



TAMPEREEN
AMMATTIKORKEAKOULU

OPINNÄYTETYÖRAPORTTI

WWW-sivuston toteutussuunnitelma
Case: Tampereen kesäyliopisto

Sari Stockus

Tietojenkäsittelyn koulutusohjelma
maaliskuu 2007
Työn ohjaaja: Petri Heliniemi

TAMPERE 2007



Tekijä(t):	Sari Stockus	
Koulutusohjelma(t):	Tietojenkäsittely	
Opinnäytetyön nimi:	WWW-sivuston toteutussuunnitelma Case: Tampereen kesäyliopisto	
Työn valmistumis- kuukausi ja -vuosi:	maaliskuu 2007	
Työn ohjaaja:	Petri Heliniemi	Sivumäärä: 55

TIIVISTELMÄ

Opinnäytetyön tavoitteena oli suunnitella nykyaikainen WWW-sivusto. Tekniikan kehittyminen on mahdollistanut sen, että Internetiin pääsee esimerkiksi matkapuhelimilla. Osa sivustoista on kuitenkin vain rajatun käyttäjäryhmän saavutettavissa, koska ne vaativat esimerkiksi tietynlaista selainta tai laitetta. Lisäksi usein sivuston ylläpito on hankalaa. Jos sivustoa ei pidetä ajan tasalla, käyttäjät alkavat etsiä tietoa muilta, ajan tasalla olevilta sivustoilta.

Työn toimeksiantaja, Tampereen kesäyliopisto halusi korjata nykyisen WWW-sivustonsa ongelmakohtia. Tein toteutussuunnitelman, jossa huomioidaan sivuston päivitys, käytettävyys ja saavutettavuus. Näihin asioihin pyritään vaikuttamaan uusimalla koko sivuston rakenne sekä tekninen toteutus. Suunnitelman pohjalta kesäyliopisto valitsee toteuttajan.

Suunnitelmaa tehtäessä on haastateltu kesäyliopiston henkilökuntaa sekä muutamia muita sivuston ylläpidon kanssa tekemisissä olevia henkilöitä. Lisäksi lähteinä on käytetty kirjoja sekä Internet-sivustoja. Tiedyt tekniikat ja ratkaisut sopivat useille muillekin sivustoille, mutta jokaista sivustoa suunniteltaessa on myös huomioitava erikseen sivuston tarkoitus, vaatimukset ja rajoitteet.



Author(s): Sari Stockus
Degree Programme(s): Business Information Systems
Title: Desinging plan for a website
Case: Summer University of Tampere
Month and year: March 2007
Supervisor: Petri Heliniemi **Pages:** 55

ABSTRACT

The purpose of this thesis was to design the modern website. Every websites are not attainable with new devices. Also there are often some problems with usability and the website updating might be difficult. A good website notice customers and administrators.

I made the website design plan for the Summer University of Tampere. They wanted to solve usability and website updating problems with their own website. I made some interviews, read books and browsed different websites and tried to solve these problems. I made the plan how they can renew their website, and what kind of things they should notice.

There are a lot of websites with these same problems. In every websites have their own limits and challenges, but this study might help and give some good advices.

SISÄLLYSLUETTELO

Käsitteet ja lyhenteet	6
1 Johdanto	7
2 Merkkaukielet	9
2.1 HTML	9
2.1.1 HTML-dokumentti	9
2.2 XML	10
2.3 XHTML	10
2.3.1 XHTML 1.0 eroja verrattuna HTML 4.0 standardiin:.....	11
2.3.2 XHTML-dokumentin rakenne	11
2.3.3 Esimerkki XHTML dokumentista:.....	12
2.3.4 Eroja Strict ja Transitional dokumenttityyppien määrittelyssä	13
3 CSS Tyyliiedostot	14
3.1 Tyylien määrittäminen	14
3.2 Div-elementti	16
3.3 Pseudoluokat	17
3.4 Tyylien arvojärjestys.....	18
4 Palvelinsovellusten ohjelmointia	19
4.1 PHP	19
4.3 SSI.....	20
5 WWW-sivuilla käytettäviä tekniikoita	21
5.1 Tietokannat.....	21
5.2 JavaScript	21
5.3 Julkaisuohjelmat.....	22
6 Nykyinen WWW-sivusto	23
6.1 Tekniset ongelmat	23
6.2 Ulkoasu	24
6.3 Käytettävyys ja päivitys	24
7. Uuden sivuston mahdollisuudet ja rajoitukset	26
7.1 Ratkaisuja Suomen muissa kesäyliopistoissa	26
7.2 Tampereen yliopiston palvelin.....	27
7.3 Tampereen kesäyliopiston tavoitteita uudelle sivustolle.....	28
7.4 Laatuksiteerit.....	28
8 Uusi sivusto	30
8.1 Visuaalinen ulkoasu	30
8.2 Kuvat.....	31
8.3 Päänavigaatio	32
8.4 Pääsivujen vasemman reunan alanavigointi.....	32
8.5 Typografia.....	33
8.6 Otsikot.....	33
8.7 Värit	33

8.8 Hakukone	34
8.9 Tyylitiedosto	35
9 Sivuston päivitys	36
9.1 Kurssien päivitys.....	36
9.1.1 Ammatilliset täydennyskoulutukset - Eduix ja Kuha	36
9.1.2 Avoin yliopisto-opetus - Meteori-tietokanta ja täsmähaku	37
9.2 Sivuston muu päivitys	38
9.2.1 Adobe Contribute	38
9.2.2 Macromedia Dreamweaver.....	39
9.2.3 Macromedia Dreamweaver vastaan Adobe Contribute.....	40
9.3 Toistuvat osiot – PHP	41
10 Sivuston toteutukseen liittyviä asioita	43
10.1 Tiedostojen järjestäminen ja käsittely	43
10.2 PDF-tiedostot	43
10.3 Testaaminen ja validointi	44
10.4 Toteutussuunnitelma	45
10.4.1 Budjetti ja Aikataulu.....	45
10.4.2 Materiaali.....	45
11 Tulosten tarkastelu ja arviointi.....	46
Lähteet.....	47
Liitteet.....	49
Liite 1 – Kesäyliopistojen sähköpostihaastattelu	49
Liite 2 - Navigaatiokartta	52
Liite 3 - Tyylitiedosto.....	53

Käsitteet ja lyhenteet

CCS	Cascading Style Sheets CSS Tyyliohje, joka määrittelee WWW-sivun ulkoasun.
DTD	Document Type Definition Määrittelee dokumentin tyyppin sekä sen sisältämät elementit ja niiden kieliopin.
HTML	HyperText Markup Language Rakenteellinen formaatiokieli WWW-sivustojen tekemiseen.
PDF	Portable Document Format Aboden kehittämä sähköinen julkaisukeino
SGML	Standard Generalize Markup Language Yleinen dokumenttien rakenteen kuvaamisen kieli.
SSI	Server Side Include Palvelinpuolen ohjelmoinnissa käytettävä tekniikka.
PHP	Hypertext Preprocessor Palvelinpuolen skriptikieli.
XHTML	Extensible HyperText Markup Language XML-pohjainen kieli WWW-sivustojen tekemiseen.
XML	Extensible Markup Language Metakieli, jolla luodaan rakenteellisia merkkauksikieliä.
XSL	eXtensible Style Language XSL Tyyliohje, jolla määritellään XML-dokumentin ulkoasu.
W3C	World Wide Web Consortium Julkaisukielten kehitystä ohjaava organisaatio.
WAI	Web Accessibility Initiative W3C:n laatima ohjeistus. Määrittelee sekä teknisiä, että sisällöllisiä toteutustapoja WWW-sivustoilla.
Validointi	Dokumentin kielen tarkastamista dokumenttityypin ja yleisten sääntöjen mukaan.
WWW	World Wide Web

1 Johdanto

WWW-sivujen saavutettavuudella tarkoitetaan erityisryhmien huomioimista eli sitä, että sivusto olisi mahdollisimman monen käyttäjän saatavilla. Standardien ja W3C:n laatiman ohjeistuksen mukaisesti tehdyillä sivustoilla on mahdollisuus saavuttaa hyvinkin suuri käyttäjäryhmä. Tämä ohjeistus eli WAI määrittelee sekä teknisiä, että sisällöllisiä toteutustapoja. Esimerkiksi sivuston sisällön pitäisi olla saavutettavissa, vaikka käyttäjä määritteli itse sivuston käytölle rajoituksia, kuten asettaisi kuvat, tyyli tai selainpohjaiset ohjelmointikielet pois käytöstä. (Keränen, Lamberg ja Penttinen 2006: 16)

Saavutettavuus ei vielä tee sivustosta helppokäyttöistä. Sivuston suunnittelussa tulisi huomioida myös sivuston käytettävyys: onko tieto helposti löydettävissä ja löytyykö sivustolta tai tietyn linkin takaa se tieto, jonka siellä voisi olettaa olevan.

Tämän raportin tavoitteena on antaa lukijalle mielikuvia siitä, mitä asioita kannattaa huomioida täysin uuden ja standardien mukaisen sivuston luomissa. On varmasti muitakin asioita ja tapoja tehdä tai suunnitella uusi sivusto, mutta haluan esitellä tässä joitakin vaihtoehtoja. Sivuston rakentaminen ei ole vain koodin kirjoittamista, vaan se vaatii etukäteissuunnittelua ja monien eri asioiden huomioimista.

HTML- ja XHTML-kielen erot eivät WWW-sivujen tekijän kannalta ole mahdollisimman suuria. Saavutettavuudessa ja käytettävyudessa erot voivat kasvaa jo suuremmiksi. Oikein tehdyllä XHTML-sivustolla on mitä todennäköisimmin suurempi käyttäjäryhmä kuin HTML-sivustolla. Tekniikan kehittyessä WWW-sivustoja selataan yhä enemmän muillakin laitteilla kuin tietokoneilla. Myös erityisryhmät pystyvät selaamaan Internetiä parantuvien laitteiden avulla. Käyttäjien lisäksi sivustoa suunniteltaessa pitäisi muistaa ottaa huomioon sivuston ylläpitäjät/päivittäjät. Laajan sivuston päivittämisessä on paljon työtä. Jos työ alkaa tuntua ylivoimaiselta, sivustoa ei enää pidetä ajan tasalla ja käyttäjät alkavat etsiä tietoa muilta, ajan tasalla olevilta sivustoilta.

Päivittämiseen on tarjolla paljon erilaisia ratkaisuja. Useimmat ratkaisusta vaativat tiettyjä ominaisuuksia palvelimelta, joka pyörittää sivustoa. Useimmat palvelimet pystyvät tarjoamaan näitä vaadittuja ominaisuuksia, kuten tuen tietokannoille ja PHP-ohjelmointikielelle. Useita sivustoja päivitetään jo suoraan selaimen kautta, eikä päivittämisestä huolehtivan tarvitse tietää mitään sivun koodauksesta.

Toimeksiantaja – Tampereen kesäyliopisto

Tampereen kesäyliopisto on perustettu vuonna 1955. Se on Suomen neljäs kesäyliopisto. Tampereen kesäyliopisto tarjoaa ammattillista täydennyskoulutusta sekä avointa yliopisto-opetusta. Kesäyliopiston koulutus on avointa kaikille iästä ja aikaisemmasta koulutuksesta riippumatta. Toiminta-alueena on koko Pirkanmaa. (Tampereen kesäyliopisto, 2004)

Suunnittelin Tampereen kesäyliopistolle WWW-sivuston toteutussuunnitelman. Nykyisin käytössä olevan sivusto on vanhentunut monilta osin. Suunnitelmaa tehdessä yritin huomioida käyttäjän sekä kesäyliopiston henkilökunnan näkökannan. Sivuston toteuttamiseen käytetään XHTML-kieltä ja päivittämiseen on yritetty löytää vaihtoehtoisia ratkaisuja. Sivuston testaamiseen käytetään useita menetelmiä ja niiden avulla pyritään luomaan uudesta sivustosta mahdollisimman saavutettava ja käytettävä. Työ rajattiin siten, että en toteuta uutta sivustoa, vaan suunnittelin sen toteutuksen ja ylläpidon.

Lähteet

Työni lähteitä olivat Jeffrey Zeldmanin *Designing with Web standards* (2003) ja Tero Linjaman *XHTML*-kirja (2001). Kirjoista löytyi paljon tarvittavaa teoretietoa perustellusti selitettynä. Lisätietoa asioista etsin Internetistä sekä muusta lähdekirjallisuudesta. Työn loppuvaiheessa tärkeäksi kirjaksi muodostui vuonna 2006 julkaistu Keräsen, Lambergin ja Penttisen *Web-julkaiseminen & Multimedia*. Kirja esittelee lyhyesti, mutta ymmärrettävässä muodossa Internet-julkaisussa käytettäviä tapoja ja ohjelmointikieliä. Painettuja lähteitä etsiessäni, tärkeimpänä kriteerinä pidin sisällön ajantasaisuutta. Sisältöä arvioin esimerkiksi sillä, miten hyvin se vastaa W3C:n virallisen sivuston määrittelemiä suosituksia.

Internet ja siihen liittyvät tekniikat kehittyvät nopeasti. Joistakin asioista on vaikea tai jopa mahdotonta löytää ajan tasalla olevaa lähdetietoa painetussa muodossa. Internetissä tietoa on paljon ja mitä erilaisimmilla sivustoilla. Lähes kuka tahansa voi julkaista tietoa, joten mistä tietää, mikä tieto on luotettavaa. Luotettavien sivustojen arvioimiseen ei ole yhtä ja oikeaa kaavaa, vaan käyttäjän pitää arvioida luotettavuutta esimerkiksi sivuston ylläpitäjän ja mahdollisen lähdeluettelon perusteella.

Niin sanotut viralliset sivustot, kuten PHP:n kotisivu, www.php.net, tai W3C:n oma sivusto www.w3.org/, ovat varmasti luotettavia sivustoja. Internetissä on paljon niin sanottuja wikipedia-sivustoja. Niille käyttäjät kirjoittavat tietoa, jota joku muu voi korjata tai täydentää. Ei kuulosta lähdetietona kovinkaan luotettavalta, mutta toisaalta tunnettuja wiki-sivuja käyttävät tuhannet ihmiset, ja useimmiten väärään tietoon reagoidaan. Wikitietoa voi käyttää ainakin tiedon täydentämiseen tai aineistona, kun jollekin asialle etsii selitystä. Kun asia selkenee, myös muunlaisen luotettavamman lähdetiedon löytyminen voi helpottua.

2 Merkkkauskielet

2.1 HTML

1990-luvun puolivälissä Internetin ja WWW-sivujen suosio kasvoi räjähdysmäisesti. Tämä johtui hypertekstiin perustuvien WWW-sivujen lisääntymisestä sekä Internet-selaimien kehittymisestä helppokäyttöisemmiksi ja monipuolisemmiksi. Lisäksi uudet tiedonsiirtotekniikat ja modeemien tiedonsiirtokyky olivat huomattavasti parantuneet sekä palveluntarjoajia ilmaantui koko ajan lisää. Tiedon julkaiseminen oli helppoa yksinkertaisen, sekä virheitä hyvin sietävän HTML-koodin avulla. (Linjama 2001: 20-21)

HTML on rakenteellinen formaatiokieli, jota käytetään dokumentin muotoiluun selainikkunassa. HTML:n pohjana on SGML-standardi. SGML-kieli julkaistiin 1980-luvulla, mutta kieli ei saanut suurta suosiota osakseen, koska se on virheherkkää ja kieliopiltaan laaja ja vaikeasti hallittava. HTML-dokumenteista haluttiin helposti muokattavia ja ulkoasultaan näyttäviä, joten rakennekomentoihin liitettiin esitystapaa, eli formatointia, koskevia ohjeita. Helppokäyttöisyytensä vuoksi HTML:n suosio kasvoi nopeasti. Ongelmien välttämiseksi perustettiin W3 Consortium, jonka tehtävä oli päättää minkälaista kieltä selainohjelmat tulkitsevat. Selainvalmistajat eivät kuitenkaan noudattaneet kaikkia W3C:n ohjeita, vaan tekivät myös omia ratkaisujaan selainten tulkitsemaan HTML-kieleen. (Linjama 2001: 20-21)

Lähes kaikki HTML:n komennot sisältävät rakenteellisen kuvauksen lisäksi kuvauksen ulkoasusta. Elementtejä on vuosien saatossa tullut paljon lisää, ja osa sisäkkäin rakennetuista komennoista voi olla hyvinkin ristiriitaisia keskenään. HTML-dokumentteja on Internetissä satoja miljoonia eikä niiden kirjoitettu merkkkaus ole yhtenäistä. Tämä hankaloittaa esimerkiksi tiedon etsintää ja käsittelyä. (Linjama 2001: 17-23, Keränen ym. 2006: 30-31)

2.1.1 HTML-dokumentti

HTML on rakenteisen tekstin merkintäkieli. Kielessä on elementtejä, jotka kirjoitetaan toistensa sisälle. Dokumentissa voi olla vain yksi juurielementti. HTML-dokumentin perusajatus on Linjaman (2001: 62) mukaan seuraava:

```
<html>
<head>
<title>Tampereen kesäyliopisto</html>
<body>
<h1>Tampereen Kesäyliopisto</h1>
<p>Tampereen kesäyliopisto järjestää avointa yliopisto-
opetusta, ammatillista täydennyskoulutusta sekä kieli- ja
abikursseja ympäri vuoden.</p>
</body>
</html>
```

HTML-kieltä kirjoitettaessa koodiin tulee kuitenkin usein virheitä, esimerkiksi päättötagit puuttuvat tai attribuutin arvon lainausmerkeistä puuttuu toinen. Lisäksi HTML on formatointikieli, joten HTML-dokumentti sisältää usein ulkoasumäärittelyjä. Selaimet sallivat virheellistä HTML-koodia tai yrittävät korjata sitä oletuksien perusteella. Siksi sama sivusto saattaa näyttää erilaiselta ja toimia eri tavoin eri selaimissa. (Linjama 2001: 67)

2.2 XML

XML on metakieli, jolla luodaan rakenteellisia merkkäuskieliä. Se on SGML-kielestä johdettu kielioppi, joka perustuu rakenteellisten dokumenttien laajennettavaan ajattelumallin. XML-kielestä luodaan uusia kieliä, joilla kaikilla on sama kielioppisääntö. Tämä mahdollistaa esimerkiksi sen, että sama dokumentti voidaan avata useassa eri ohjelmassa tai sovelluksessa. Tieto on tekstimuodossa ja se sisältää vain rakenteellisia ohjauskomentoja. XML on laajennettavissa ja järjestelmäriippumaton. Sitä käytetään muun muassa XML-pohjaisten kielten luomiseen, alustariippumattomaan tiedonsiirtoon, tiedon tallentamiseen, sekä tiedon hallintaan. (Linjama 2001: 23-24, 806, Keränen ym. 2006:32)

XML-kieltä ei ole suunniteltu WWW-sivujen tekoon, vaikka se soveltuukin siihen. Vanhoilla selaimilla ei ole tukea XML-kielelle, mutta uudemmat selaimet tukevat jo joitakin ominaisuuksia. XML-dokumentti sisältää vain tietosisällön ja rakenteellisesti kuvaavat elementit. Ulkoasun muotoilu voidaan tehdä CSS- tai XSL-tyyleillä. (Linjama 2001: 806, 811, 823)

XML-dokumentit koostuvat elementeistä. Jokaisen XML-dokumentin pitää sisältää yksi juurielementti. Otsikko kertoo, että dokumentti sisältää XML-standardin mukaan jäsenneiltyä tietoa. Keräsen ym. (2006: 32) mukaan XML-dokumentti rakentuu seuraavasti. Esimerkin juurielementtinä on dokumentti.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<dokumentti>
<nimi>asiakirja</nimi>
<versio>toinen versio</versio>
</dokumentti>
```

2.3 XHTML

XHTML esiteltiin tammikuussa vuonna 2000. XHTML on HTML-kielen kaltainen julkaisukieli, mutta kaikki elementit esitetään XML-kielen säännöin. XHTML sisältää komentoja, joiden nimistä voidaan päätellä jotain tietosisällöstä. Lisäksi siinä on tarkoin määritelty kielioppi, joten selainohjelmat eivät periaatteessa voi tulkita komentoja väärin, vaan ohjeet ovat yksiselitteiset. Sivuja voidaan myös katsoa muilla kuin tietokoneen selaimilla, esimerkiksi XHTML-selaimella varustetulla matkapuhelimella. (Keränen ym. 2006: 30-31, Linjama 2001: 23-25, 854)

2.3.1 XHTML 1.0 eroja verrattuna HTML 4.0 standardiin:

Merkittävimmät erot HTML:n ja XHTML:n välillä ovat Zeldmanin (2003: 154-165) mukaan:

1. Kaikki elementit ja attribuutit kirjoitetaan pienillä kirjaimilla.
2. Kaikki elementit täytyy päättää /-merkillä, myös yksiosaiset (esimerkiksi `
`). HTML sallii käytettävän esimerkiksi `p`-elementtiä vain avaamalla sen, mutta XHTML:ssä kaikki elementit päätetään (esimerkiksi `<p>`). Avonaiset elementit saattavat aiheuttaa muun muassa ongelmia tyylien suhteen.
3. Elementtien avaus- ja sulkujärjestyksen täytyy täsmätä. Dokumentti ei saa sisältää elementtejä, joita ei päätetä, mutta ei myöskään suljettuja elementtejä, joita ei ole aloitettu.
4. Attribuuteilla täytyy olla jokin arvo, esimerkiksi formin yhteydessä `selected="selected"`.
5. Attribuuttien arvot laitetaan lainausmerkkeihin. Esimerkiksi `height="25"`.
6. Elementin valinnan tulisi ilmaista aina sisällön rakennetta (ei ulkoasua). Esimerkiksi jos dokumentissa on lista, se tulisi tehdä lista-elementillä eikä esimerkiksi rivinvaihdoilla.
7. `<` ja `&`-merkit täytyy esittää muodoissa `<` ja `&`. Lisäksi kaikki erikoismerkit olisi hyvä enkoodata kuten esimerkiksi ä ja ö.

2.3.2 XHTML-dokumentin rakenne

XHTML-dokumentin alussa määritellään dokumentin tyyppi eli dtd. Dtd määrittelee dokumentin tyyppin sekä sen sisältämät elementit ja niiden kielopin. Tämä tehdään !DOCTYPE-merkinnällä. Suositeltavia dokumenttimäärittelyksiä ovat Strict tai Transitional. Tyypimäärittelyn avulla dokumentti validoidaan. Validointi tarkoittaa kielen rakenteen oikeellisuuden tarkastamista. (Linjama 2001: 854-855, Keränen ym. 2006: 42)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

tai

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

XHTML-dokumentin juurielementtinä on aina HTML-elementti. HTML-elementtiin liitetään xmlns-attribuutti, joka kertoo dokumentin XML-nimiavaruuden. Attribuutin arvoksi annetaan W3C:n XHTML nimiavaruuden osoite. (Keränen ym. 2006: 43)

```
<html xmlns="http://WWW.w3.org/1999/xhtml">
```

HTML-elementti rajaa dokumentin tekstit ja se sisältää aina head- sekä body-elementin. Head-elementin vähimmäisisältö on dokumentin kirjoitettu otsikko, eli title-elementti. Title-elementin sisältö näkyy selaimen yläreunassa. Kaikki selainikkunassa näkyvät tekstit kirjoitetaan body-elementin sisään. (Linjama 2001: 148-151, Keränen ym. 2006: 42-43)

Metatieto on kuvaavaa tietoa dokumentista, esimerkiksi dokumentin tekijät, hakusanat ja sisällönkuvaus voidaan merkitä metatietoon. Meta-elementit sijoitetaan head-elementin sisälle. Elementteillä tulee olla content-määrite sekä joko name- tai http-equiv-määrite. WWW-sivulla olisi hyvä olla vähintään merkistökoodauksen metatieto. ISO-8859-1, joka tunnetaan myös nimellä ISO Latin 1, on Suomessa yleisimmin käytetty merkistösopimus. ISO Latin 1 on suunniteltu länsi- ja pohjoiseurooppalaisille kielille. (Keränen ym. 2006: 44, Korpela ja Linjama 2004: 155)

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
```

2.3.3 Esimerkki XHTML dokumentista:

XHTML-dokumentin runko mukailleen Zeldmania (2003: 158-159):

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://WWW.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
html xmlns="http://WWW.w3.org/1999/xhtml">

<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta name="author" content="kesäyliopisto" />
<meta name="description" content="Tampereen kesäyliopiston kotisivut" />
<meta name="keywords" content="kesäyliopisto, koulutus, kurssit, Tampereen kesäyliopisto" />

<title>Tampereen Kesäyliopisto</title>
<link href="tyylit.css" rel="stylesheet" type="text/css" media="screen" />
</head>

<body>
<h1>Tampereen Kesäyliopisto</h1>
<p>Tampereen kesäyliopisto järjestää avointa yliopistopetusta, ammatillista täydennyskoulutusta sekä kieli- ja abikursseja ympäri vuoden.</p>

</body></html>
```

2.3.4 Eroja Strict ja Transitional dokumenttityyppien määrittelyssä

Suurimmat erot Strict ja Transitional dokumenttimäärittelyssä ovat Zeldmanin (2003: 69-72) mukaan:

Strict:

1. Kaikki ulkoasuun liittyvä määrittely on tyylietiedostossa.
2. Ei taulukkotaittoa; ainoastaan tekstiä/asiaa sisältäviä taulukkoja.
3. Suositeltava tapa uusien sivujen perustaksi, jos käyttäjäkunnassa on suurimmalla osalla käytössään uudehkoja selaimia.

Transitional:

1. Sallii joidenkin ulkoasuun vaikuttavien elementtien ja attribuuttien käytön.
2. Pääasiallisesti tyylimäärittelyt CSS:ään, esimerkiksi tekstiin, väreihin ja marginaaleihin liittyvät tyylit.
3. Voidaan tehdä taulukkotaittoa, mikäli siten saadaan parempi tulos käytettävyydessä kuin CSS:llä.
4. Ei suositella kokonaan uusien sivujen perustaksi, ellei käyttäjäkunnassa ole vielä paljon vanhojen selaimien käyttäjiä.

3 CSS Tyylitiedostot

Tyylien tarkoitus on kuvata sivuston ulkoasua. Tyylien avulla pystytään dokumentista karsimaan pois kaikki tyylimäärittelyt ja yhdellä muutoksella muuttamaan esimerkiksi sivuston kaikkien h1-elementtien väri. Sivuston sisällön tulisi olla saatavilla myös ilman tyylien käyttämistä. Tyylitiedosto voidaan tehdä sisäiseksi tai ulkoiseksi. Sisäinen tyylitiedosto on dokumentin head-osassa Zeldmanin (2003: 228) mukaan esimerkiksi näin:

```
<style type="text/css">
<!--
body    {
        color: #000000;
    }
-->
</style>
```

Sisäistä tyylitiedostoa ei kuitenkaan suositella käytettävän, koska siinä menetetään tyylitiedoston suurimmat hyödyt: muutokset eivät vaikuta kerralla kaikkeen sekä tiedostojen koot ovat suuria, koska käyttäjä joutuu lataamaan tyylit jokaiselle sivulle erikseen. Ulkoisessa tyylitiedostossa tyylit luodaan erilliseen css-päätteiseen tiedostoon, joka linkitetään jokaisen dokumentin head-osioon. Esimerkki linkityksessä tyylitiedoston nimi on tyylit.css. (Zeldman 2003: 228-230)

```
<link href="tyylit.css" rel="stylesheet" type="text/css" media="all" />
```

Sivustolle voidaan tuoda tyylejä useammasta kuin yhdestä tyylitiedostosta. Tämä tehdään esimerkiksi @import-määrittelyn avulla. Myöhemmin määriteltä tyylit syrjäyttää vastaavat aiemmin määritellyt tyylit, joten @import-määrittely suositellaan lisättävän ennen muita tyylin määrittelyjä. @import tyylejä voidaan käyttää esimerkiksi, jos sivustolle halutaan määrittellä erilaisia tyylejä eri selaimille. (Zeldman 2003: 231, Linjama 2001: 512)

```
<style type="text/css" media="all">@import "tyylit2.css";
</style>
```

3.1 Tyylien määrittäminen

Tyylit määritellään tyylitiedostossa sen HTML-elementin nimellä, johon tyylimäärittelyt vaikuttavat. Tyylin nimen jälkeen kerrotaan aaltosulkeiden ({ ja }) välissä tyylin ominaisuudet. Ominaisuuden nimen ja arvon väliin tulee kaksoispiste (:) ja arvon perään laitetaan pilkku (;). (Linjama 2001: 504-505)

```
body{
    font-family: Times, "Times New Roman";
    margin: 0px;
    background-color: #FDF6E7;
}
```

Esimerkissä body elementille määritellään fontti, marginaali sekä taustaväri. Useampisanaiset fonttien nimet kirjoitetaan lainausmerkkien sisään.

Tyylejä, joilla on samoja ominaisuuksia, voidaan ryhmitellä. Esimerkiksi otsikko tyyleille h1, h2 ja h3 voidaan määrittää yhtenäinen väri. (Linjama 2001: 514)

```
h1, h2, h3 {
    color: #000000;
}
```

Tyyleille voidaan määrittellä myös aliluokkia. Näiden avulla yhdelle HTML-elementille voidaan määrittellä erilaisia esitystapoja. Sivustolla voi olla esimerkiksi h1-otsikko, jota halutaan korostaa erivärillä kuin sivuston muita h1-otsikoita. Aliluokka määritellään antamalla tyylille yhdistelmänimi, joka koostuu tyylin alkuperäisestä ja aliluokan nimestä. Nimet erotetaan toisistaan pisteellä (.). (Linjama 2001: 517)

```
h1 {
    text-align: center;
    color: #000000;
}

h1.korostus {
    color: #BB5E18;
}
```

Mikäli aliluokkaa ei haluta yhdistää tiettyyn tyyliin, siitä voidaan tehdä myös erillinen luokka, jota käytetään minkä tahansa elementin yhteydessä. (Linjama 2001: 518)

```
.korostus {
    color: #BB5E18;
}
```

Dokumentissa tyylien aliluokka tai luokka otetaan käyttöön class-attribuutilla. Jos tyylin korostus on omana luokkana, eikä aliluokkana, sitä voitaisiin käyttää samaan tapaan myös millä tahansa elementillä. (Linjama 2001: 518-519)

```
<h1 class="korostus">Lorem Ipsum</h1>
```

Class-attribuutin sijasta voidaan käyttää myös id-attribuuttia. Id-attribuutilla yksilöidään luokkia. Id-luokka saa löytyä vain kerran samasta dokumentista. Tällöin tyyli tiedostossa pisteen tilalla käytetään #-merkkiä ja dokumentissa class-sanan tilalla on sana id. (Linjama 2001: 519-520)

```
#korostus {
    color: #BB5E18;
}

<h1 id="korostus">Lorem Ipsum</h1>
```

3.2 Div-elementti

Alun perin div-elementti on tarkoitettu korvaamaan center-elementti, mutta tyyplejä käytettäessä siitä on tullut yksi tärkeimmistä elementeistä ja sen käyttö on laajentunut. Se on asemointilohkon muodostava elementti, jolla rajataan ne HTML-elementit, joihin tietty toiminto halutaan kohdistaa. Div-elementillä rajataan esitettävä lohko, ja lohkolle voidaan määritellä erilaisia arvoja, kuten sen sijainti selainikkunassa, taustan ulkoasu jne. Sivuston ulkoasun asettelussa asemoitavat kappaleet saadaan div-elementin avulla ”irrotettua” omiksi kappaleikseen. Tällä tavoin sivuston ulkoasun rakenteen tekeminen on helppoa. Tyyli tiedostoon voidaan määritellä erikseen esimerkiksi navigaatiopalkki (navlist), vasen reuna (leftnav) ja tekstialue (content). Jokaiselle lohkolle annetaan halutut arvot ja lohkot otetaan käyttöön dokumentissa div-elementin avulla. (Linjama 2001: 202-203)

```
#leftnav {
    width: 170px;
    background-color: FFF9ED;
    float: left;
}

#content {
    width: 470px;
    margin-left: 180px;
}

.navlist{
    width: 700px;
    color: #10558C;
    background-color: #FDB882;
}
```

Esimerkissä vasemmalle reunalohkolle on määritetty leveys ja taustaväri. Float-ominaisuus määrittää lohkon paikan ”kellumaan” dokumentin vasemmassa reunassa. Tekstialue-lohkolle on määritetty leveys ja vasen marginaali. Vasenreuna- ja tekstilohkossa käytetään id-attribuuttia, koska ne esiintyvät vain kerran samassa dokumentissa. Navigaatiopalkille on määritetty leveys, väri ja taustaväri. Lohkon määrittämiseen käytetään class-attribuuttia.

Linjaman (2001: 204) mukaan tyyliä otetaan käyttöön dokumentissa div-elementillä.

```
<div class="navlist">
<a href="index.html">Index</a>
<a href="koulutus.html">Koulutus</a>
</div>

<div id="leftnav">
<p>Tampereen kesäyliopisto järjestää avointa yliopisto-
opetusta.</p>
</div>

<div id="content">
<h1> Lorem ipsum</h1>
<p>Kesäyliopiston koulutus on avointa kaikille iästä ja ai-
kaisemmasta koulutuksesta riippumatta.</p>
</div>
```

3.3 Pseudoluokat

Tyyliluokkien erikoistapauksia ovat näennäis- eli pseudoluokat. Näillä voidaan elementeille antaa ominaisuuksia, joita ei ole mahdollista toteuttaa tyylien avulla. Esimerkiksi linkkien värin muuttaminen vieraillemattoman ja vierailun linkin välillä voidaan toteuttaa tällä tavalla. Näissä tapauksissa tyylin nimi koostuu tyylin nimestä ja näennäisluokan nimestä. Nämä erotetaan toisistaan kaksoispisteellä (:). (Linjama 2001: 526-527)

```
a:link{
  text-decoration: none;
  color: #10558C;
}

a:visited {
  text-decoration: none;
  color: #10558C;
}

a:active {
  text-decoration: underline;
  color: #F47B2D;
}

a:hover {
  text-decoration: underline;
  color: #F47B2D;
}
```

Linkeille on määritetty alleviivaus/alleviivaamattomuus (underline/none) sekä väri.

MofiWiki - kotimaisen Mozilla-yhteisön wikipedia (2006) sivustolla on määritelty näennäisluokkia seuraavasti:

link: peruslinkkiteksti, kun mikään muu pseudoluokka ei ole voimassa,

visited: sivu, jolla on vierailtu aiemmin,

active: aktiivinen linkki eli linkki, joka on valittuna,

hover: kun hiiren kursori on linkin päällä.

3.4 Tyylien arvojärjestys

Kaikkia elementtejä ei voida laittaa minkä tahansa elementtien sisälle, vaan HTML-elementeillä on tietty arvojärjestys. Elementtejä on pääosin kahdenlaisia: lohko-tyyppisiä ja sisä-tyyppisiä. Lohko-elementeillä on aina lopettava tagi. Sisä-elementit voivat olla sekä yksitagsisia että alku- ja lopputagillisia.

Lohko-elementtejä ovat esimerkiksi:

```
<div>: ilman erikoismerkitystä oleva peruselementti
<h1>, <h2>, <h3>, <h4>, <h5> ja <h6>: otsikkoelementit
<p>: kappale-elementti
<form>: lomake-elementti
```

Sisä-elementtejä ovat esimerkiksi:

```
<a>: linkkielementti
<img>: kuvaelementti
<span>: ilman erikoismerkitystä oleva peruselementti
<strong>: korostus/lihavointi
```

Lohko-tason elementtejä ei voida sijoittaa sisä-elementtien sisälle. Näistä div- sekä form-elementit ovat korkeimmassa asemassa, ja niiden alle voidaan sijoittaa muita elementtejä. Otsikko- ja kappale-elementeistä, otsikkoelementit ovat korkeammassa asemassa. (MofiWiki - kotimaisen Mozilla-yhteisön wikipedia 2006)

Miksi kannattaa käyttää XHTML/CSS -ratkaisuja

Mukaiillen Zeldmania (2003: 150-151) ja sivustoa Why tables for layout is stupid (2003):

1. XHTML on tämänhetkinen standardi.
2. XHTML on suunniteltu toimimaan yhteistyössä muiden XML-kielien kanssa.
3. Sekä vanhat, että uudet selaimet toimivat XHTML-sivustoilla.
4. XHTML toimii kaikenlaisilla laitteilla, se on laitteistoriippumaton kieltä.
5. Säästää kaistaa, sillä sivut ovat kooltaan pienempiä → Sivut latautuvat nopeammin
6. Tyylitiedosto ja sivun ulkoasuun liittyvä grafiikka on erillään ja säilyvät selaimen välimuistissa.
7. Päivitettävyyden merkittävästi helpompaa ja nopeampaa → myös kustannussäästö.
8. Antaa aiempia parempia hakutuloksia.

4 Palvelinsovellusten ohjelmointia

ASP, Java ja PHP ovat esimerkkejä palvelinsovellusten ohjelmointiin käytettävistä kielistä. Sovellukset toimivat WWW-palvelimella tai sovelluspalvelimella. Esimerkiksi verkkokaupan tai keskusteluryhmän toimimiseen tarvitaan palvelinsovelluksia. (Keränen ym. 2006: 34)

4.1 PHP

PHP:n ensimmäinen versio, nimeltään Personal Home Pages, julkaistiin 1994. Silloin se tosin oli vain joukko Web-julkaisumakroja. Vuonna 1997 kieltä alettiin kehittää ja tuloksena syntyi PHP3. Tämän jälkeen PHP:n suosio alkoi kasvaa huomasti. Vuoden 1999 marraskuussa PHP:tä käytettiin yli miljoonassa palvelimessa ja helmikuussa 2000, luku oli jo 1,4 miljoonaa palvelinta. Nykyään PHP tunnetaan nimellä HyperText Preprocessor. (Zandstra 2001: 20)

PHP on palvelinpuolen skriptikieli, jota käytetään tavallisimmin HTML/XHTML:n yhteydessä. PHP-skripti jäsenellään PHP-ohjelman tai -moduulin toimesta. Kun PHP koodi on tulkattu ja suoritettu, sen tulostus yhdistetään skriptissä olevaan HTML-koodiin ja tulos lähetetään käyttäjälle. PHP on avoin ohjelmisto ja sen lähdekoodi on Internetissä vapaasti kaikkien saatavilla ilmaiseksi. Se on myös pitkälti alustariippumaton ja sen voi asentaa esimerkiksi Windows-, Unix-, Linux- ja Mac-alustoilla. PHP on kehitetty toimimaan yhdessä tietokantojen ja XML-tiedostojen kanssa. (Zandstra 2001: 20, 2kmediat.com - PHP n.d.)

PHP muistuttaa syntaksiltaan muun muassa C, Java ja Perl-kieliä, mutta sillä on myös täysiä omia piirteitään. PHP-koodia sisältävä dokumentti tallennetaan .php-päätteiseksi, jotta palvelin ymmärtää käsitellä tiedostoa PHP-koodina. Muita vaihtoehtoja ovat esimerkiksi .php3 ja .php4. Dokumentissa oleva PHP-koodi erotetaan muusta koodista PHP-tageilla. Vaihtoehtoja on neljä erilaista, mutta standarditagin käyttö on suosittelavin käytötapa. (Zandstra 2001: 36-37)

Esimerkki PHP:n standardi-tagin käytöstä. Koodi tulostaa selaimelle tekstin Hello world.

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

PHP:ssä on paljon funktioita, eikä niitä kaikkia kannata tai edes voi opetella ulkoa. Funktiot löytyvät PHP:n kotisivulta, osoitteesta www.php.net. Hyödyllisiä funktioita WWW-sivuston rakennetta ajatellen ovat esimerkiksi `include_once`- ja `require_once`-funktiot. Näiden funktioiden avulla voidaan PHP-dokumentteihin liittää tiedostoja. Funktioiden ero on siinä, että liitettävien tiedostojen puuttuessa, `include_once`:ssa, selain antaa varoituksen, mutta jatkaa sivun näyttämistä, `require_once`:ssa taas, selain antaa virheilmoituksen ja keskeyttää sivun esittämisen. Tiedostojen liittäminen on järkevää, jos sivuston eri sivuilla toistuu samoja asioita, kuten navigointi tai alatunniste.

Liitettävä tiedosto kirjoitetaan omaan tiedostoonsa ja se voidaan tallentaa esimerkiksi `.html`- tai `.php`-päätteiseksi. Tiedosto liitetään sivustolle haluttuun kohtaan Zandstran (2001: 174) mukaan esimerkiksi seuraavasti:

```
<?php include_once("navigointi.php"); ?>
```

4.3 SSI

SSI eli Server Side Includes on yksi vanhimmista palvelinpuolen ohjelmoinnissa käytettävistä tekniikoista. Sen avulla voidaan esimerkiksi lisätä sivuille toistuvia koodinpalasia sekä noutaa perustietoja palvelimen toiminnasta. SSI-tekniikka on menetelmänä ilmainen ja suhteellisen yksinkertainen. Useimmat palveluntarjoajat tukevat sitä oletusarvoisesti, ja useimmissa tapauksissa tällöin tiedostojen päätteenä on `.stm`, `.shtm` tai `.shtml`. (2kmediat.com - SSI n.d.)

SSI:n perustoiminto, sivujen sisällyttäminen (`include`), tehdään Ohjelmointiputka - SSI -sivuston (2003) mukaan seuraavasti. Teksti lisätään HTML-koodin sekaan haluttuun kohtaan esimerkiksi näin:

```
<!--#include virtual="ylaosa.html" -->
```

5 WWW-sivuilla käytettäviä tekniikoita

5.1 Tietokannat

Tietokanta on tehokas tapa tallentaa tietoa. Monen verkkopalvelun taustalla toimii tietokanta. Tietokanta koostuu tauluista, jotka sisältävät tietueita. Tietokantojen ylläpitämiseksi on olemassa tietokannan hallintajärjestelmiä, joiden avulla esimerkiksi luodaan tietokantoja ja suoritetaan hakuja. Hallintajärjestelmiä ovat muun muassa Oracle ja MySQL. Järjestelmiä varten hankitaan yleensä erillinen palvelin. (Keränen ym. 2006: 37)

SQL-kielillä tietokantaan voidaan luoda uusi taulu alla olevalla tavalla. Esimerkissä luodaan taulu person, jonka kentät ovat name ja address. (Ohjelmointiputka - PHP 2003)

```
CREATE TABLE person
(
  name          VARCHAR(80)
  ,address      VARCHAR(80)
);
```

Person-tauluun voidaan manuaalisesti tallettaa tietoa seuraavalla tavalla:

```
INSERT INTO person
(name, address)
VALUES
("Matti Mallikas", "Mallikkaantie 56")
;
```

SQL-kielessä kysely muodostuu select- ja from-lauseista. Kyselyn tarkentamiseen voi liittyä myös where-, and- ja or- sekä order by -lauseita. (Keränen ym. 2006: 38)

5.2 JavaScript

Vuonna 1995 Netscapen Navigator2-selaimen mukana julkaistiin JavaScript 1.0-kieli. Tämä mahdollisti ensimmäistä kertaa interaktiivisten HTML-dokumenttien tuottamisen. JavaScript-kieltä voidaan käyttää esimerkiksi lomakekenttien tarkastamiseen tai selainikkunan avaamiseen. JavaScript-ohjelma toteutetaan aina selaimessa tai muussa JavaScript-tulkin sisältävässä sovelluksessa. Se on siis selainpohjainen ohjelmointikieli. Kieltä suoritetaan rivi riviltä sitä mukaan kuin skriptiä luetaan. (Linjama 2001: 604)

JavaScript-koodi kirjoitetaan HTML-tiedostoon tai ulkoiseen JavaScript-tiedostoon. Seuraava Keräsen ym. (2006: 33-34) esimerkki avaa uuden ikkunan selainikkunaan:

```
<script language="JavaScript">
function UusiIkkuna(){
window.open('index.html', 'ikkuna', 'width=800, height=600');
}
</script>
```

JavaScript-koodia ei tulisi käyttää WWW-sivuilla toimintoihin, jotka ovat sivuston käytön kannalta oleellisia. JavaScript voidaan ottaa pois käytöstä ja siksi sitä pitäisi käyttää esimerkiksi lomakkeiden painikkeiden täydentämiseen, ei kokonaan korvaamiseen. (Korpela ja Linjama 2004: 116)

5.3 Julkaisuhjelmat

Sivuston ylläpitämiseksi on tarjolla paljon erilaisia valmiita julkaisuohjelmia. Julkaisuohjelmilla sivujen päivittäminen on helppoa suoraan selaimen kautta. Useimmat tarjolla olevat ratkaisut toimivat kuukausimaksulla ja ohjelman käyttäminen vaatii sivuston luomista kyseessä olevan ohjelman kautta. Poikkeuksiakin tosin löytyy, ja näistä yksi on esimerkiksi Joomla!.

Suominen (18.8.2006 haastattelu) vastasi Joomla!:an liittyviin kysymyksiini sähköpostiviestissä. Tässä poimintoja vastauksista: Joomla! on ilmainen, avoimen lähdekoodin ohjelma. Joomla!:n räätälöinti, muokkaaminen, laajentaminen, kouluttaminen ja konsultointi ovat kaupallista toimintaa ja tällä tavoin ilmaisen perustuotteen ympärille on syntynyt oma teollisuudenala. Joomla! on monipuolinen ja pelkällä Internet-selaimella käytettävä julkaisujärjestelmä.

Joomla!:a voi käyttää kaupalliseen toimintaan ja se soveltuu suurten yritysten kuin myös yksityisten kotisivujen ylläpitämiseen. Käyttäminen on ilmaista. Joomla! toimii web-palvelimella (yleensä Apache) ja vaatii MySQL-tietokannan sekä PHP-ohjelmointikielen tuen.

6 Nykyinen WWW-sivusto

Tampereen kesäyliopiston nykyisen sivuston on tehnyt Tampereen ammattikorkeakoulun hypermedian opiskelija vuonna 2004 osana harjoitteluaan kesäyliopistossa. Kesäyliopiston henkilökunta käyttää sivustoa päivittäin työssään ja sen sisältö on pyritty pitämään mahdollisimman ajantasaisena.

6.1 Tekniset ongelmat

Kesäyliopiston nykyisen sivuston koodi on sekavaa, eikä se vastaa enää alkuperäistä. Sivusto on tehty HTML-kielellä, käyttäen SSI (Server Side Includes) -tekniikkaa. SSI-tekniikalla voidaan esimerkiksi lisätä sivuille toistuvia osoita, kuten navigaatio. Usealla sivulla ei kuitenkaan enää toteuteta SSI-komentoja. Esimerkiksi navigointia ei tuoda sivulle Include-komennolla, vaan sivu sisältää koodin alusta loppuun.

Sivujen ulkoasua on määritelty koodissa ja tämä on edesauttanut virheiden syntymistä myös ulkoasuun. Sivusto on rakennettu taulukkotaitolla. Esimerkiksi Why tables for layout is stupid (2003) -sivustolla on kerrottu perustellusti, miksi taulukoita ei kannattaisi käyttää sivun taitossa. Taulukoiden käyttö on yhä edelleen paljon käytetty keino visuaalisesti raskaiden sivustojen toteutuksessa, vaikka sitä ei suositellakaan. Taulukot on HTML-kielessä tarkoitettu taulukkomuotoisen tiedon esittämiseen, mutta `border="0"` on mahdollistanut taulukoiden käytön kehikkona kuville ja tekstile. Taulukkopohjaiset sivut ovat vaikeasti käytettävissä erikoisryhmille tai kannettavien päätelaitteiden käyttäjille. Sivuston uudelleen suunnittelu ja päivitys ovat myös työlästä taulukkopohjaisilla sivustoilla, koska muutokset täytyy tehdä jokaiselle sivulle erikseen. Käyttäjät joutuvat lataamaan ulkoasu jokaisella sivulla jolla käyvät, koska ulkoasu on sisällytetty sivun koodiin.

Tyylitiedostot, jotka hallitsevat ulkoasua, jäävät käyttäjien selainten väliin muistin talteen ja sivut latautuvat nopeammin. Sivujen tiedostokoot pienentyvät, kun käytetään standardeja. Visuaalisen ilmeen säilyttäminen on myös helppoa tyylitiedoston avulla, koska muutos tyylitiedostoon vaikuttaa kaikkiin sivuihin samalla kertaa. Taulukoiden käyttö on perusteltua joissakin tilanteissa, kuten erityisesti taulukkotiedon esittämisessä, mihin ne onkin tarkoitettu sekä joskus myös ulkoasuun liittyvissä seikoissa. Mutta koko sivun asetteluun sitä ei suositella.

Nykyisellä sivustolla otsikot on tehty kuvina, eikä kuville ole määritetty alt-attribuuttia. Jos sivustoa katsoo ilman kuvia, käyttäjä ei näe otsikoita lainkaan. Kuvien käyttäminen sivuston kannalta oleellisissa paikoissa ei ole järkevää, koska sivuja saatetaan lukea laitteelta, missä kuvia ei voida tai haluta esittää. Lisäksi uusien otsikoiden tekeminen on työlästä, koska aina joudutaan tekemään uusi kuva.

6.2 Ulkoasu

Sivuston visuaalinen ulkoasu ei vastaa esimerkiksi kesäyliopiston kurssikirjeitä tai markkinointiesitteitä. Kurssikirje/todistuspaperissa on kesäyliopiston logon värimaailma. Nykyisellä sivustolla väreinä on sininen ja valkoinen. Kesäyliopiston logo löytyy vain etusivulta, sijoitettuna sivun oikeaan alareunaan. Leipätekstin ja linkkien värinä on sama tummansininen. Kuva 1 havainnoi nykyisen etusivun ilmettä.

Tampereen kesäyliopisto
- oppia ja osaamista 50 vuotta

In English

Etusivulle

Aluekehittäjän koulutusohjelmat Pirkanmaan tarpeisiin

Finnish for Foreigners

Hallinto

Henkilökunta

Ilmoittautuminen

Koulutus

Linkit

Lukio-opiskelijoille

Opinto-ohjaus

Palautte

Peruutuskäytännöt

Pro Pirkanmaa korkeakoulujen palvelupisteet

Seminaarit

Sivukartta

Tenitit

Tutorpankki

Yhteystiedot

Tampereen kesäyliopisto järjestää avointa yliopisto-opetusta, ammatillista täydennyskoulutusta sekä kieli- ja abikursseja ympäri vuoden. Toiminta-alueenamme on koko Pirkanmaa. Kesäyliopiston koulutus on avointa kaikille iästä ja aikaisemmasta koulutuksesta riippumatta.

Ajankohtaista

- Tuomiokirjat tutuksi 1.11.-22.11.
- Photoshop ja Web 7.11.-14.11.
- Venäjän kielen jatkokurssi 8.11.2006-28.3.2007
- Asiakaspalvelun henkilöturvallisuus 9.11.
- Kehityskeskustelut johtamisen tukena 23.11.
- Äänenhuollon peruspaketti 25.11. alkaen
- Kaakkois-Pirkanmaan seutuseminaari 6.11.
- Luoteis-Pirkanmaan seutuseminaari 8.11.
- Lounais-Pirkanmaan seutuseminaari 10.11.
- Ylä-Pirkanmaa seutuseminaari 17.11.
- Etelä-Pirkanmaan seutuseminaari 21.11.
- Syksyllä 2006 alkaneet avoimen yliopiston opinnot

Lapset, tunteet ja tietoyhteiskunta -seminaari 8.-9.2.2007

Toimiva lapsi & perhe -koulutus, alk. 15.1.2007

Opetushallituksen rahoittamaa koulutusta:

- Akateemisten maahanmuuttajien kieli- ja tietoyhteiskuntakoulutus

Työvoimapolitiittista koulutusta:

- Aineenopettajan pedagogiset täydennysopinnot maahanmuuttajataustaisille hakijoille 2006 - 2007
- Kielipainotteinen Venäjän kaupan ja matkailun koulutusohjelma
- Mainonta ja markkinointi tietoverkossa

TAMPEREEN
kesä
yliopisto

Kuva 1 Tampereen kesäyliopiston etusivu, www.uta.fi/kesayliopisto, 4.11.2006

6.3 Käytettävyys ja päivitys

Käytettävyyden kannalta suurimpia ongelmia ovat navigaatio sekä tiedon vaikea löydettävyys sivustolta. Tästä on tullut asiakkaiden suunnalta palautetta kesäyliopistolle sekä myös henkilökunta nostaa tämän esiin yhtenä suurimpana ongelmakohtana.

Sivuston navigaatio on rakennettu sivun vasempaan reunaan. Linkkejä on paljon ja ne on järjestetty aakkosjärjestykseen. Tällä tavalla kesäyliopiston kannalta tärkeimmät päälinkit, kuten esimerkiksi *koulutus*, jonka takaa kaikki kurssitarjonta löytyy, tippuu navigaatiolistassa liian alas. Myös esimerkiksi *ilmoittautuminen* ja *tentit* pitäisi löytyä heti ensisilmäyksellä. Neljä ylimmäistä linkkiä, *Aluekehittäjien koulutusohjelmat Pirkanmaan tarpeisiin*, *Finnish for Foreigners*, *Hallinto* ja *Henkilökunta* eivät ole asioita, joita valtaosa käyttäjistä etsii. Lisäksi navigaation alla on sininen taustakuva, joka vaikeuttaa linkkien hahmottamista entisestään, sillä myös linkkien väri on sininen. Joillakin sivuilla tekstiosuus on lyhyempi kuin navigaatiolista, ja siten myös tärkeimpiä linkkejä saattaa jäädä huomaamatta. Useimmilla näytöillä navigaatio ei näy kokonaisuudessaan sivulla, vaan sivua täytyy vierittää alaspäin.

Suurin osa henkilökunnasta kokee päivittämisen ongelmallisena. Kaikki päivitykset tehdään Dreamweaver-ohjelman avulla, jonka jälkeen työkooneelle tallennettu tiedosto siirretään WinSCP-SFTP-ohjelmalla yliopiston palvelimelle. Esimerkiksi uusien ammatillisten täydennyskoulutuksien päivittäminen on suhteellisen työlästä. Ensimmäiseksi kurssikuvaus luodaan oikealle sivulle, kuten esimerkiksi *Opetus ja kasvatus*. Tämän jälkeen kurssinimi kirjoitetaan *Opetus ja kasvatus* -sivun yläreunaan linkkinä ja linkitetään se kurssikuvaukseen. Sama linkki liitetään myös *Ammatillinen täydennyskoulutus* -sivustolle *Opetus ja kasvatus* -otsikon alle. Jos kurssi alkaa lähiviikkoina tai kuukausina, lisätään linkki vielä etusivun ajankohtaista -palstalle. Yhden kurssikuvauksen päivittämiseksi täytyy siis päivittää kolme eri sivua. Kun kurssi on loppunut, kurssi pitäisi muistaa poistaa ainakin etusivulta ja myöhemmin myös näiltä kahdelta muulta sivulta.

Avoin yliopisto opetuskurssien kurssikuvaukset on poimittu suoraan Avoinen yliopiston Meteoritietokannasta, mutta nekin linkitetään sivustolle erikseen. Linkki kirjoitetaan *Avoin yliopisto* -sivulle sekä pian alkavat kurssit etusivun *ajankohtaista*-laatikkoon. Tässäkin tapauksessa täytyy siis päivittää kahta eri sivua, vaikka kurssien kuvaukset linkitetään Meteori-tietokannasta.

7. Uuden sivuston mahdollisuudet ja rajoitukset

Uuden sivuston suunnittelussa täytyy toiveiden lisäksi huomioida mahdolliset rajoitukset, esimerkiksi palvelin tai budjetti voivat aiheuttaa rajoituksia. Tein kyselyn Suomen kesäyliopistoille heidän sivustoistaan sekä selvitin Tampereen yliopiston palvelimen sekä kesäyliopiston henkilökunnan toiveet uuden sivuston suhteen.

7.1 Ratkaisuja Suomen muissa kesäyliopistoissa

Suomessa on 21 kesäyliopistoa. Lähetin 21.8.2006 kaikkiin Suomen kesäyliopistoihin, lukuun ottamatta Ahvenenmaata ja Tamperetta, sähköpostitse kyselyn heidän WWW-sivustostaan. Kysyin muun muassa miten ja milloin sivut on tehty, monennetko ne ovat, miten ja kuka niitä päivittää, paljonko ne ovat tulleet maksamaan ja koetaanko sivut helppokäyttöisiksi. Tarkoitus oli kartoittaa, miten muut samaa toimialaa harjoittavat yksiköt ovat ratkaisseet WWW-sivuihin liittyviä ongelmia.

Lähetin kyselyn uudestaan 19.9.2006 ja lopulta sain yhteensä kahdeksan vastausta. Vastausprosentti oli 42,10 %. Syynä alhaiseen vastausprosenttiin saattoi olla esimerkiksi ajankohta. Syksy on kiireistä aikaa kesäyliopistoilla. Toinen syy voi olla se, että sähköposti on mennyt kesäyliopiston yleiseen osoitteeseen, eikä se ole tavoittanut oikeaa henkilöä.

Vastanneiden sivuista kahdet oli tehty vuonna 2002, neljät vuonna 2004 ja yhdet vuonna 2003 ja 2006. Vain yksi ilmoitti, että he ovat uusimassa sivujaan lähiaikoina. Useimmat vastanneista olivat tyytyväisiä omiin sivuihinsa ja kokivat, että sivujen päivittäminen on helppoa ja vaivatonta. Päivityksestä vastasi useimmiten joku tietty vakituinen työntekijä.

Vastanneista kuusi kesäyliopistoa on tilannut sivut ulkopuoliselta taholta ja kaksi tehnyt ne itse FrontPage-ohjelmalla. Tilatuista sivuista on maksettu 1300-5000 euroa. Kysymykset ja vastaukset löytyvät liitteestä 1.

Vaikka sain vastauksia vähän, oli niistä kuitenkin hyötyä mietittäessä Tampereen kesäyliopiston WWW-sivujen päivitysongelmia. Useassa kesäyliopistossa, kuten Tampereellakin, käytetään kurssinhallintaohjelma Kuhaa. Vastanneista, kahdessa kesäyliopistossa on käytössä päivitystyökalu Kuhan ja WWW-sivujen välillä. Myös avoimen yliopiston tietokannan käytöstä sain uusia näkökulmia kyselyn avulla.

7.2 Tampereen yliopiston palvelin

Tampereen kesäyliopiston sivut sijaitsevat tällä hetkellä Tampereen yliopiston palvelimella osoitteessa www.uta.fi/kesayliopisto. Myös uudet sivut halutaan säilyttää yliopiston palvelimella. Tämä on johdon tahto. Sivujen säilyttämisen merkitys on historiallisen yhteyden säilyminen yliopistoon. Johto perustelee tätä myös siten, että kun kotisivujen osoite on www.uta.fi, se on imago- ja uskottavuuskysymys.

Käytännössä yliopiston palvelin asettaa merkittäviä rajoituksia uudelle sivustolle, esimerkiksi minkäänlaisia tietokantaratkaisuja päivittämisessä tai sivujen rakenteessa ei voida käyttää. Lähetin elokuussa yliopiston webmasterille kyselyn palvelimen rajoituksista/vaatimuksista. Sain ensimmäisen vastauksen 23.8.2006 ja toisen 11.9.2006.

Tampereen kesäyliopisto ei ole Tampereen yliopiston yksikkö, eikä hallinnollisesti oikeastaan mitenkään yhteydessä Tampereen yliopistoon. Yliopiston tietokonekeskus ei anna ilman erillistä sopimusta tukea kesäyliopiston WWW-toiminnoille. Tampereen yliopiston hallitus on vahvistanut kokouksessaan 18.6.1999 linjaukset WWW:n käytön ja kehittämisen perustaksi Tampereen yliopistossa. Yliopiston WWW-politiikka määrittelee yleisohjeen myös kaikille yliopiston palvelimella sijaitseville sivustoille. Siinä on kielletty mm. mainostaminen ja muu kaupallinen toiminta.

Tampereen yliopiston WWW-politiikka (2001) määrittelee sivustoille muun muassa seuraavia ohjeita:

”Sivujen laadinnassa ja ylläpidossa on otettava huomioon lainsäädäntö, kuten henkilötietolaki ja tekijänoikeussäännökset sekä yliopiston tietokonejärjestelmien ja tietoliikenneverkon käytösäännöt ja rajoitukset esimerkiksi talletustilan käytöstä. Sivujen aineiston tulee olla sellaista, että se ei loukkaa yksityisyyden suojaa eikä muullakaan tavalla loukkaa kolmatta osapuolta.”

”Mikäli jokin sivu ei ole yliopiston WWW-politiikan (ja sen nojalla annettujen ohjeiden) mukainen, huomautetaan asiasta sivusta vastaavalle. Jos huomautus ei johda toivottuihin tuloksiin sovittu ajan kuluessa, poistetaan kyseinen sivu. Mikäli WWW-palvelin ei toimi politiikan mukaisesti, voidaan se poistaa yliopiston tietoliikenneverkosta.”

Yliopiston palvelimella on käytössä PHP, ja tämänhetkinen (11.9.2006) PHP:n versio on 4.3.10.

7.3 Tampereen kesäyliopiston tavoitteita uudelle sivustolle

Lähetin 28.9.2006 kyselyn sähköpostitse kesäyliopiston henkilökunnalle. Kysyin muun muassa, mitä mieltä vastaaja on nykyisestä sivustosta esimerkiksi rakenteen ja päivittämisen suhteen, mitä huonoa ja hyvää sivustolla on ja mitä erityisesti tulisi huomioida uusia sivuja suunniteltaessa, sekä pystyykö hän itse päivittämään sivustoa ja, jos ei, miten se pitäisi olla hoidettuna.

Henkilökunta pitää sivuston rakennetta ongelmallisena. Päivitys Dreamweaverin kautta ei onnistu kaikilta, eikä ohjelman opettelemiseen ole aikaa tai edes kiinnostusta. Esiin tuli myös maininta, että sivuston visuaalinen ulkonäkö kärsii, jos päivittäjiä on useampi. Tähän saadaan ratkaisu, kun siirrytään pois taulukkoasettelusta ja viedään kaikki ulkoasuun liittyvä erilliseen tyylitiedostoon.

Hyvinä asioina vanhalla sivustolla pidetään kurssien jaottelua omille sivuilleen Avoimessa yliopisto opetuksessa sekä Ammatillisessa täydennyskoulutuksessa. Myös etusivun toivotaan edelleen olevan informatiivinen sivu, josta löytyy tiivistetysti ajankohtaiset asiat tai ainakin selkeästi ohjataan siitä eteenpäin.

Kysyin myös, kuinka tärkeänä WWW-sivut koetaan ja tässä siteeraus saamastani vastauksesta: *"On erittäin tärkeä asia. Asiakaspalvelu siirtyy yhä enemmän verkkoon ja ihmiset haluavat löytää tiedon nopeasti verkosta. Tällöin tietoa on mahdollista saada milloin vaan eikä tarvitse yrittää soitella toimisto aikaan informaation saamiseksi."* (19.10.2006 haastattelu)

Uusien sivujen suunnittelussa keskitytään erityisesti sivuston rakenteeseen sekä päivittämisen helpottamiseen. Näiden asioiden parantamisen toivotaan helpottavan tiedon löytymistä sekä sen ajankohtaisena pysymistä. Työtä tehdessäni, olen yhteydessä kesäyliopiston yhteyshenkilöni Sanna Vaurulaan.

7.4 Laatukriteerit

Laatua Verkkoon -sivustolla (Suomi.fi – Laatua verkkoon, 2006) on listattu WWW-sivujen laatukriteereitä. Näiden tarkoituksena on parantaa verkkopalvelujen laatua erityisesti palvelun käyttäjien näkökulmasta, lisätä asiakastyytyvyyttä ja sitä kautta verkkopalvelujen käyttöä. Laatukriteerit on tarkoitettu julkisten verkkopalvelujen arvioinnin ja kehittämisen välineeksi. Ne soveltuvat pääosin myös yritysten tai muiden kaupallisten verkkopalvelujen arviointiin.

Laatukriteereissä listataan muun muassa sivuston rakennetta ja sisältöä koskevia asioita. Seuraavassa mukaillen ja poimien Laatua verkkoon sivuston ohjeistuksia (Suomi.fi – Laatua verkkoon, 2006):

Rakenteen tulee olla selkeä ja visuaalisen ilmeen avulla palvelu pitää pystyä tunnistamaan. Vakiotoiminnoilla (haku, navigaatio), sisältöalueilla ja muilla elementeillä on vakiopaikat. Asiakokonaisuuksissa ja otsikoissa ei ole perusteetonta toistoa.

Käyttöliittymän tyylin, asettelun, värien ja kirjasintyyppien koot ja toiminnot ovat selkeitä ja yhtenäisiä. Eri sivuilla tulee olla yhtenäinen ilme ja väriellä ilmaistu asia välittyy myös muulla tavoin. Värisokeus, joista tavallisin on punavihersokeus, on otettu huomioon värien suunnittelussa.

Navigaation laatukriteereitä ovat muun muassa seuraavat:

1. Linkit ovat kuvaavia ja toimivat sekä ovat ajan tasalla.
2. Linkit ovat kuvaavia ja osoittavat käyttäjälle, minne ne johdattavat.
3. Tekstin joukossa olevia linkkejä on hyvä välttää.
4. Erialaisten linkkien käyttö on ohjeistettu. Esimerkiksi jos linkki avaa uuden selainikkunan, siitä ilmoitetaan käyttäjälle.
5. Linkin nimen ja kohdesivun otsikon avainsanat ovat samoja.
6. Käyttäjä näkee linkistä, onko hän käyttänyt sitä.

Tekstin ja taustan väreissä on riittävä tummuuskontrasti, taustakuva ei häiritse tekstin lukemista tai tekstin kohdalla ei ole taustakuvaa. Tekstin kirjaintyyppit ja -koot ovat helposti luettavia ja tarvittaessa muutettavissa. Sivustolla ei ole käytetty kovin montaa eri kirjaintyyppiä ja kirjainten kokoa. Kirjaintyyppit ovat päätelaitteille sopivia ja käyttäjillä käytössä. Näytöllä päätteetön kirjaintyyppi (esimerkiksi Verdana, Arial) on helpommin luettavissa kuin päätteellinen kirjaintyyppi (esimerkiksi Times New Roman).

Kirjaimen koon tulee olla riittävän suurta, jotta se on luettavaa. Tekstiä voi suurentaa ja pienentää ainakin selaimen asetusten avulla. Suunnittelussa on varauduttu alusta pitäen suurimpaan huomioon otettavaan kirjaimen koon.

Kuvien, grafiikan, äänen, animaation ja videon laatukriteerit liittyvät lähinnä niiden tarkoituksenmukaiseen käyttöön. Kuvilla ja vastaavilla voidaan tietoa havainnollistaa ja painottaa, mutta sivustoa pitää pystyä käyttämään myös ilman kuvaa tai ääntä. Kaikella ei-tekstimuotoisella sisällöllä on oltava tiedon välittävä tekstivastine, kuten esimerkiksi kuvilla alt-teksti. Kuvien ja vastaavien tulisi olla laadukkaita ja nopeasti latautuvia, eikä häiritseviä ilmaisukeinoja, kuten välkkyviä kuvia tulisi käyttää.

8 Uusi sivusto

Sivujen tekninen toteutus uusitaan kokonaan. Taulukkoasettelusta luovutaan ja sivut luodaan XHTML-kielellä ja CSS-tyylitiedostolla. Sivustosta halutaan helppokäyttöinen ja saavutettava. Saavutettavuudella huomioidaan erilaiset käyttäjät.

8.1 Visuaalinen ulkoasu

Sivuston visuaalinen ulkoasu toistuu jokaisella sivulla samanlaisena. Yläkuva voidaan muuttaa tarpeen mukaan. Uuden sivuston ulkoasu näkyy kuvassa 2.



Kuva 2 Uusi etusivu

8.2 Kuvat

WWW-sivustolla olevien kuvien tiedostokoko tulisi pitää mahdollisimman pienenä, koska jokainen sivulla oleva kuva siirretään erikseen palvelimelta. Kuvan pikselikoko ja värisyvyys vaikuttavat kuvan tiedostokokoon. Kuvia voidaan tallentaa eri tiedostomuodoissa. WWW-sivuilla käytetään JPG-, GIF- ja PNG-kuvia. (Keränen ym. 2006: 120, 122)

Kaikki sivuston kuvat tallennetaan esimerkiksi img-kansioon, josta ne linkitetään sivustolle:

```

```

Yläkuva voi olla erilainen eri sivuilla, mutta kuvien suhteen noudatetaan yhdenmukaisuutta, kuten muutenkin sivuston rakenteessa. Jokaisessa yläkuvassa toistuu kesäyliopiston logo. Kuva 3 havainnoi yläkuvan rakennetta.



Kuva 3 Etusivun yläkuva

Alatunniste toistuu jokaisen sivun alareunassa. Alatunnisteessa on kesäyliopiston logon teksti pilkottuna yhdelle riville, sekä puhelinnumero ja sähköpostiosoite (kuva 4). Alatunniste on kokonaisuudessaan kuva. Yhteystiedot löytyvät sivustolta myös tekstimuodossa.

TAMPEREEN ke s ä YLIOPISTO 03-223 8433, kesayliopisto@uta.fi

Kuva 4 Alatunniste

Sivustolle suunniteltiin varjo ala- ja oikeaan reunaan. Tässä mallissa sivun oikean reunan varjo on tehty taustakuvalla. Taustakuva on annettu CSS -tyyleissä, container eli, koko sivun kiertävälle lohkolle. Taustakuva on värittään valkoinen 142 pikseliä korkea ja 740 pikseliä leveä kappale, jolla on oikeassa reunassa varjo. Taustakuvaa toistetaan sivulla.

Navigaation taustakuvana on liukuväripalkki. Värit näkyvät kuvassa 5.



Kuva 5 Navigaatiopalkin taustakuva

Sivuston muita kuvia ovat esimerkiksi Henkilökunta-sivun kasvokuvat.

8.3 Päänavigaatio

Uudelle sivustolle rakennetaan päänavigaatio sivun yläosaan. Päälinkeiksi tulevat *Etusivu*, *Koulutus*, *Ilmoittautuminen*, *Yhteystiedot*, *Palaute*, sekä *In English*. Rakenne esitellään kuvassa 6. Kaikki navigointilinkit kirjoitetaan alkamaan isolla alkukirjaimella. Linkkien väri on tummansininen (#15558E).



Kuva 6 Päänavigaatio

Päänavigaatioon on mahdollista tehdä pudotusvalikot. Pudotusvalikot parantaisivat käytettävyyttä, mutta se saattaisi tuoda mukanaan joitakin saavutettavuusongelmia, esimerkiksi jos toiminto toteutetaan JavaScript-kielillä. Navigaation pitää olla saavutettavissa, vaikka käyttäjä olisikin asettanut tiettyjä ominaisuuksia (kuten selainpohjaiset ohjelmointikieliset tai tyyli) pois päältä.

8.4 Pääsivujen vasemman reunan alanavigointi

Pääsivuille sijoitetaan vasempaan reunaan linkkejä kyseessä olevan otsikon alle kuuluviin asioihin. Lajittelu näkyy kuvassa 7. Pääsivun alisivuilla on yhtenäinen sivunavigointi, esimerkiksi *Koulutus*-sivun alisivuilla on kaikilla sama sivunavigointi

Koulutus	Ilmoittautuminen	Yhteystiedot	Palaute
Avoin yliopisto	Ilmoittautumislomake	Yhteystiedot	Kurssipalaute
Ammatillinen täydennys		Tietoa kesäyliopistosta	
Kielikoulutus		Henkilökunta	
Lukiolaiskurssit		Hallitus	
Finnish For Foreigners		Linkit	
Tentit			
Peruutuskäytännöt			
Ohjeita opiskeluun			
Alueellinen toiminta			
Opiskelijahaastattelu			

Kuva 7 Vasemman reunan navigointi

Etusivulle sijoitetaan oikopolut: *Finnish for foreigners* ja *Tentit. In English* -sivulle käännetään tärkeimmät asiat englanniksi. Tarkempi navigaatiokartta on liitteessä 2.

8.5 Typografia

Tekstin lukeminen näytöltä on hitaampaa kuin paperilta. Internetissä käytettävällä tekstillä on erilaisia vaatimuksia kuin painetulla, esimerkiksi fonttivalinnalla on merkitystä. Tekstin lukemista näytöltä helpottavat muun muassa lyhyet kappaleet sekä selkeät otsikot. Teksti on myös osa sivuston graafista ilmettä. (Korpela ja Linjama, 2004: 184-185)

Sivuston kirjasintyypiksi valittiin Arial. Se on yleinen kirjasinlaji ja pääte-laitteilla helposti luettava. Leipätekstin koko on 12px ja rivinvälitys 120 % (line-height: 120%;).

8.6 Otsikot

Jokaisella sivulla on yksi h1-tason pääotsikko. Otsikko on kyseisen sivun nimi.

```
h1 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 17px;
    color: #BB5E18;
}
```

H3-tason otsikoita käytetään esimerkiksi vasemman reunan otsikoissa. H3-tason otsikot kirjoitetaan ISOILLA kirjaimilla.

```
h3 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #BB5E18;
}
```

8.7 Värit

Väriympyrä voidaan jakaa kylmään ja lämpimään puoleen. Lämpimiä värejä ovat esimerkiksi punainen ja keltainen, kylmiksi väreiksi voidaan sanoa esimerkiksi sinistä ja vihreää. Silmä ärsyyntyy helposti katsottaessa vastakkaisia värejä. Väreillä on vaikutusta ihmisen mieleen ja niiden avulla voidaan herättää tunteita, tunnelmia, makuja ja symboliikkaa. Väriin vaikuttavat myös sen vierellä ja taustalla olevat värit. Kun samanväriset, mutta eri vaaleusasteita edustavat pinnat ovat vierekkäin, ihmisen silmä kokee optisen harhan. Harha ilmenee pintojen rajakohdassa, ja esimerkiksi tumma pinta tummuu entisestään vaalean pinnan vieressä. (Loiri 2006: 111-113)

RGB on kolmivärijärjestelmä, joka koostuu punaisesta, vihreästä ja sinisestä. Musta luodaan muuntotapahtuman avulla. Digitaalikameroiden kuvatal-
lenteet ovat RGB-muotoisia, ja sähköisessä viestinnässä käytetään RGB
-formaattia. RGB-muotoiset kuvat eivät ole painokelpoisia, mutta järjestel-
män vahvuus on värien suuri kirkkaus. Lisäksi kolmivärisyytensä ansiosta
sen tiedostokoko on tietokoneella noin neljännes pienempi kuin nelivärijär-
jestelmien tallenteet. (Loiri 2006: 116)

Uudella sivustolla käytetään kesäyliopiston logon värimaailmaa hieman
mukaillen. Mukaan on haluttu ottaa sekä lämpimiä, että kylmiä värejä. Leipä-
tekstin värinä on niin sanottu tuhkan harmaa (#31302E). Tekstin taustan väri
on valkoinen (#FFFFFF) ja sivuston ulkopuolisen taustan väri on mahdoli-
sesti haalea vaaleanoranssi (#FDF6E7). Kaikki sivuston linkit ovat tumman-
sinisiä (#15558E). Kun hiiri viedään linkin päälle, lukuun ottamatta pääna-
vigaatiota, linkit alleviivautuvat, lisäksi väri muuttuu kaikissa linkeissä vaa-
leanoranssiksi (#F47B2D).

8.8 Hakukone

Sivustolla käytetään Googlen hakukonetta. Koodi on Googlen sivustolta,
mutta se on muutettu validiksi XHTML-koodiksi. Hakukoneella voi etsiä
tietoa kesäyliopiston sivulta sekä Internetistä. Hakutulokset ohjataan avau-
tumaan kesäyliopiston sivulle. Hakukone voidaan sijoittaa jokaisen sivun
vasempaan reunaan tai tehdä erillinen hakusivu.

```
<!-- SiteSearch Google -->
<form method="get" action="http://www.google.com/search">
<p></p>
<p><input type="text" name="q" size="15" maxlength="255"
value="" /><br />

<input type="submit" name="btnG" value="Google Search" /><br
/>

<input type="hidden" name="http://www.uta.fi/kesayliopisto"
/><br />
<input type="radio" name="sitesearch" value="" /> www
<input type="radio" name="sitesearch"
value="http://www.uta.fi/kesayliopisto" checked="checked"
/>kesayliopisto <br /></p>

</form>
<!-- SiteSearch Google -->
```

8.9 Tyylytiedosto

Sivustolle suunniteltu tyylytiedosto on liitteenä 3. Tyylien avulla on rakennettu sivun taitto sekä määritelty kaikki ulkoasumäärytykset, kuten esimerkiksi leipäteksti, otsikot ja navigaatio.

Sivun rakenne tyyleillä rakennettuna näkyy kuvassa 8. Sivun kiertävä container-lohko sisältää content- sekä leftnav-lohkon. Footer-lohkoissa on sivun alatunniste.



Kuva 8 Sivun taitto

9 Sivuston päivitys

9.1 Kurssien päivitys

Nykyisen sivuston yhtenä suurimpana ongelmana on päivittämisen vaikeus. Uutta sivustoa suunniteltaessa huomioidaan myös sivuston ylläpito. Helpoin tapa ylläpitäjille olisi selaimen kautta päivitettävä sivusto. Yliopiston palvelimen takia tietokantoratkaisut ovat kuitenkin poissuljettuja. Sivustolle ei voida rakentaa sisällönhallinta työkalua, eikä valmiita julkaisuohjelmia voida käyttää.

9.1.1 Ammatilliset täydennyskoulutukset - Eduix ja Kuha

Tampereen kesäyliopistolla kurssien hallintaan käytetään kurssinhallintaohjelma Kuhaa. Kuhassa on muun muassa kaikki kesäyliopiston kurssit, ilmoittautumiset, oppilaiden sekä opettajien yhteystiedot ja sen kautta hoidetaan laskutus kursseille. Tampereen kesäyliopiston nykyisellä WWW-sivustolla on käytössä sähköinen ilmoittautumislomake kursseille. Lomakkeen tiedot ladataan suoraan Kuhaan, tiedot käsitellään ja ilmoittautumiset siirretään kursseille. Suurin osa ilmoittautumisista tulee WWW-lomakkeen kautta. Sähköinen ilmoittautumislomake on tilattu tamperelaiselta yritykseltä Eduix Oy.

Eduix Oy on myös kehittänyt Kuhan ja WWW-sivujen välille julkaisutyökälun tai voidaan oikeammin puhua kurssiselaimesta. Kuhan kurssiselaimesta kertoi Eduix Oy:n asiakkuuspäällikkö, Markus Varvio sähköpostivastauksessaan (11.1.2007 haastattelu).

Kuhan kurssitoiminnoissa on web-välilehti. Siihen kirjoitetut tiedot on mahdollista ladata suoraan WWW-sivustolle ja koota Kuhan kursseista Internetissä toimiva kurssiselain. Teksti kirjoitetaan tavallisena tekstinä, eikä siinä tarvitse hallita esimerkiksi HTML-kieltä. Kurssiselaimelle voidaan määritellä, milloin kurssi poistuu WWW-sivulta, esimerkiksi silloin kun se on alkanut tai päättynyt. Ohjelma on tavallaan laajennus jo nyt käytössä olevaan ilmoittautumislomakkeeseen. Taustalla oleva tekniikka on vastaava kuin ilmoittautumislomakkeellakin, ohjelma lukee Kuhasta kurssitiedot ja näyttää niitä verkkosivuilla. Selaus voi olla esimerkiksi opintoaloittain, tyypeittäin tai kurssipaikoittain kesäyliopiston määritelmän mukaisesti.

Ohjelmiston käyttöönotto ei vaadi muutoksia yliopiston palvelimelle, koska ohjelmisto ajetaan Eduix:n palvelimella. Kursseja on mahdollista hakea ja selata eri kriteerein. Selaimen asennuksen yhteydessä määritellään, millä perusteella kurssit näkyvät sivulla - aikarajoitukset voivat olla sidottu esimerkiksi Kuhan ilmoittautumispäivään ja alkupäivämäärään.

Kuhan WWW-sivujen päivitystyökalun käyttöönotolle on monia hyviä perusteita. Kuhan käyttäminen on jo ennestään tuttua kaikille Tampereen kesäyliopistolla työskenteleville, joten kurssien päivittäminen WWW-sivustolle ei vaadi minkään uuden ohjelman opettelemista. Tässä tapauksessa ei tarvitsisi myöskään hallita ohjelmointikieltä. Jokainen työntekijä pystyisi päivittämään omia kurssejaan koskevia tietoja helposti ja vaivattomasti WWW-sivustolle.

9.1.2 Avoin yliopisto-opetus - Meteori-tietokanta ja täsmähaku

Tampereen kesäyliopisto tarjoaa myös avointa yliopisto-opetusta. Nykyisellä WWW-sivustolla avoin yliopisto-opetus on linkitetty avoimen yliopiston Meteoritietokantaan, mutta erillisen linkityssivun kautta. Kesäyliopiston sivuston *Avoin yliopisto* -sivulle kirjoitetaan siis kurssin nimi ja se linkitetään siitä kurssin Suvi-dokumenttiin, joka sijaitsee meteoritietokannassa. Suvi-dokumentti on dokumentti, jossa on tietoa kurssista, esimerkiksi kurssin tavoitteet, esitietovaatimukset, kouluttaja, aika, paikka ja sisältö. Suvi-dokumentteja päivitetään suoraan selaimen kautta. Linkitysten tekeminen kesäyliopiston omille sivuille on suhteellisen työlästä, koska kursseja on varsinkin kesäkaudella paljon. Avoimen yliopiston tarkkoja kurssitietoja ei ole järkevää siirtää Kuhaan ja päivittää niitä esimerkiksi Kuhan WWW-sivujen päivitystyökalulla, koska tiedot on joka tapauksessa kirjattava Suvi-dokumentteihin.

Päivittämisen kannalta paras ratkaisu olisi täsmähaun tekeminen ja sen avautuminen suoraan kesäyliopiston *Avoin yliopisto* -sivulle. Avoimen Meteorio-petustietokannasta on mahdollista poimia täsmähaku (kuva 9). Tämä tarkoittaa sitä, että määritellään hakuehdot, joiden mukaan avautuu sivu, jossa on listattuna esimerkiksi kaikki Tampereen kesäyliopiston järjestämä avoin yliopisto-opetus. Tällä tavoin ei tarvitsisi enää päivittää avoimen yliopisto-opetuksen kursseja tai poistaa jo päättyneitä kursseja omilta sivuilta. Täsmähaku hakee opetustiedot reaaliajassa eli päivitykset näkyvät heti myös haun tuloksissa.

Opetustiedot

Opetusohjelmat on listattu aakkosjärjestyksessä oppiaineen mukaan järjestettyinä.

	YO:	Paikka:	Toteuttaja:	Alkaa:
BIOLOGIA				
■ Biologian perusopinnot, monimuoto-opinnot (25 op)	TäY>	Tampere	Tampereen kesäyliopisto	Syky 2006
ENGLANTI				
■ Englannin kielen kommunikatiivinen (suullinen) taito, ryhmä 3 (3 op, 2 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
■ Englannin kielen kommunikatiivinen (suullinen) taito, ryhmä 4 (3 op, 2 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
■ Englannin kielen valmennuskurssi (teksti) (2 op, 1 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
ERITYISPEDAGOGIIKKA				
■ Erityispedagogiikan perusopinnot (25 op)	HY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
ESPANJA				
■ Espanjan kielen jatkokurssi (6 op, 4 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
■ Espanjan kielen kirjallinen täydennyskurssi (2 op, 1 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
FILOSOFIA				
■ Filosofian perusopinnot (25 op, 15 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
INFORMAATIOTUTKIMUS				
■ Tiedonhauksen perusteet (informaatiotutkimuksen erillinen perusopintojakso) (4 op, 3 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
ITALIA				
■ Italian kielen jatkokurssi (6 op, 4 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
KANSANTALOUSTIEDE				
■ Kansantaloustieteen opinnot (21 op)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006
■ Johdatus kansantaloustieteeseen (6 op, 4 ov)	TäY>	Tampere	Tampereen kesäyliopisto	Kesä 2006

Kuva 9 Täsmähaun tuloksia, kun toteuttajana on Tampereen Kesäyliopisto ja kurssien alkamisajankohta on 2006.

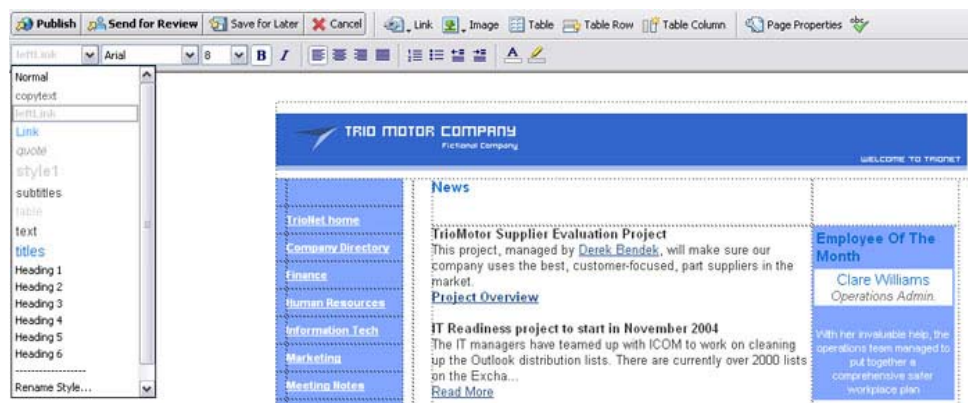
Tampereen kesäyliopiston työntekijöillä on tunnussanat ja käyttöoikeus Me-teoritietokantaan. Suvi-dokumenttia on helppo muokata ja se on tuttua jo ennestään avoimesta yliopisto-opetuksesta vastaaville kesäyliopiston työntekijöille. Täsmähaun käyttäminen olisi paras tarjolla olevista vaihtoehdoista, koska Suvi-dokumentteja päivitetään joka tapauksessa.

9.2 Sivuston muu päivitys

Sivuston lähes päivittäiset ja suurimmat päivitykset liittyvät kursseihin. Kurssien päivittämisen helpottaminen auttaa paljon. Sivuston muita asioita, kuten esimerkiksi tenttejä, etusivun ajankohtaisia kursseja tai henkilökunnan muuttuvia yhteystietoja, päivitetään satunnaisesti ja tarvittaessa. Näiden päivitysten osuus koko sivuston ylläpidosta on kuitenkin suhteellisen vähäistä.

9.2.1 Adobe Contribute

Adobella on työkalu WWW-sivujen päivittämiseen, nimeltään Adobe Contribute. Contribute:n avulla WWW-sivuja voidaan päivittää helposti ja vaivattomasti. Päivittäessä ruutuun kirjoitetaan tekstiä, kuten mihin tahansa tekstieditoriin. Contribute:n työpöytä näkyy kuvassa 10.



Kuva 10 Adoben Contribute 4:sen työpöytä

Keskellä näkyy muokattava sivu. Sivulle voidaan lisätä esimerkiksi tekstiä, kuvia, linkkejä ja taulukoita. Kuvan yläpaneelissa näkyvät muun muassa työkalut *Link*, *Image* ja *Table*. Näiden avulla muotoileminen on helppoa. Tekstin tai kuvien poistaminen tehdään esimerkiksi näppäimistön *Delete*-painikkeella. Kuvan vasemmassa reunassa oleva avonainen valikko on lista tyyleistä, joita sivuston tyyliin on luotuna. Listasta voidaan valita tyyli viemällä hiiren kursori oikeaan kohtaan ja kirjoittaa halutulla tyyllillä. Contribute:n kautta kirjoitettu teksti näyttää koodissa esimerkiksi tältä:

```
<p style="margin-bottom: 0;">testi-teksti1</p>
<h2 style="margin-top: 0;">testi-teksti2</h2>
<h3>testi-teksti3</h3>
<p><span class="style1">testi-teksti4</span></p>
<p style="margin-top: 0; margin-bottom: 0;"></p>
```

Contribute ei pyydä automaattisesti kuviin alt-tekstiä, mikä on XHTML-kielessä vaadittava ominaisuus. Contribute:n tekemä koodi ei muutoinkaan ole kovin hyvää XHTML-kieltä, koska esimerkiksi kappale-elementissä on erikseen määriteltynä tyyli.

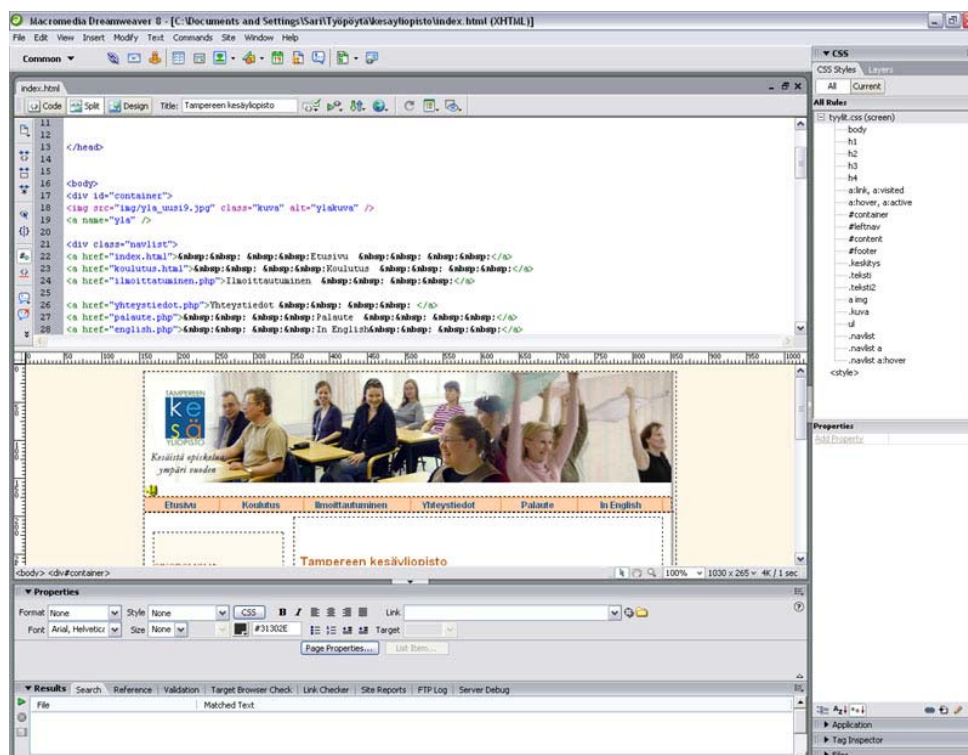
Sivulle tehdyt muutokset voidaan julkaista Internetissä painamalla *Publish*-painiketta (näkyvät kuvan 9 vasemmassa yläreunassa) ja muutokset näkyvät saman tien. Tämä edellyttää sitä, että sivustolla on luotu yhteys palvelimelle, *Create Connection* ja palvelimen käyttäjätunnus ja salasana on tallennettuna. Yhteys voi olla esimerkiksi ftp-yhteys tai suojattu sftp-yhteys.

9.2.2 Macromedia Dreamweaver

Macromedian Dreamweaver on suosittu WWW-tuottamisohjelma. Dreamweaverissä voidaan valita kolme erilaista työtilaa: Code, Split ja Design. Code-työtilassa nähdään pelkkä koodi, Design näyttää luodun sivun ulkoasuun ja Split jakaa työtilan kahteen osaan (kuva 11), missä ylempänä näkyy koodi ja alempana ulkoasu. Sivua voidaan muokata Design- tai Split-työtilassa näkyvään ulkoasuun, ja täten sivun muokkaaminen muistuttaa Contribute:n kautta tehtävää päivitystä. Samalla voidaan kontrolloida, miten

muutokset näkyvät koodissa. Sivua voidaan myös kirjoittaa pelkästään koodina Code-työtilassa. (Lyytikäinen 2003: 18-23)

Dreamweaverissä esimerkiksi kuvien, taulukoiden, listojen ja linkkien lisääminen on yhtä helppoa kuin Contribute:ssa. Dreamweaverissä on valmiita toimintoja näitä varten, eikä esimerkiksi kuvan lisäämiseksi tarvitse osata koodia. Dreamweaverin 8-versio kysyy alt-tekstiä kuvaa lisättäessä. Kaikkia näytöllä näkyviä elementtejä voidaan muokata suoraan Properties-ikkunasta.



Kuva 11 Dreamweaver 8:n Split-työtila

9.2.3 Macromedia Dreamweaver vastaan Adobe Contribute

Contribute on yhteensopiva Macromedia Dreamweaverin kanssa. Molempien ohjelmien työkalu-pikakuvakkeet ovat samantapaisia.

Contributen hyviä puolia on muun muassa se, että sitä käyttäessä ei tarvitse osata HTML-kieltä tai edes FTP-yhteyden käyttöä. Sivua avataan, tehdään muutokset ja julkaistaan. Päivitysten tekeminen on nopeaa ja helppoa. Huonoiksi puoliksi voidaan laskea se, että Contribute:n kautta ei pysty näkemään koodia. Mikäli koodia halutaan muokata, sivu täytyy avata Dreamweaveriin. Contribute on tehty hieman yksinkertaisemmaksi kuin Dreamweaver, mutta senkin käytön aloittaminen vaatii opettelua.

Myös Dreamweaverillä voidaan Design-ikkunassa tehdä pieniä päivityksiä ilman koodiin kirjoittamista. Lisäksi moniin perustoimintoihin on valmiita

pikakuvakkeita. Dreamweaverissä on enemmän toimintoja kuin Contribute:ssa ja siksi se saattaa aluksi tuntua vaikealta ja monimutkaiselta. Kesäyliopistolla on tällä hetkellä käytössä Dreamweaver 2004MX.

9.3 Toistuvat osiot – PHP

Tampereen yliopiston palvelin tukee PHP:tä, ja tämänhetkinen (11.9.2006) PHP:n versio on 4.3.10. Ennen mahdollisten PHP-funktioiden tekemistä, kannattaa aina tarkastaa palvelimen PHP:n versio ja selvittää, tukeeko se käytettäviä funktioita.

PHP-dokumentteihin voidaan liittää tiedostoja esimerkiksi `include_once ()` -tai `require_once ()` -functiolla. Esimerkiksi sivuston yläosa, navigointi ja alatunniste ovat kaikille sivuston sivuille yhteisiä. Ei ole mielekästä tai edes järkevää liittää näiden osioiden koodia jokaiselle sivulle. Lisäksi, jos esimerkiksi alatunnisteeseen halutaan tehdä muutoksia, muutoksen tekeminen jokaiselle sivulle olisi todella työlästä. Jos liitämme alatunnisteen yhdestä tiedostosta jokaiselle sivulle, voimme tehdä muutoksen yhteen paikkaan, ja se vaikuttaa samalla kertaa kaikkialle. Tällä tavoin ei myöskään ole vaaraa siitä, että vahingossa poistaisi jotain, esimerkiksi navigoinnista, jos päivityksiä tekee suoraan koodiin.

Periaatteessa tiedostojen yhdistäminen voitaisiin tehdä myös SSI-tekniikalla. Yleisimmin käytetään PHP:tä esimerkiksi sen monikäyttöisyyden takia.

Index.php sivun koodi voisi näyttää esimerkiksi tältä:
(tekstiosuoksia on lyhennetty)

```
<?php include_once("yla.php"); ?>

<h1>Tampereen kesäyliopisto</h1>
<p>Tampereen kesäyliopisto järjestää avointa yliopisto-
opetusta, ammatillista täydennyskoulutusta sekä kieli- ja
abikursseja ympäri vuoden.</p>

<p><strong>Kesäyliopisto on avoinna:</strong><br />
ma-pe klo 8.00-16.00<br />
Yliopistonkatu 60 A, puh 03-2230 8433, kesayliopis-
to(at)uta.fi</p>

<h3>AJANKOHTAISTA</h3>
<ul><li><a href="linkki.php">Venäjän kieli 15.11.2006-
4.4.2007</a></li>
<li><a href="linkki.php">Kehityskeskustelut johtamisen tukena
23.11. </a></li>

<?php include_once("ala.php"); ?>
```

```
<?php include_once("yla.php"); ?>
```

Tämä rivi tuo sivulle esimerkiksi yläosion ja navigoinnin. Yläosan ja navigoinnin koodi on kirjoitettu normaalisti XHTML-kielellä tiedostoon yla.php.

```
<?php include_once("ala.php"); ?>
```

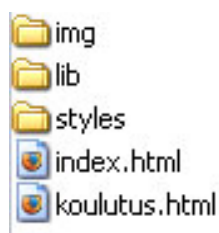
Tämä rivi tuo sivulle esimerkiksi alatunnisteen. Alatunnisteen koodi on kirjoitettu normaalisti XHTML-kielellä tiedostoon ala.php.

Sivuston päänavigointi on jokaisella sivulla sama, mutta päänavigoinnin alasuilla on myös oma navigointinsa sivuston vasemmassa reunassa. Alasuujen navigointi voitaisiin myös tehdä omiksi tiedostoiksi, jotka tuodaan PHP:llä tarvittaviin paikkoihin.

10 Sivuston toteutukseen liittyviä asioita

10.1 Tiedostojen järjestäminen ja käsittely

Koska sivuston tiedostoja joudutaan käsittelemään myös käsin ja niitä siirretään WinSCP-SFTP-ohjelmalla yliopiston palvelimelle, tiedostojen hyvä ja looginen järjestys on tärkeää. Esimerkiksi kaikki kuvat, toistuvat osiot (yläosiot, navigoinnit yms.) ja tyylitiedostot tulisi laittaa omiin kansioihin, josta ne linkitetään sivulle. Järjestämisen voisi tehdä esimerkiksi kuten kuvassa 12.



Kuva 12 tiedostojen järjestäminen

Img-kansio sisältää sivuston kuvat. Ne linkitetään sivustolle:

```

```

Lib-kansio sisältää sivuston toistuvat osiota, esimerkiksi yläosion ja navigoinnit. Ne linkitetään sivustolle:

```
<?php include_once("lib/yla.php"); ?>
```

Styles-kansiossa on sivuston tyylitiedostot. Ne linkitetään sivustolle:

```
<link href="styles/tyylit.css" rel="stylesheet" type="text/css" media="all" />
```

Palvelimella ei myöskään kannata säilyttää tiedostoja, joita ei käytetä sivustolla. Lisäksi kaikki sivuston tiedostot pitäisi olla tallessa tietokoneen muistissa.

10.2 PDF-tiedostot

PDF-tiedosto on Aboden kehittämä sähköinen julkaisukeino. Ensimmäinen versio ohjelmasta, jolla PDF-tiedostoja voitiin lukea ja muokata oli vuonna 1993 julkaistu Adobe Acrobat. PDF-dokumentti voi sisältää tekstiä, valokuvia ja grafiikkaa ja niihin voidaan sisällyttää myös paljon ominaisuuksia, jotka tekevät siitä interaktiivisen. PDF-dokumentteja voidaan käyttää esimerkiksi julkaisujen jakamisen WWW-sivuilla, painettavien julkaisujen siirtämiseen tai sähköisiin kirjoihin. (Keränen ym. 2006: 216-218)

PDF-tiedostoja tulisi käyttää harkitusti WWW-sivustolla. PDF:ät vaativat erillisen lukijan, joten ei voida olettaa, että PDF:ssä oleva tieto saavuttaa kaikki sivuston käyttäjät. PDF-tiedostot ovat myös usein suuria kooltaan ja niiden avautuminen kestää huomattavasti kauemmin kuin tavallisen HTML/XHTML-sivun. Jos PDF-tiedostoja halutaan jakaa WWW-sivustolla, niiden tiedostokokoo tulisi muokata sopivaksi. (Keränen ym. 2006: 223-225)

10.3 Testaaminen ja validointi

Kun sivusto on kokonaisuudessaan valmis, sitä tulisi testata käytettävyyden sekä toteutuksen suhteen. Käytettävyyden testaamiseksi voidaan esimerkiksi valita erillinen käyttäjäryhmä, joka testaa sivustoa ja antaa arvion. Testaamiseen voidaan käyttää esimerkiksi soveltuvin osin Laatu verkkoon sivuston valmista arviointityökalua. Arviointilomake löytyy osoitteesta:

http://www.suomi.fi/suomifi/laatuverkkoon/laatukriteerit/arvioi_verkkopalvelu/index.html

Sivusto olisi hyvä testata myös eri näyttöresoluutioilla ja, jos mahdollista eri värisyvyyksillä. Sivujen latautumista voidaan kokeilla eri yhteysnopeuksilla. Selaimet ovat erilaisia ja siksi sivustoa tulisi käyttää erilaisilla selaimilla ja selainversioilla. Huomioon otettavia selaimia ovat ainakin: Internet Explorer, Mozilla, Firefox, Opera ja mahdollisesti myös Safari ja Netscape. (Keränen, Lamberg ja Penttinen. 2003: 19-20)

Eryteisesti XHTML-dokumentin validointi on tärkeää. XHTML suhtautuu tiukasti virheisiin ja periaatteessa virhe voi lopettaa dokumentin käsittelyn. Käytännössä useimmat selaimet kuitenkin käsittelevät XHTML-dokumentin kuten HTML:n, ja sivu näkyy kokonaisuudessaan virheistä huolimatta. Validointi tarkoittaa dokumentin kielen tarkastamista dokumenttityypin (DTD) ja yleisten sääntöjen mukaan. (Korpela ja Linjama, 2004: 109-111)

XHTML- ja CSS-dokumentit voidaan tarkastaa W3C:n sivuilta löytyvillä validaattoreilla:

XHTML <http://validator.w3.org/> ja CSS <http://jigsaw.w3.org/css-validator/>.

Saavutettavuutta voidaan testata esimerkiksi alla olevan validaattorin avulla: <http://webxact.watchfire.com/>.

Saavutettava.fi on WWW-sivujen saavutettavuuden tietopankki. Sieltä löytyvällä WWW-standardien tarkistuslistalla voidaan tarkastaa oman sivuston ominaisuuksia:

<http://saavutettava.fi/artikkelit/web-standardien-tarkistuslista/>

Värisokeuden osalta sivuston voi testata, esimerkiksi Colorblind Web Page Filter-ohjelmalla osoitteessa: <http://colorfilter.wickline.org/>.

10.4 Toteutussuunnitelma

Sivuston toteuttaa kesäyliopistolle hypermedian harjoittelija. Toteuttajalta vaaditaan seuraavaa osaamista:

1. Hyvät tiedot XHTML+CSS.
2. PHP-kielen perustaidot.
3. Kuvankäsittelyn perusteet (käytössä PhotoShop).
4. Tarvittaessa osaa toteuttaa navigaatioon avautuvan listan.

10.4.1 Budjetti ja Aikataulu

Sivuston kustannukset tämän suunnitelman pohjalta tehtynä ovat Eduix Oy:n kehittämän kurssiselaimen asennus ja käyttöönotto. Eduix Oy:n kurssiselain maksaa 500 euroa (kertamaksu) + asennus. Eduix Oy:n tuntitaksa on 65 euroa ja asennuksessa menee noin 2-4 tuntia (hintatiedot syksy 2006). Muita kustannuksia ovat mahdolliset sivuston toteuttajan palkkiot.

Mahdollisuuksien mukaan voidaan myös ostaa Adoben Contribute tai päivittää Studio-paketti uuteen versioon, jolloin käytössä olisi uudempi versio Dreamweaveristä.

Aikataulullisesti uuden sivuston käyttöönoton pitäisi tapahtua vuoden 2007 aikana.

10.4.2 Materiaali

Kesäyliopistolle tuleva materiaali: opinnäytetyö, suunnittelua varten tehty tyylitiedosto sekä kuvat (jpg- ja psd-formaattina) toimitetaan Tampereen Kesäyliopistolle. Tampereen Kesäyliopisto saa oikeuden käyttää materiaalia joko kokonaan tai soveltuvien osien. Materiaalia saa muokata tarvittaessa.

11 Tulosten tarkastelu ja arviointi

Sain aiheen työlleni elokuun lopussa 2006. Suoritin silloin tutkintooni kuuluvaa harjoittelua Tampereen kesäyliopistolla. Käytin päivittäin kesäyliopiston nykyistä WWW-sivustoa sekä myös päivitin sivuja. Sain kattavan kuvan siitä, mitä asioita kannattaisi muuttaa tai parantaa sivustoon liittyen.

Sivuston suunnitteluvaiheessa kesäyliopiston yhteyshenkilöni Sanna Vaurulan kanssa on ollut helppo työskennellä. Olen yrittänyt ottaa huomioon kesäyliopiston puolelta tulleet mielipiteet ja toiveet. Mielestäni saimme kasaan hyvän ja toivottavasti käytännössäkin toimivan paketin. Yliopiston palvelimen rajoitusten takia päivittämiseen ei löytynyt yhtä ratkaisua, eikä koodin kautta tapahtuvasta päivittämisestä päästy kokonaan eroon. Löysimme kuitenkin keinoja, joilla päivittämistä voidaan helpottaa ja niin sanotun turhan työn tekemisestä vähentää.

Sivuston rakennetta ja navigointia selkeytettiin paljon ja tärkeät asiat ovat uudella sivustolla paremmin esillä kuin nykyisellä. Näinkin laajalla sivustolla yhdenmukaisuuden ja loogisuuden merkitys on tärkeää. Kun asiaa on paljon, käyttäjän pitäisi pystyä päättämään nopeasti, mistä hän löytää etsimänsä tiedon. Kaikkien linkkien tulisi olla samannäköisiä keskenään ja erottua muusta tekstistä. Visuaalinen ulkoasu on myös tärkeä, mutta se ei saa olla sivuston suunnittelun pääasia. Tärkeintä on, että väreillä tai kuvilla ei ärsytetä tai loukata ketään ja, että sivusto olisi mahdollisimman monen käyttäjän saatavilla. Ulkoasuun liittyvillä asioilla ei saa estää sivun toimintoja.

Toimeksiannon tekemisen suurimpia ongelmia oli päivittämisen ratkaiseminen ja sivuston suunnitteleminen sellaiseksi, ettei taas vuoden päästä kesäyliopiston sivuja pidetä vaikeakäyttöisinä ja sekavina. Yliopiston palvelin saneli suurimmat ehdot sivuston tekniselle toteutukselle. Toisaalta Kuhan kautta päivittäminen voi olla parempi vaihtoehto kuin täysin selainpohjainen päivitystyökalu. Kuha on kesäyliopiston jokapäiväinen ja tärkein työkalu, ja tiedot kirjataan sinne joka tapauksessa. Kesäyliopistolla on uuden sivuston suhteen vielä suuri haaste edessä. Uuden käyttöönottoaminen ja siihen totuttaminen vie aikansa. Toivon kuitenkin, että tämä työ jaksetaan tehdä ja asiat viedään loppuun saakka.

Omalta osaltani onnistuin työn tekemisessä. Olin ehtinyt syksyn aikana pohdita aiheitani ja etsiä avoimena oleviin kysymyksiin ratkaisuja. Kun ryhdyin kirjoittamaan asioita tähän muotoon, tekstiä tuli helposti. Työssä summautuvat koulussa oppimiani asioita, erityisesti suuntautumisvaihtohehtoni osalta. Suurin osa asioista oli tuttuja, mutta työtä tehdessäni opiskelin monia asioita uudestaan ja osaltaan syvensin tietojani.

Mielestäni myös työlle tavoitteeksi asetetut tavoitteet täyttyivät. Lopputuloksena on kattava paketti kokonaan uuden sivuston suunnittelusta. Opin näytetyä esittelee erilaisia vaihtoehtoja sivuston toteuttamiseen sekä valmiin sivuston ylläpitämiseen.

Lähteet

Painetut lähteet:

Korpela, Jukka K & Linjama, Tero 2004. XHTML-käsikirja. Jyväskylä: Docendo.

Keränen, Vesa, Lamberg, Niko & Penttinen, Jukka 2003. Verkkojulkaisun hallinta. Jyväskylä: Docendo.

Keränen, Vesa, Lamberg, Niko & Penttinen, Jukka 2006. Web-ulkaiseminen & Multimedia. Jyväskylä: Docendo.

Linjama, Tero 2001. XHTML. Jyväskylä: Docendo.

Loiri, Pekka 2006. HUOM! Visuaalisen viestinnän käsikirja. Jyväskylä: Inforviestintä.

Lyytikäinen Miikka 2004. Dreamweaver MX 2004. Jyväskylä: Docendo.

Zandstra, Matt 2001. PHP (Trainer Kit). Helsinki: Edita, IT-Press.

Zeldman, Jeffrey 2003. Designing with Web standards. Indianapolis: New Riders.

Verkkolähteet:

2kmediat.com - PHP.

[online] [viitattu 7.1.2007].

<http://www.2kmediat.com/php/johdanto.asp>

2kmediat.com - SSI.

[online] [viitattu 7.1.2007].

<http://www.2kmediat.com/dhtml/ssi.asp>

MofiWiki - kotimaisen Mozilla-yhteisön wikipedia [online] [viitattu 6.1.2007].

http://www.mozilla.fi/wiki/XHTML:CSS:n_perusteet

Ohjelmointiputka - PHP [online] [viitattu 11.12.2006].

<http://www.ohjelmointiputka.net/opas.php?tunnus=phpj8>

Ohjelmointiputka - SSI [online] [viitattu 11.12.2006].

<http://www.ohjelmointiputka.net/opas.php?tunnus=ssi>

Suomi.fi – Laatu verkkoon [online] [viitattu 7.12.2006].

<http://www.suomi.fi/suomifi/laatuverkkoon/>

Tampereen kesäyliopisto [online] [viitattu 9.12.2006].

<http://www.uta.fi/kesayliopisto>

Tampereen yliopisto – WWW-politiikka [online] [viitattu 3.11.2006].
<http://www.uta.fi/palvelut/webmaster/WWWpolitiikka.html>

Why tables for layout is stupid 2003 [online] [viitattu 11.12.2006].
<http://WWW.hotdesign.com/seibold/>

Sähköpostihaastattelut:

Suominen Markku: Joomla!-portaali, 18.8.2006.

Varvio Marku: Eduix Oy, 11.1.2007.

Vaurula Sanna: Tampereen kesäyliopisto, 19.10.2006.

Aineistona käytettyjä lähteitä:

Abode – Contribute [online].
<http://www.adobe.com/fi/products/contribute/>

Avoin yliopisto – Täsmähauan ohjeet [online].
<http://www.avoinyliopisto.fi/fi-FI/tasmahakulomake/>

Joomla! portaali – ilmainen julkaisujärjestelmä [online].
<http://www.joomlportal.fi/content/view/93/39/>

W3C - World Wide Web Consortium [online].
<http://www.w3.org/>

Saavutettava.fi [online].
<http://saavutettava.fi/>

Google - Lisää/Päivitä URL [online].
<http://www.google.fi/intl/fi/searchcode.html>

Webmaster: Tampereen yliopisto, 23.8.2006 ja 11.9.2006.

Liitteet

Liite 1 – Kesäyliopistojen sähköpostihaastattelu

Kysymykset kesäyliopistolle:

1. Milloin käytössänne olevat www-sivut on tehty?
2. Monennetko nykyiset sivunne ovat?
3. Miten sivut on tehty? (joku työntekijä tehnyt, tilattu jostain - mistä?, jokin julkaisuohjelma - mikä?)
4. Onko sivujen tekemiseen laitettu paljon rahaa, voit antaa noin arvion tai jonkin hintahaitarin?
5. Millä palvelimella sivut fyysisesti sijaitsevat?
6. Mitä ohjelmointikieliä sivujen tekemiseen on käytetty?
7. Miten sivuja päivitetään ja miten usein? (siis miten päivitys tapahtuu käytännössä - viedäänkö esim. kurssit nettiin kirjoittamalla suoraan koodiin, jonkin ohjelman avulla, miten?)
8. Kuka sivuja päivittää? (joku tietty työntekijä, jokainen työntekijä, joku muu?)
9. Ovatko sivut helppokäyttöiset ja helppo päivittää?
10. Jotain mitä haluat mainita sivustostanne (onko jotakin erikoista, jotain hyväksi havaittua, jotain turhaa, mitä suosittelet meidän uusille sivuillemme)?
11. Oletteko tyytyväisiä omiin sivuihinne (vai oletteko uusimassa niitä lähiaikoina - miksi?)

Vastaus 1:

1. nykyiset sivut on tehty noin kaksi vuotta sitten
2. sivut ovat kolmannet
3. ulkopuolinen tekijä, kesäyliopistossa atk-kursseja pitänyt henkilö (FrontPage, ohjelman valinta on määräytynyt sen mukaan, että ohjelma on käytössä ympäristössä, jossa olemme mukana (so. yliopisto)
4. varmaa tietoa ei ole
5. xxx yliopiston täydennyskoulutuskeskuksen palvelimella
6. käsittääkseni shtml (koskapa dokuissa on tuo päätte)
7. tarvittaessa, päivitys käytännössä jatkuvaa (kolme henkilöä, kukin oman toimialueensa osalta)
8. -
9. ei ihan helpommasta päästä, mutta ei kovin vaikeatkaan; tyylikoodiston voi helposti hävittää ja saada sivut sekaisin, varminta onkin koodiin päivittäminen
10. Omamme eivät ole käyttäjäystävällisimmät sivut em. syistä
11. Emme ole lähiaikoina uusimassa, mennään nyt näin eteenpäin

Vastaus 2:

1. 2002
2. toiset
3. Firma nimeltä Datafisher
4. n. 5000 e
5. Puhelinyhdistyksen palvelimella
6. ?
7. Päivitämme itse muuttuvat tekstit (esim. koulutustarjonta). Staattisten osien päivitys ostetaan Datafisheriltä

8. Kuka tahansa voi päivittää. Ei vaadi erityisiä taitoja.
9. Ovat
10. Mahdollisimman pitkälle itse päivitettävät sivut ovat parhaat
11. Pieniä muutoksia on tehty jatkuvasti, mutta periaatteessa ollaan tyytyväisiä

Vastaus 3:

1. Sivut on otettu alun perin käyttöön joskus -90 -luvun puolivälissä. Sivut on kokonaan uusittu viimeksi syksyllä 2004.
2. Ehkä neljännet tai viidennet
3. Viimeksi oli tilattu Image Wordistä, mutta nyt itse tehty. FrontPage -ohjelmalla.
4. Vain työpanos.
5. xxx yliopiston palvelimella.
6. html
7. Avoin yliopisto-opetus napataan Suvi-sivustosta ja muut kurssit päivittyvät automaattisesti Kuhasta, muuten päivitetään pääsivua ja ajankohtaista sivua viikoittain.
8. Useat työntekijät päivittävät.
9. Kyllä.
10. Suosittelen avoimen yliopisto-opetuksen nappaamista suoraan Suvista, jolloin ei tarvitse päivittää kahta sivustoa. Myös kurssien nappaaminen Kuhasta helpottaa päivittämistä huomattavasti.
11. Olemme pääpiirteissään tyytyväisiä eikä suuria uudistuksia ole tulossa.

Vastaus 4:

1. v. 2003
2. toiset
3. tilattu xxx yrityksestä nimeltään Oy Creamarketing Ab
4. Valmistus 1300-1400 eur, vuosimaksu 200-250 euroa. Uusien juttujen tekeminen maksaa erikseen (ei mitään suuria summia).
5. Creamarketingin palvelimmella.
6. PHP4
7. Aina tarvittaessa, sitä mukaa kun saamme uutta tietoa kurseistamme, kiireisimpänä aikana useampi kerta viikossa.
8. sihteerit päivittävät
9. on helppokäyttöiset
10. Sivumme ovat vaatimattomat mutta helpot käyttää. Liian sekavia sivuja kannattaa varmaan välttää. Ainakin sähköinen ilmoittautuminen on hyvä. Huomasin talvella että teillä oli sähköpostin kautta, kun poikani ilmoittautui teille abikurssille. Se ainakin kannattaa uudistaa. Ja varmaan tämä e-lomakekin on hyvä saada käyttöön (ilmoittautumiset suoraan Kuhaan). Itse ainakin aiomme ottaa sen, kun uudistamme sivuja.
11. Olemme uusimassa... Kaikki hyvät vinkit ovat meillekin tarpeen, voisit pistää kaikille vastaajille jakeluun, jos saat hyviä vinkkejä.

Vastaus 5:

1. 01/2006
2. 1998- (9)
3. -
4. n 1900 €
5. Sydweb Ab
6. ?
7. Päivitän itse, tarpeen vaatiessa. Se on vaivatonta.

8. Päivitin itse.
9. Kyllä.
10. Koko ohjelma näkyy ruudussa heti. Ei mitään turhia tasoja.
11. Olen tyytyväinen.

Vastaus 6:

1. 2004 päivitetty
2. kolmannet
3. Delta suunnittelu oy, tilaustyö
4. 10 000-15000mk
5. -
6. html
7. Front Page, vaivatonta ja helppoa
8. -
9. Kyllä.
10. -
11. Olen tyytyväinen.

Vastaus 7:

1. Vuonna 2004.
2. Toiset.
3. Pohja, eli ulkoasu on tilattu (Eduix-nimiseltä yritykseltä), sisältö on tuotettu itse.
4. Hintatiedot eivät ikävä kyllä ole julkisia.
5. Eduixin palvelimella.
6. Pohjan tekniikasta ei ole tarkkaa tietoa. Sivujen tiedot päivitetään tekstitiedostoista, tarvittaessa käytetään html-kieltä (esim. lihavoointeihin). Kurssitiedot ohjelma hakee suoraan KuHa-kurssinhallintaohjelmasta, muut tekstit (ilmoittautumisohjeet jne.) vietään Eduixin tekemällä Sivallus-ohjelmalla.
7. Sivuja päivitetään ympäri vuoden tarpeen mukaan, esim. uusi kurssitarjonta, muuta ajankohtaista, kurssien täyttymiset ja peruuntumiset jne.
8. Pääasiassa suunnittelijat ja koulutussihteerit.
9. Sikäli helpot, että tekniikka on helppo oppia ja on siinä mielessä käyttäjäystävällinen, mutta päivittäminen on varsin työlästä, kun kaikki tehdään ns. käsityönä.
10. -
11. Olemme muuten tyytyväisiä sivuihin, mutta e-lomakkeiden käyttö ja nettikauppa (joita silmällä pitäen sivut aikoinaan uusittiin) on vielä aivan kesken. Sitä puolta on tarkoitus kehittää lähitulevaisuudessa.

Vastaus 8:

Nyt en ole ihan varma milloin nettisivumme tulivat, mutta siinä 1900-2000 -luvun vaihteessa. Ne ovat meidän ensimmäiset sivut ja vielä käytössä ainoastaan ulkonäköä (etusivun kuvat ja tekstin väri) muutamme vuosittain tai puolivuositain syksyn ja kevään esitteen mukaan. Perusrunkoa on kerran muutettu, otin yläpalkin pois. Sivut on tehty Front-Page ohjelmalla, tunnin pikakoulutuksen jälkeen. Siis minä olen tehnyt sivut ja päivitin myös niitä tarvittaessa. Päivitys tapahtuu FrontPagella ja siirto ftp-ohjelmalla Soneran palvelimelle. Sivujen päivitys ei ole vaikeaa. FrontPagella tehdessä ei tarvitse kirjoittaa html koodia. Mielestäni sivut on helppokäyttöiset. Sivuja olemme uusimassa syksyn aikana.

Liite 2 - Navigaatiokartta

<i>Etusivu -></i>	(Oikopolut: Finnish for foreigners, Tentit)
<i>Koulutus -></i>	Ammatillinen täydennyskoulutus Avoin yliopisto-opetus Finnish for foreigners Lukiolaiskurssit, ->lukio-opiskelijille Kielikoulutus Peruutuskäytännöt Ohjeita opiskeluun kesäyliopistossa Tentit Alueellinen toiminta ja hankkeet -> aluekehittäjän ko. -> pro pirkanmaa korkeakoulujen palvelupisteet -> tutorpankki Haastattelu yhdestä opiskelijasta
<i>Ilmoittautuminen -></i>	Ilmoittautumislomake (selkeästi maininta peruutuskäytännöistä)
<i>Yhteystiedot -></i>	Yhteystiedot Henkilökunta Hallitus -> pöytäkirjat -> esityslistat Tietoa kesäyliopistosta Linkit
<i>Palaute-></i>	Kurssipalautelomake (toiveita/ehdotuksia/risuja/ruusuja kursseista tms.)
<i>In English-></i>	Tärkeimmät asiat + finnish for foreigners kurssit

Liite 3 - Tyylitiedosto

```
body{
  font-family: Arial, Helvetica, sans-serif;
  line-height: 120%;
  background-color: #FDF6E7;
  color: #31302E;
  margin: 0px;
}

h1 {
  font-family: Arial, Helvetica, sans-serif;
  font-size: medium;
  font-style: normal;
  color: #BB5E18;
  padding: 7px;
}

h2 {
  font-family: Arial, Helvetica, sans-serif;
  font-size: 14px;
  font-style: normal;
  color: #DE8B4C;
  padding: 7px;
}

h3 {
  font-family: Arial, Helvetica, sans-serif;
  font-size: 12px;
  color: #BB5E18;
  font-style: normal;
  padding: 7px;
}

p {
  padding: 7px;
  font-size: small;
}

a {
  font-size: 12px;
}

a:link{
  font-size: 12px;
  text-decoration: none;
  color: #15558E;
}

a:visited{
  font-size: 12px;
  text-decoration: none;
  color: #F47B2D;
}

a:hover, a:active {
  text-decoration: underline;
  color: #F47B2D;
}
```

```
#container {
    width: 740px;
    margin-right:auto;
    margin-left: auto;
    background-image: url(img/tausta4.jpg);
    background-repeat: repeat;
    background-color: #FFFFFF;
    margin-top: 10px;
    border-left: 1px solid #C4C3C1;
    border-top: 1px solid #C4C3C1;
}

#leftnav {
    float: left;
    width: 170px;
    margin-left: 2px;
    margin-top: 20px;
    background-color: #FFFFFF;
}

#content {
    width: 510px;
    margin-left: 175px;
    border-left: 1px dotted #FDB882;
    background-color: #FFFFFF;
    padding: 10px;
    margin-top: 0px;
    padding: 10px;
}

#footer {
    width: 740px;
    margin-right:auto;
    margin-left: auto;
    margin-bottom: 15px;
    border-left: 1px solid #C4C3C1;
    background-color: #FFFFFF;
}

.keskitys {
    text-align: center;
}

a img{
    border: 0px;
}

.kuva {
    border: 0px;
}

ul {
    margin: 10px;
    padding: 5px;
    list-style-type: none;
    font-size: 12px;
}

.navlist{
    padding: 0;
```

```
list-style-type: none;
width: 734px;
color: #15558E;
background-color: #FFD0AB;
font-family: Arial, Helvetica, sans-serif;
font-size: 12px;
font-weight: bold;
border-bottom: solid 1px #FC7F20;
border-top: solid 1px #FC7F20;
text-decoration: none;
}

.navlist a{
width: 5em;
color: #15558E;
background-image: url(img/navi.jpg);
background-repeat: no-repeat;
background-color: #FFD0AB;
padding: 0.1em 2.5em;
text-align: center;
text-decoration: none;
border-right: solid 1px #FC7F20;
text-decoration: none;
}

.navlist a:hover{
color: #F47B2D;
text-decoration: none;
}
```