittäin tyytyväisiä. eTuvan värimaailma on erittäin raikas ja lopullinen tulos edustaa tätä raikkautta.

## 9.2.3 Kuvat

Sivustolla on aluksi ainoastaan yksi kuva, joka on yläreunassa oleva eTuvan logo. Logo tuo mukavaa raikkautta ja logo löytyy myös samasta kohdasta eTuvan virallisilta kotisivuilta. Myöhemmin sivustolle liitetään mahdollisesti karttakuva jokaisen asiakkaan tietoihin. Kuvan sijoituspaikkana tulisi olemaan sivuston oikea reuna.

## 9.2.4 Navigaatio



*Kuva 8 Navigaatio*

Sivuston navigaatio sijaitsee sivuston vasemmassa reunassa, eTuvan logon vieressä. Linkit löytyvät jokaiselta sivulta samasta kohdasta. Navigoinnista haluttiin tehdä helppoa, linkkejä ei haluttu tehdä liikaa. Myöhemmin sovelluksen laajentuessa, linkkilistaa voidaan helposti päivittää.

## 9.2.5 Typografia

Sivuston fonttina käytetään Verdanaa, joka on ulkonäöllisesti erittäin selkeä ja helppolukuinen fontti.

Leipätekstin koko on 11px, kuten myös otsikoiden. Otsikot ovat eroteltu muusta joukosta sinisellä värillä (#005897) ja alleviivauksella. Alleviivauksena käytetään pisteitä. Alleviivauksen avulla si-

vustolle saatiin rytmiä, rytmitys erottaa sivuston eri sisällöt, kuten lomakekentät ja hakutulokset toisistaan.

#### Haku asiakkaista:

*Kuva 9 Otsikon tyyli*

## 9.2.6 Sivuston osat

Sivusto on rakennettu käyttäen XHTML-kieltä. Sivuston osat tuodaan PHP:n include-komennolla, jotta sivustoa on helppo päivittää päivittämällä tiedot vain yhteen tiedostoon.

### Etusivu

The screenshot shows the home page of the Tupa website. At the top left is the Tupa logo. To its right is a navigation menu with links: etusivu, lisää uusi asiakas, raportit, and asetukset. Below the navigation is a search form titled "Haku asiakkaista:". The form contains several input fields, each with a "Hae" button next to it: Sukunimi (with "Asiakas" entered), Etunimi, Puhelin, Asiakasnúmero, Osoite, and Vapaahaku. To the right of the search form is a sidebar titled "Viimeisimmät muokkaukset:" containing two links: "Asko Asiakas" and "Esko Asiakas". Below the search form is a section titled "Hakutulokset:" with a single link: "Asko Asiakas".

*Kuva 10 Etusivu*

Etusivulta käyttäjä löytää nopeasti ja vaivattomasti hakukentät, joista voidaan hakea asiakkaita erilaisilla hakuehdoilla. Vanhassa asiakasrekisterissä käyttäjä joutui ensin käynnistämään Microsoft Access-ohjelman, tämän jälkeen sulkemaan avautuvan ylimääräisen ikkunan, käynnistämään oikean ikkunan ja vielä päälle painamaan haku-nappia, jotta pääsi kirjoittamaan hakusanaa. Toimenpiteiden määrä oli siis kohtuuton ja aikaa vievä. Nyt käyttäjälle riittää asiakasrekisterin avaaminen selaimen ja hakusanan syöttäminen oikeaan kohtaan.

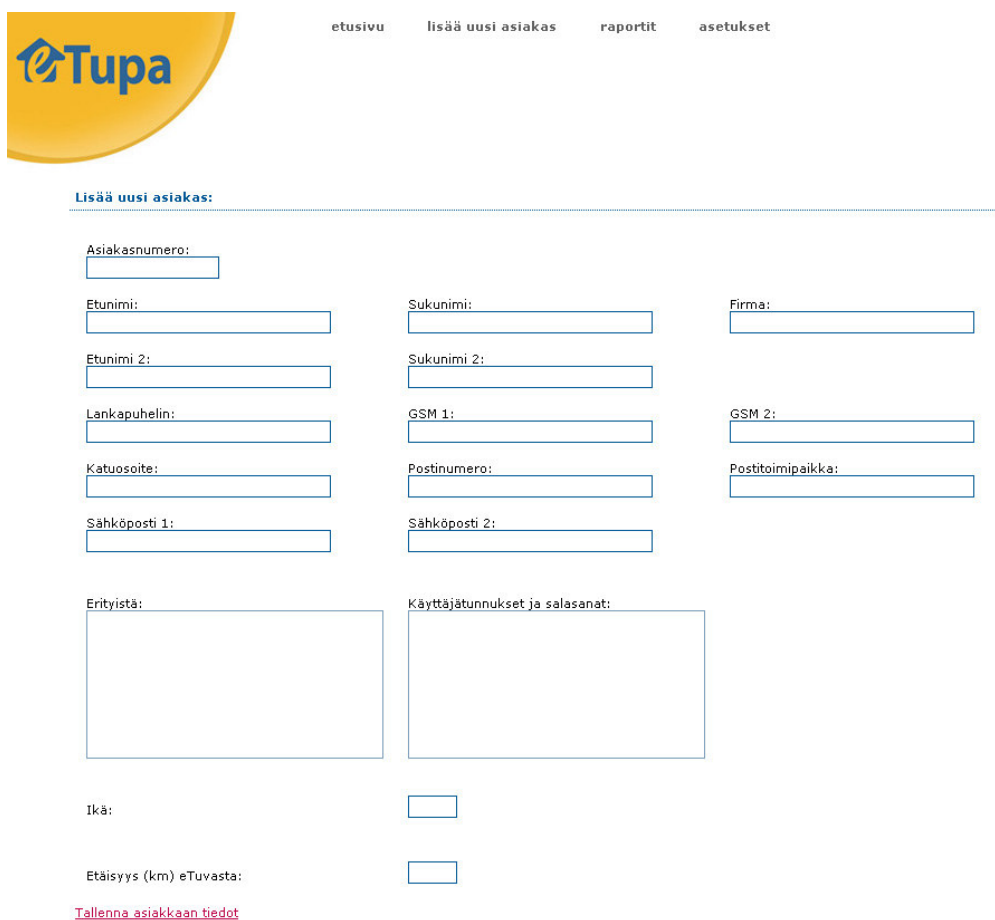
Hakukentiksi valittiin etu- ja sukunimi, asiakasnumero, katuosoite, puhelinnumero ja vapaahaku. Vapaahaussa käyttäjän on mahdollista hakea kaikilla mahdollisilla sanoilla.

Hausta haluttiin tehdä helppoa, joten sovellus osaa itse kääntää isot ja pienet kirjaimet sovelluksen halun mukaiseksi. Käyttäjä voi siis huolelta kirjoittaa etunimen pienellä kirjaimella ja sovellus osaa tästä huolimatta etsiä oikean asiakkaan.

Hakukenttien alapuolelle avautuu haun jälkeen lista löydetyistä asiakkaista. Linkkiä painamalla avautuvat asiakkaan perustiedot ja tehdyt toimenpiteet päivämääräjärjestyksessä sekä toimenpideluettelo, josta asiakkaalle voidaan lisätä uusi toimenpide. Käyttäjä voi valita seuraavista toimenpiteistä: kotikäynti, toimistolla tapahtuva työ tai laitehankinta. Hakukentät muistavat käyttäjän kirjoittamat hakuehdot, joten hakua on helppo myöhemmin tarkentaa, mikäli siihen on tarvetta.

Etusivun oikeasta reunasta käyttäjä näkee kahdeksan viimeisintä asiakastapahtumaa/muokkausta. Tämä helpottaa työntekijöiden arkea, jos asiakkaan tietoja pitää hakea useasti. Listan avulla asiakkaan tiedot saadaan esiin erittäin nopeasti, esimerkiksi puhelinnumeron etsintää varten. Linkkiä painamalla käyttäjä pääsee näkemään asiakkaan perustiedot ja listan käynneistä.

## Uuden asiakkaan lisäys



etusivu   lisää uusi asiakas   raportit   asetukset

**Lisää uusi asiakas:**

Asiakasnumero:

Etunimi:  Sukunimi:  Firma:

Etunimi 2:  Sukunimi 2:

Lankapuhelin:  GSM 1:  GSM 2:

Katuosoite:  Postinumero:  Postitoimipaikka:

Sähköposti 1:  Sähköposti 2:

Erityistä:  Käyttäjätunnukset ja salasanat:

Ikä:

Etäisyys (km) eTuvasta:

[Tallenna asiakkaan tiedot](#)

*Kuva 11 Uuden asiakkaan lisäys*

Ylhäältä olevasta linkkistasta käyttäjä löytää uuden asiakkaan lisäyksen. Uuden asiakkaan lisäämistä käytetään vain siinä tapauksessa, mikäli rekisteristä ei vielä löydy mitään merkintää asiakkaasta. Vanhoille asiakkaille toimenpiteet lisätään etusivun haun kautta.

Käyttäjän pitää aluksi täyttää asiakkaan perustiedot nimestä ja osoitteesta lähtien. Nimikenttiä on lisätty rekisteriin kaksi kappaletta, koska samasta perheestä saattaa olla moni eri ihminen hoitanut perheen asioita. Näin yhden asiakasnumeron alle saadaan lisättyä koko perhe. Postitusten yhteydessä on turhaa lähettää samaan perheeseen montaa kirjettä. Kirjoitettuaan perustiedot, käyttäjä tallentaa tiedot. Tämän jälkeen käyttäjä näkee yhteenvedon asiakkaan tiedoista (kuva 12) ja käyttäjän on mahdollista muokata tai poistaa asiakas.



etusivu lisää uusi asiakas raportit asetukset

Tänne tulee karttakuva.

#### Asiakkaan tiedot:

<b>Nimi:</b>	Asko Asiakas	<b>GSM 1:</b>	040 111 1111	<b>GSM 2:</b>	-
<b>Katuosoite:</b>	Osoite 1 A 2	<b>Postinumero:</b>	33100	<b>Postitoimipaikka:</b>	TAMPERE
<b>Email 1:</b>	asko.asiakas@asiakas.fi	<b>Email 2:</b>	asko.asiakas@asiakas.fi		
<b>Erityistä:</b>	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eget eros nec felis luctus tincidunt.				
<b>Safasanat:</b>	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eget eros nec felis luctus tincidunt.				
<b>Etäisyys (km) eTuvasta:</b>	25 km	<b>Ikä:</b>	25 v		

Muokkaa tietoja Tulosta tiedot Lisää laitteisto Poista

#### Valitse uusi tapahtumalaji:

[Kotikäynti](#)  
[toimistolla tapahtuva työ](#)  
[laittehankinta](#)

### *Kuva 12 Tallennetut perustiedot*

Seuraavaksi asiakkaalle lisätään laitteisto. Laitteiston lisääminen näkyy toimenpiteen lisäämisen yhteydessä, jossa voidaan valita oikea laite kyseiseen toimenpiteeseen. Asiakkaalle voidaan siis lisätä monta eri laitetta, kuten tietokone ja digisovitin.

## Tapahtumalajin valitseminen

Laitteiston lisäämisen jälkeen käyttäjän pitää lisätä asiakkaalle tapahtuma. Tapahtumalajia painamalla käyttäjälle avautuu näyttö johon kirjoitetaan tapahtumasta riippuen käynnin tiedot. Tässä osuudessa käyttäjä voi valita listasta, mitä laitetta käynti koskee. Mikäli laitteita ei ole vielä lisätty asiakkaalle, voidaan lisäys tehdä vielä tässä kohtaa kirjoittamalla laitteen nimi siihen tarkoitukseen tehtyyn kenttään. Laitte lisäättyä automaattisesti asiakkaan laitetietoihin ja näkyy seuraavilla kerroilla, jos asiakkaalle lisätään toimenpidettä. Koska laitteistolle lisätään tässä kohdassa vain nimi, voidaan laitteistoa käydä tarkentamassa asiakastietojen kohdassa ”muokkaa tietoja”.

Kun tapahtuma on kirjoitettu ja se on tallennettu, käyttäjälle avautuu näkymä, josta on mahdollista tulostaa huoltoseloste, lasku, käteiskuitti tai käydä muokkaamassa käynnin tietoja. Käyttäjä ei voi valita tässä yhteydessä laskun tulostamista, mikäli hän on valinnut maksutavaksi käteisen.

Kotikäynnin yhteydessä kirjataan seuraavat tiedot tietokantaan:



.....

**Salasanat:** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eget eros nec felis luctus tincidunt.

**Etäisyys (km)** 25 km      **Ikä:** 25 v

**eTuvasta:**

Muokkaa tietoja   Tulosta tiedot   Lisää laitteisto   Poista

**Kotikäynnin tiedot:**

**Päivämäärä (pp.kk.vvvv):**

**Laittevalinta:**  
 Topfield 6000C  
 Acer 1234  
 Uusi laite?

**Tyyppi:**  
Valitse

**Kävijä 1:**  
Valitse

**Kävijä 2:**  
Valitse

**Kulkuväline:**  
Valitse

**Ongelma:**

**Ratkaisu:**

**Jatkotoimet / erityistä:**

**Työaika (h):**

**Hinta:**

**Maksutapa:**  
Valitse

Tallenna

*Kuva 13 Kotikäynti*

Mikäli työ on tehty toimistolla, käyttäjä täyttää seuraavan lomakkeen:

**Salasanat:** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eget eros nec felis luctus tincidunt.

**Etäisyys (km) eTuvasta:** 25 km **Ikä:** 25 v

Muokkaa tietoja Tulosta tiedot Lisää laitteisto Poista

---

**Toimistolla tapahtuva työ:**

**Päivämäärä (pp.kk.vvvv):**

**Laitevalinta:**  
 Topfield 6000C  
 Acer 1234  
 Uusi laite?

**Tyyppi:**  
Valitse

**Käsittelijä 1:**  
Valitse

**Käsittelijä 2:**  
Valitse

**Ongelma:**

**Ratkaisu:**

**Jatkotoimet / erityistä:**

**Työaika (h):**

**Hinta:**

**Maksutapa:**  
Valitse

Tallenna

*Kuva 14 Toimistolla tapahtuva työ*

Laitehankinta toteutettiin siten, että käyttäjä kirjoittaa jokaisen tuotteen ja sen hinnan erikseen lomakekenttään ja painaa tämän jälkeen lisää-nappia. Nappi lisää lomakenttien yläpuolelle rivejä, joihin kerääntyvät lisätyt tuotteet. Samankaltainen toiminto on myös lisätietoja-kentässä. Hinta yhteensä lisääntyy automaattisesti, joten käyttäjän ei tarvitse tehdä laskutoimituksia. Lisääntyneet rivit pysyvät poistamaan, mikäli niihin on tullut virheitä. Kun laitehankinta näyttää oikealta, voidaan se tallentaa lopuksi kokonaisuudessaan.

**Salasanat:** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque eget eros nec felis luctus tincidunt.

**Etäisyys (km)** 25 km      **Ikä:** 25 v

**eTuvasta:**

Muokkaa tietoja    Tulosta tiedot    Lisää laitteisto    Poista

#### Syötä tuotteen tiedot:

Lisää painike tuo tuoterivit tähän.

Tuote:       Hinta:      

Lisätietoja:      

Hinta yhteensä:

Hinta yhteensä lisääntyy automaattisesti lomakekenttään.

[Hvähäy tarious](#)

*Kuva 15 Laitehankinta*

## Raportit

Raportti-osiosta käyttäjän on mahdollisuus valita nähtäväksi esimerkiksi listat ajetuista kilometreistä, kassakirjan ja listan käteiskuiteista. Lisäksi on mahdollisuus selata tehtyjä laskuja. Käyttäjä valitsee ensin listasta haluamansa raportin ja pääsee tämän jälkeen tarkastelemaan tarkemmin kyseisiin raporteihin tulleita lisäyksiä ja muutoksia.

## Asetukset

Asetusten olemassaoloa perusteltiin sillä, että käyttäjät haluavat päästä nopeasti ja helposti vaihtamaan otsikkoja tai lisäämään uusia valintoja valintalistoihin. Tämä kaikki voidaan tehdä myös HTML-kielen avulla, mutta mikäli käyttäjältä ei löydy taitoja HTML-kielestä, on päivittäminen helpompaa tämän käyttöliittymän avulla. Asetusten lopullinen käyttöliittymä päätetään myöhemmin, eikä sen suunnittelua ole sisällytetty tähän opinnäytetyöhön. Yrityksessä halutaan miettiä tarkemmin, mitä asetuksiin sisällytetään. Kaikkia muutoksia ei voida tehdä tämän käyttöliittymän sisällä, vaan ne pitää toteuttaa HTML- tai PHP-kielen avulla.

---

## 10 Yhteenveto

Mahdollisuus tämän opinnäytetyön toteuttamiseen nousi esiin suorittaessani työharjoittelua eTuvassa kesällä 2006. eTuvassa oltiin jo hetken aikaa mietitty uuden asiakasrekisterin rakentamista. Maksullisia vaihtoehtoja ei kuitenkaan haluttu ottaa huomioon, koska eTuvalla on huomattava määrä omaa ammattitaitoa asiakasrekisterin tekemiseen. Omalta osaltani asiakasrekisterin varsinaisen toiminnallisen toteuttaminen ei tullut kysymykseen ohjelmointitaitojen puuttumisen takia. Ehdotin, että toteuttaisin sivuston ulkonäön ja lomakekentät valmiiksi, jotta joku toinen voi sivuston toteuttaa tämän opinnäytetyön perusteella. Työskennellessäni eTuvassa sain kattavan kuvan yrityksen toiminnasta ja näin uutta asiakasrekisteriä oli helppo lähteä suunnittelemaan. Varsinaisesti uuden asiakasrekisterin suunnittelu alkoi syksyllä 2006.

Yrityksen puolesta suunnitteluprojektissa olivat mukana kaikki eTuvan työntekijät. Suurin sanavalta oli kuitenkin yrityksen omistajalla Emma Littusella.

Koska yrityksessä oltiin erittäin turhautuneita vanhan asiakasrekisterin toimintaan, suunnitteluvaiheessa esitettiin erittäin paljon erilaisia vaihtoehtoja uuden toimintaa ajatellen. Jokaisella oli paljon painavia mielipiteitä. Vanha asiakasrekisteri jopa haittasi yrityksen toimintaa, koska asiakkaiden tiedot olivat puutteellisia kenttien epäselvyydestä johtuen. Vaikean käytettävyyden takia työntekijöiden motivaatio asiakasrekisterin täyttämiseen oli matala. Uudesta rekisteristä oli tärkeää saada toimiva juuri eTuvan kannalta ja näin työntekijöiden motivaatio asiakasrekisterin tarkkaan täyttämiseen nousee.

Ulkonäöllisesti uudesta rekisteristä haluttiin saada raikas. Lisäksi vanhan asiakasrekisterin kapeasta näytöstä haluttiin päästä eroon. Tämä toteutettiin uuteen, eli uusi asiakasrekisteri esittäytyy huomattavasti suuremmalla osalla näyttöä ja se vähentää turhaa näytön rullausta. Lomakepohjat uusittiin osin. Asiakkaan perustiedot pysyivät suhteellisen samoina, mutta suurimmat uudistukset tehtiin laitetietoihin, joihin työntekijät voivat lisätä samalle asiakkaalle useita laitteita. Lisäksi työntekijöiden arki helpottuu automaattisen reskontran avulla.

Mielestäni onnistuin sivuston toteutuksessa erittäin hyvin verrattuna siihen, että en ole ennen suunnittelut näin suurta kokonaisuutta. Matkan varrella opin paljon uusia asioita käymällä alan kirjallisuutta tarkemmin läpi. Koulutusohjelman aikana tulleet opit nivoutuivat yhteen ja ammattitaitoni syveni huomattavan paljon.

Projektin aikataulu venyi, koska yrityksessä oli kiireitä talven aikana ja suunnittelupalavereita ei ehditty pitämään. Lisäksi aikataulun venymiseen liittyi omat kiireet. On kuitenkin tärkeää, että työ on valmis ennen kesää, koska kesällä Suomi hiljentyy kesän viettoon ja yrityksenkin kiireet pienevät samalla. Työn toteutukseen löytyy siis hyvin aikaa ja sen testaamiseen ehditään paneutumaan kunnolla. Asiakasrekisteri tulee hioutumaan uomiinsa, kun se saadaan toteutettua kesän 2007 aikana. Täydellistä sovellusta ei koskaan saada aikaiseksi, mutta tämän pohjalta on hyvä lähteä eteenpäin.

---

# 11 Lähteet

## Kirjallisuus

- Ek, Jesper, Eriksson Ulrika 2001. XHTML-käsikirja. Suom. R. Lundahl. Vantaa: Schildts Kustannus Oy.
- Heinisuo Rami 2003. PHP ja MySQL: tietokantapohjaiset verkkopalvelut. Helsinki: Talentum.
- Korpela, Jukka, Linjama, Tero 2005. Web-suunnittelu. Porvoo: WS Bookwell.
- Kuutti, Wille 2003. Käytettävyys, suunnittelu ja arviointi.
- Linjama, Tero 2001. XHTML. Jyväskylä: Docendo Finland Oy.
- Lukkarila, Jarmo 2001. Tekstuuri - typografia julkaisijan työvälineenä. Keuruu: Otavan kirjapaino Oy.
- Sinkkonen, Irmeli, Kuoppala, Hannu, Parkkinen, Jarmo & Vastamäki, Raino 2002. Käytettävyyden psykologia. Helsinki: Edita Oyj.
- Zeldman, Jeffrey 2003. Designing with web standards.
- Veen, Jeffrey 2002. Inside Webdesign. Suom. T. Haanpää. Edita: Edita Publishing Oy.
- Wiio, Antti 2004. Käyttäjätavallisen sovelluksen suunnittelu. Helsinki: Edita Prima Oy.

## Verkkolähteet

- Heinonen, Petri, Lahtonen, Tommi 2004. Lomake-elementit. [online] [viitattu 21.3.2007] <http://appro.mit.jyu.fi/doc/forms/index2.html>
- Laak, Timo 2006. Saavutettavaa typografiaa - osa 1. [online] [viitattu 27.4.2007] <http://saavutettava.fi/artikkelit/saavutettavaa-typografiaa-osa-1/>
- Mäntylä, Jukka, Lahtonen, Tommi & Ekonoja Antti 2007. Www-lomakkeet. [online] [viitattu 21.3.2007] <http://appro.mit.jyu.fi/www/luennot/luento7/#TOC2>
- W3Schools 2007. XHTML dtd. [online] [viitattu 21.3.2007] [http://www.w3schools.com/xhtml/xhtml\\_dtd.asp](http://www.w3schools.com/xhtml/xhtml_dtd.asp)
- W3Schools 2007. A very simple HTML document. [online] [viitattu 6.4.2007] [http://www.w3schools.com/html/tryit.asp?filename=tryhtml\\_basic](http://www.w3schools.com/html/tryit.asp?filename=tryhtml_basic)

**Muut lähteet**

Littunen, Emma 2005. eTuvan perustaminen, selvitys neuvontapalvelun toimintamallista.