

Joonas Manninen

INTERNETSIVUJEN TEKNINEN TOTEUTUS

Kuinka luot internetsivut ja käytät Joomla! -
sisällönhallintajärjestelmää

Opinnäytetyö
Tietojenkäsittely

Maaliskuu 2010




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Opinnäytetyön päivämäärä 8.3.2010
Tekijä(t) Joonas Manninen	Koulutusohjelma ja suuntautuminen Tietojenkäsittelyn koulutusohjelma	
Nimeke Internetsivujen tekninen toteutus - Kuinka luot internetsivut ja käytät Joomla! - sisällönhallintajärjestelmää		
Tiivistelmä <p>Tämä opinnäytetyö kertoo sekä web-sivujen teknisestä toteutuksesta että Joomla! – sisällönhallintajärjestelmästä. Teoriaosuudessa kerrotaan web-sivuilla käytettävistä tekniikoista nyt sekä tulevaisuudessa. Sisällönhallintajärjestelmä on web-sivuissa nykyään yleinen, koska yritykset ja yhteisöt haluavat itse päivittää omia sivujaan ja usein henkilökunnalla ei ole teknistä osaamista web-sivujen koodaamiseen.</p> <p>Käytännön osuudessa toteutin Kangasniemen Seurakunnalle uudet web-sivut valmiin ulkoasu suunnitelman pohjalta. Seurakunnan vaatimuksena oli, että he itse pystyisivät päivittämään sivujen sisältöä ja että sivuilla olisi normaalien tekstisivujen lisäksi myös tapahtumalistausta sekä Kangasniemen seurakunnan kirkkoherran oma blogi.</p>		
Asiasanat (avainsanat) Sisällönhallintajärjestelmä, internet-sivu		
Sivumäärä 31+1	Kieli Suomi	URN URN:NBN:fi:mamk-opinn2010A2261
Huomautus (huomautukset liitteistä)		
Ohjaavan opettajan nimi Arto Väättäinen	Opinnäytetyön toimeksiantaja MyNoste Ky	

DESCRIPTION

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Date of the master's thesis 8 March 2010
Author(s) Joonas Manninen	Degree programme and option Business information technology	
Name of the master's thesis Creating web pages - How to make web pages and use the Joomla! content management system		
Abstract This bachelor's thesis was about creating web pages and using the Joomla! content management system. Theory was about different techniques used on web pages at the present time and in the future. Content management systems are very common nowadays, because companies and groups want to update their websites by themselves, but employees do not usually have skills to update web pages by HTML. The practical part of the study was about creating a new website for the Kangasniemi church from a pre-designed web site layout. The church's requirements were that they would be able to update the site by themselves and that there would not be info about church only, but also a list of happenings and the vicar's blog. The study resulted in completed website currently in use. The church was satisfied with the result.		
Subject headings, (keywords) Content management system, website		
Pages 31+1	Language Finnish	URN URN:NBN:fi:mamk-opinn2010A2261
Remarks, notes on appendices		
Tutor Arto Väätäinen	Bachelor's thesis assigned by MyNoste Ky	

SISÄLTÖ

1	JOHDANTO	1
2	INTERNETSIVUJEN TEKNINEN TOTEUTUS	1
2.1	Kolmikerrosmalli	2
2.2	HTML	4
2.3	XHTML	4
2.4	DTD	5
2.5	CSS	6
2.6	JAVASCRIPT	7
2.7	PHP	9
2.8	Tietokannat ja SQL	10
2.9	Web-sivujen toteutus tulevaisuudessa	11
3	JOOMLA!	12
3.1	ASENNUS JA KÄYTTÖÖNOTTO	13
3.1.1	Artikkelien hallinta	14
3.1.2	Valikkojen hallinta	15
3.2	JOOMLA!:N KUSTOMOINTI	16
4	KANGASNIEMEN SEURAKUNNAN NETTISIVUT	19
4.1	Sivurunko	21
4.2	Sivuston valikkorakenteet	24
4.3	Artikkelit	24
4.4	Artikkelin tulostaminen ja fontin koon muuttaminen	25
4.5	Tapahtumat -moduuli	25
4.6	Kirkkoherran blogi ja kirkkoherra-valikko	26
4.7	Viikko-ohjelma -moduuli	27
4.8	Lähetä viesti-, Sivukartta- sekä Linkit-sivut	28
4.9	Miete-moduuli	29
5	PÄÄTÄNTÖ	30
	LÄHTEET	1

1 JOHDANTO

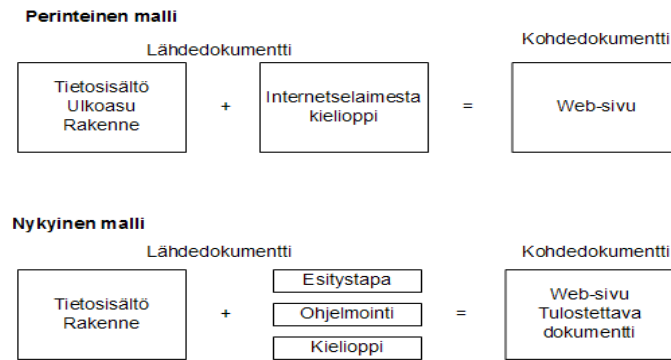
Tämän opinnäytetyön tavoitteena on opastaa lukijaa internetsivujen teknisessä toteutuksessa. Opinnäytetyössä kerrotaan myös kuinka internetsivuissa voidaan hyödyntää Joomla! -julkaisujärjestelmää. Samalla on tavoitteena saada lukija ymmärtämään, miten nykyaikaiset internetsivut tehdään Joomla! -sisällönhallintajärjestelmään. Käytännön osuutena toimivat Kangasniemen Seurakunnan uudet internetsivut, joiden sisältö on tuotettu Joomla!:lla.

Nykyään moni yritys haluaa internetsivujensa olevan helposti heidän itsensä päivitettävissä. Yleensä yrityksellä ei ole käytettävissä ohjelmoijia, vaan sivujen päivitys pitäisi onnistua suoraan tekstieditorilla ilman ohjelmointitaitoja.

Miika Hakala teki opinnäytetyönään Kangasniemen Seurakunnalle internetsivujen ulkoasuehdotuksen. Uusilla sivuilla olisi tarkoitus julkaista kirkkoherran blogi, seurakunnan viikko-ohjelma sekä tulevat tapahtumat. Seurakuntalaisten pitäisi pystyä päivittämään näitä tietoja helposti itse.

2 INTERNETSIVUJEN TEKNINEN TOTEUTUS

Yhdennäköisen web-sivun voi tehdä monella tavalla, vaikka webbi toimiikin yhteisillä standardeilla. Internetsivun voi muotoilla joko perinteisesti HTML -kuvauskielen kommennoilla tai, kuten suurin osa nykyisistä sivustoista on tehty, erityisesti kuvailuun erikoistuneilla kielillä, kuten CSS (HTML -dokumentit) ja XSL (XML -dokumentit). (Linjama, 2001.)



KUVA 1. Web-sivun perinteinen ja uusi rakenne sekä malli

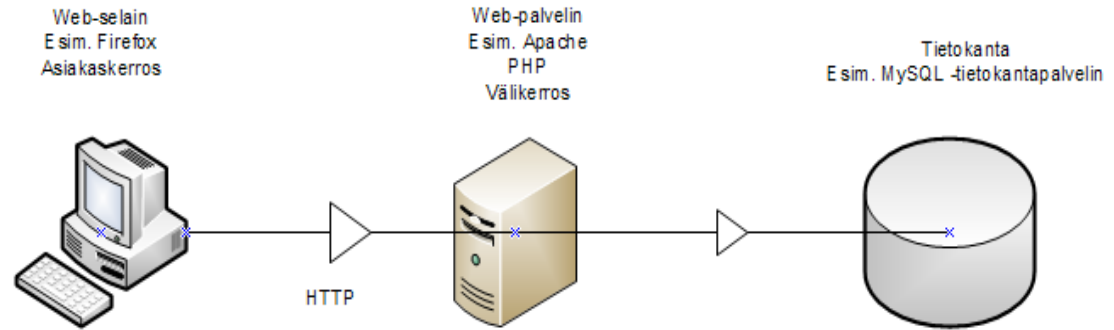
Perinteisen mallin mukaisesti muotoilluissa sivuissa rakenteen kielioppi ja web-sivuston graafinen ulkoasu on määritelty internet-selaimessa. Uuden rakenteen ideana on, että web-sivustoissa voidaan käyttää itse keksittyjä logiikoita, luoda omia uusia elementtejä ja kertoa selaimelle niiden määrittelyt.

Työkalut

Itse käytän webbisivujen tekemisessä ilmaista, avoimeen lähdekoodiin perustuvaa Notepad++ -tekstieditoria. Notepad++:ssa osaa näyttää kätevästi koodin eri osat eri väreillä, joten kokonaisuuksien hallinta pysyy hyvin hanskassa. Ohjelmaan voi myös ladata eri liitännäisiä, jollon se osaa mm. ehdottaa käyttäjälle HTML -tageja käyttäjän syöttämän tekstin perusteella. Notepad++ löytyy osoitteesta <http://notepad-plus.sourceforge.net/uk/site.htm>.

2.1 Kolmikerrosmalli

Dynaamiset sivustot toteutetaan yleensä kolmikerrosmallin mukaisesti. Kolmikerrosmalliin kuuluu asiakaskerros (esim. web-selain), välikerros (web-palvelin) sekä tietokantakerros (esim. MySQL -tietokantapalvelin). (Rantala 2005 254.)



KUVA 2. Kolmikerrosmalli

Internetselain ei yleensä pysty suoraan kommunikoimaan tietokantapalvelimen kanssa, joten selaimen ja tietokannan välissä käytetään välikerrosta, web -palvelinta. (Rantala 2005 254).

Apache web-palvelimen asennus omalle koneelle

Apache web-palvelimen asennus kotikoneelle onnistuu ehkäpä helpoiten WAMP -paketin kautta. WAMP koostuu Windows-, Apache-, MySQL- ja PHP -ohjelmistoista. WAMP -paketin voit ladata esimerkiksi osoitteesta <http://www.wampserver.com/en/download.php>. Itse paketin asennus on erittäin helppo operaatio, käyttäjän ei itse periaatteessa tarvitse kuin naputella "Next" -nappia asennuksen yhteydessä, koska asennusohjelman oletusvalinnat pitäisivät kaikille käyttäjille olla sopivat. Oletusasetuksilla WAMP käyttää Apache -palvelimen web-sivuston hakemistona `c:\wamp\www` -hakemistoa. Kun siis haluat tehdä nettisivuja omalle palvelimellesi, täytyy ne tallentaa tuon hakemiston alihakemistoihin (esim. `c:\wamp\www\testisivusto`). Näitä sivustoja pääset selailemaan web-selaimella osoitteessa `http://localhost/`. Tietokannan hallintaa onnistuu WAMP -paketin mukana tulevalla PhpMyAdmin -ohjelmistolla, johon päästään joko:

- klikkaamalla WAMP -kuvaketta Windowsin tehtäväpalkista hiiren vasemmalla napilla ja valitsemalla PhpMyAdmin
- suoraan internet-selaimella menemällä osoitteeseen `http://localhost/phpmyadmin/`

WAMP -serveri on oletusarvoisesti Offline -tilassa. Jos haluat, että serverillesi pääsee esim. muut koneet lähiverkossa, klikkaa WAMP -kuvaketta tehtäväpalkissa hiiren

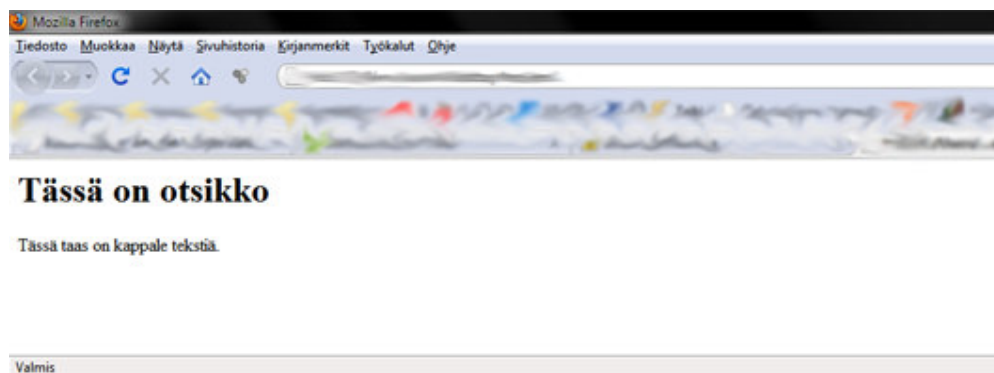
vasemmalla, ja valitse "Put Online". Toimiakseen tämä tosin saattaa vaatia palomuurin asetusten muuttamista.

2.2 HTML

HTML on kuvauskieli, jonka tarkoituksena on "kuvailla" internetsivua, HTML -dokumentti on siis internetsivu. HTML -tiedoston päätte on joko .htm tai .html. HTML on kokoelma erilaisia "tageja", eli elementtejä. Tagit kirjoitetaan kulmasulkeiden sisään, esim. <html> tai <body>. Yleensä HTML -tageilla on alkutagi ja lopputagi, joka vastaavasti kirjoitetaan </html> tai </body>. Internetselain tulkitsee näitä tageja, ja näyttää sivuston tagien määrittelyjen mukaisesti. (Introduction to HTML.)

```
<html>
<body>
  <h1>Tässä on otsikko</h1>
  <p>Tässä taas on kappale tekstiä.</p>
</body>
</html>
```

KUVA 3. Esimerkki HTML -koodista



KUVA 4. Kuvan 3 sivusto Firefox -selaimella katsottuna

2.3 XHTML

XHTML on HTML- ja XML -kielten risteytys. XHTML sisältää normaalin HTML -kielen kaikki komennot, mutta sillä voidaan tehdä myös XML-kielen mukaisesti omia elementtejä. XHTML:llä kirjoitettu dokumentti on käytännössä XML-dokumentti, johon on määritelty kaikki HTML 4.0:n kieliopin säännöt. (Linjama 2001, 125.)

XHTML:llä sivujen tekeminen on hyvin samanlaista kuin HTML:llä. Näkyvimmmät erot ovat, että XHTML:ssä kaikki "tagit" suljetaan. Esim HTML:ssä kuva lisätään -muodossa, XHTML:ssä sama tehdään , samoin HTML:n
 on XHTML:ssä
. Erona HTML:ään on myös se, että tagit kirjoitetaan pienellä, kun ne HTML:ssä voidaan kirjoittaa myös isolla. Kuten HTML -tiedosto, myös XHTML -tiedoston päätte on joko .htm tai .html. (XHTML HowTo.)

XHTML on HTML:n puhtaampi ja tarkempi muoto. HTML:ssä esimerkiksi on mahdollista kirjoittaa lihavoitu ja alleviivattu teksti seuraavasti: <u>Tähän tekstiä</u>. Tämä ei ole sallittua XHTML:ssä, vaan elementtien pitää olla sisäistetty oikein: <u>Tähän tekstiä</u>. (XHTML vs HTML.)

Alla esimerkki yksinkertaisesta XHTML -sivustosta:

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

<html>
<head>
<title>simple document</title>
</head>
<body>
<p>a simple paragraph</p>
</body>
</html>
```

KUVA 5. Esimerkki XHTML -koodista

2.4 DTD

Kuvan 5 kahdella ensimmäisellä rivillä määritellään dokumentin DOCTYPE, joka määrää sen, miten selain sivua käsittelee. DOCTYPE määritellään aina dokumentin ensimmäisellä rivillä. XHTML:lle on olemassa kolme erilaista DOCTYPEä:

- XHTML 1.0 Strict, jota käytetään silloin, kun halutaan kirjoittaa puhdasta XHTML:ää
- XHTML 1.0 Transitional, jota käytetään silloin, kun halutaan käyttää XHTML:n ohessa myös HTML:n esitysominaisuuksia
- XHTML 1.0 Frameset, jota käytetään silloin, kun sivustolla on käytössä HTML -raamit.

W3Schoolsin (XHTML DTD.) mukaan yleisin DOCTYPE on XHTML 1.0 Transitional.

Neljänneltä riviltä alkaa html -elementti, jonka sisään sivuston koodi tulee. Rivillä viisi alkaa head -elementti, (lähde http://www.w3schools.com/tags/tag_head.asp, viitattu 11.2) jonka sisään tulee esim. metatietoa sivusta, linkki tyylitiedostoon, linkkejä scripteihin ja sivuston otsikko. Sivuston otsikko määritellään title -elementillä rivillä kuusi. Kahdeksannelta riviltä alkaa body -elementti, jonka sisään tulee itse sivuston rakenteen koodi, esim. tekstit ja kuvat. Rivillä yhdeksän on määritelty pieni tekstikappale. (XHTML DTD.)

2.5 CSS

CSS on kuvauskieli, jolla kuvataan sivuston ulkoasu. Se ei ota kantaa millään tavalla sivuston sisältöön. CSS -kuvaukset kirjoitettiin aluksi samaan tiedostoon kuin itse HTML, mutta nykyään kaikki selaimet tukevat erillistä tyylitiedostoa. CSS:n tarkoituksena on yksinkertaistaa ja monipuolistaa sivujen ulkoasun tekemistä, mutta sillä on myös haittapuolena se, että nykyselaimet tulkitsevat CSS:ää erittäin vaihtelevasti. Tästä syystä sivut pitäisi suunnitella toimivaksi myös ilman tyylimäärittelyjä. (Linjama 2001, 502.)

CSS sääntö koostuu kolmesta osasta: valitsimesta, ominaisuudesta ja arvosta. Alla esimerkki CSS -koodista (CSS Syntax):

```
/* tähän voidaan kirjoittaa kommentti */
P
{
    font-family: Verdana;
    font-size: 11px;
    color: #000000;
    text-align: left;
    margin-bottom: 15px;
}
```

KUVA 6. Esimerkki CSS -koodista

Rivillä kaksi on valitsin p, eli esimerkikoodissa määritellään tagin <p> -tyyliä.

Rivillä kolme on aloittava aaltosulje, ja rivillä yhdeksän lopettava aaltosulje. Näiden

sulkeiden väliin tulee itse määrittelyt. Määrittelyt kirjoitetaan muodossa "ominaisuus: arvo;", päättyen siis puolipisteeseen. Kuvassa teksti p -elementin sisässä sijoitetaan elementin vasempaan laitaan, teksti on väriltään mustaa, fonttina on verdana, fontin koko on kiinteä 11 pikseliä ja <p>-elementin jälkeen on alhaalla 15 pikseliä tyhjää tilaa. CSS -koodissa kommentit alkavat /* -merkillä ja päättyvät */ -merkkiin. CSS -tiedoston päätte on .css.

2.6 JAVASCRIPT

Javascript on internetin suosituin scriptikieli ja sitä tukee kaikki yleisimmät internetselaimet. Javascript suunniteltiin lisäämään dynaamisuutta internetsivuille. Javascriptiä ei kannata sekoittaa Javaan, joka on täysin itsenäinen ohjelmointialusta, kun taas Javascript on scriptikieli. Javascript on ns. tulkintakieli, eli sitä ei käännetä ennen suorittamista, kuten esim. Javaa tai C -kieliä. (Javascript Introduction.)

W3Schoolsin (Javascript Introduction) mukaan Javascriptiä voidaan hyödyntää internetsivuilla monin tavoin. Sen avulla voidaan esim. tarkistaa onko lomakkeen kaikki pakolliset kentät täytetty, tai onko sähköpostiosoite annettu oikeassa muodossa. Javascript voidaan kirjoittaa joko itsenäiseen tiedostoon, tai suoraan html-sivulle. Itsenäisessä tiedostossa ollessaan linkitys tapahtuu html-sivun head elementin sisässä <script> -tagilla, esimerkiksi <script src="scripti.js" type="text/javascript"></script>. Itsenäisen Javascript -tiedoston päätte on .js. Sillä voidaan myös lukea ja kirjoittaa tietoja HTML -elementeistä.

```
<html>
  <body>
    <script type="text/javascript">
      document.write("<h1>Hei maailma!</h1>");
    </script>
  </body>
</html>
```

KUVA 7. Esimerkki javascript -koodista kirjoitettuna suoraan sivun koodin joukkoon

Kyseinen scripti kirjoittaa sivulle "Hello World!" <h1> -tagin sisään.

HTML Document Object Model (DOM)

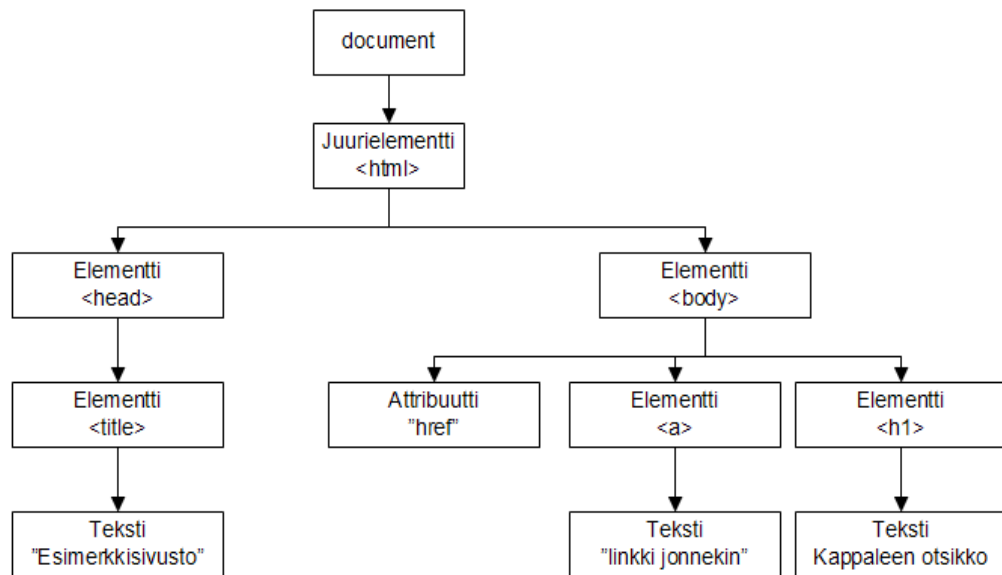
W3C:n määrittelemä DOM-standardi määrittelee ohjelmointikieliriippumattoman ohjelmointirajapinnan HTML- ja XML -dokumentteihin. DOMin avulla voidaan muuttaa dokumentin sisältöä sekä dokumentin elementtien järjestystä ja tyylejä. (sama kuin yllä). (Linjama 2001, 608.)

DOM määrittelee, että jokainen elementti html-dokumentissa on solmu, "node". Itse dokumentti on dokumenttisolmu, elementit ovat elementtisolmuja, tekstielementit ovat tekstisolmuja jne. (HTML DOM Nodes)

```
<html>
  <head>
    <title>DOM Tutorial</title>
  </head>
  <body>
    <h1>DOM Lesson one</h1>
    <p>Hello world!</p>
  </body>
</html>
```

KUVA 8. HTML -koodia

Kuvassa 8 juurisolmuna on <html>. <head>- ja <body>- ovat solmun <html> lapsisolmuja (HTML DOM Nodes).



KUVA 9. Solmupuu

Kuvan 9 avulla on helppo hahmottaa miten eri elementteihin pääsee käsiksi esimerkiksi Javascriptillä. Jos `<h1>` elementin ID olisi määritelty "otsikko":ksi, päästäisiin sen tekstiin käsiksi parillakin eri keinolla:

```
Var teksti = document.getElementById("otsikko").innerHTML;
```

Tai

```
Var teksti =
```

```
document.getElementById("otsikko").childNodes[0].nodeValue;
```

InnerHTML on yleisemmin käytetty vaihtoehto, mutta alempi vaihtoehto on aivan yhtä helppokäyttöinen, ja selventää paremmin solmujen puurakennetta. (HTML DOM Properties and Methods.)

2.7 PHP

Toisin kuin Javascript, PHP on täysiverinen ohjelmointikieli, joka suoritetaan serverillä, ei internetselaimessa. PHP on open sourcea, eli sen käytöstä ei tarvitse maksaa. PHP -tiedoston päätte on joko .php, .php3 tai .phtml, joista .php on yleisimmin käytössä. PHP toimii niin Windows, Linux kuin Unix -järjestelmissä. Toimiakseen PHP vaatii palvelimelle asennetun palvelinohjelman, esim. Apachen. (PHP Introduction.)

PHP -scriptiosuus alkaa <?php ja loppuu ?>, esimerkiksi:

```
<html>
<body>
<?php
    echo "<p>Tekstiä</p>" ;
?>
</body>
</html>
```

KUVA 10. Esimerkki PHP -scriptistä

Kuvan 10 scripti tulostaa sivulle "Tekstiä" <p> -elementin sisään. PHP:ssä voi kommentit laittaa koodin sekaan kahdella eri tavalla:

```
// tähän kommentit
```

Tai

```
/* tähän kommentit */
```

(PHP Syntax.)

PHP -dokumentit ovat html-dokumentteja, joissa on mukana PHP -koodia (Rantala, 2005, 15.). Rantalan mukaan (2005, 15) eräs PHP:n heikkouksista on, että sille ei ole olemassa kunnollista kehitysympäristöä.

PHP:n nykyominaisuudet kestävät vertailun lähes mihin tahansa lausekieliseen ohjelmointikielen. PHP:n virheenkäsittely ja olio-ominaisuudet ovat uusimmassa versiossa lähes samalla tasolla kuin parhaissa olio-ohjelmointikielissä, Javassa ja Pythonissa.

2.8 Tietokannat ja SQL

Tietokantana voidaan pitää lähes mitä tahansa loogisesti yhteenkuuluvaa sekä tallennettavaa tietokokoelmaa. Tietokantana voi olla esimerkiksi joko relaatiotietokanta tai tekstitiedosto.

SQL on kieli jolla voidaan hallita relaatiotietokantoja. SQL:ää käytetään internetsivuilla jonkin serveripuolen ohjelmointikielen kautta, esim. PHP:n. (SQL

Introduction.) Alla esimerkki SQL -lauseesta, jolla haetaan kuvan 11 esimerkkitaulusta kaikkien henkilöiden kaikki tiedot:

```
SELECT * FROM henkilot;
```

Kysely hakee henkilot -taulun kaikki tietueet. (SQL Syntax.)

Relaatiotietokantojen hallinta SQL-kielillä poikkeaa tekstitiedostojen hallinnasta omatekoisin menetelmin erityisesti siinä, että SQL-kieli itsessään tarjoaa välineet tietojen hakuun ja tallennukseen. Relatiotietokanta koostuu relaatioista, eli tauluista. (Rantala, 2005. 252-253.)

taulu: henkilot

id	etunimi	sukunimi	puhelin	email
0	Keke	Rosberg	5555-4343-2332	keke@rosberg.com
1	Mika	Häkkinen	5555-78787-343	mika@hakkinen.com
2	Heikki	Kovalainen	5555-34343-232	heikki@kovalainen.com

KUVA 11. Relatiotietokannan taulu

Tietokannassa voi olla useita tauluja. Taulut koostuvat riveistä, joita kutsutaan tietueiksi. Tauluesityksessä yksi rivi on yksi tietue. Relatiion määrittäviä tietoja (kuvassa 11 esim. etunimi, sukunimi, puhelin) kutsutaan kentiksi. Relatiomalli ei salli kahden identtisen rivin olemassaoloa samassa taulussa, joten yleensä yksi tietueista valitaan pääavaimeksi, jonka perusteella tietueisiin ensisijaisesti viitataan. (Rantala, 2005. 253.) Yleinen tapa on luoda kenttä, johon tietokanta luo automaattisesti juoksevan numeron estäen identtisten tietueiden synnyn.

2.9 Web-sivujen toteutus tulevaisuudessa

Web-sivujen teknisen toteutuksen standardeja (joita W3C kutsuu suosituksiksi) valvoo The World Wide Web Consortium (W3C), joka on kansainvälinen yhteisö. (World Wide Web Consortium, 2010).

HTML5

HTML5:sta lähdettiin kehittämään vuonna 2004 ja se on vielä kehitystasalla. Siitä milloin HTML5 -suositukset otetaan käyttöön, ei ole tietoa. (HTML5, 2010). Ensimmäinen ero HTML4:ään tulee DOCTYPE:n määrittelyssä, joka HTML5:ssä määritellään `<!DOCTYPE html>`. Kaikkia elementtejä ei ole vielä pakollista sulkea, esimerkiksi `` -elementtiä ei ole pakko sulkea, jos seuraava elementti on ``. (HTML5 - pnuuk!, 2009).

HTML5 esittelee myös paljon uusia elementtejä. `<section>` -elementti tulee korvaamaan `<div>` -elementin sivuston rakenteen (kuva 19) määrittelyssä. `<header>` -elementin sisään voidaan kirjoittaa esim. artikkelin otsikko ja päiväys, `<footer>` -elementti taas sisältää esim. tekijänoikeustiedot (HTML5 - pnuuk!, 2009).

Eräs omasta mielestäni mielenkiintoisin uudistus on videoiden lisäys sivuille ilman, että selaimet tarvitsisivat kolmannen osapuolten liitännäisiä. Videot voidaan sisällyttää sivulle `<video>` -tagin sisälle. Nykytuki selaimissa on vielä vaihteleva (johtuen siitä, että standardi ei ole vielä lyöty lukkoon), esimerkiksi Firefox tukee `<video>` -elementille vain Ogg/Theora koodekkia, kun Safari ja Chrome tukevat h.264 koodekkia. Esimerkiksi YouTube on ottanut `<video>` -tagin koekäyttöön. (HTML5-video.)

CSS3

CSS3 esittelee uusia tapoja sivuston muotoiluun. Elementillä voi olla useita eri taustakuvia, tekstille ja elementeille saa määriteltyä varjon, elementeillä voi olla pyöristetyt reunat sekä elementtien taustakuvien kokoa voidaan säädellä dynaamisesti. Kuten HTML5:n tapauksessa, myös CSS3 ei ole tuettu kovin laajasti eri selaimilla. (11 Classic CSS Techniques Made Simple with CSS3, 2009.)

3 JOOMLA!

Joomla! on ilmainen julkaisujärjestelmä, jonka avulla yksityishenkilöt, yhteisöt tai yritykset voivat lisätä tai päivittää sisältöä web-sivuilleen itsenäisesti, ilman ulkopuolista apua. Sisällönhallinta tapahtuu internet-selaimen kautta, erillisiä ohjelmia

ei tarvita. Joomla! asennetaan web-palvelimelle. Joomla! Vaatii toimiakseen PHP ja MySQL tuen palvelimelta. Joomla!-n käyttö onnistuu myös suomenkielellä. (Mikä on Joomla!?, 2005.)

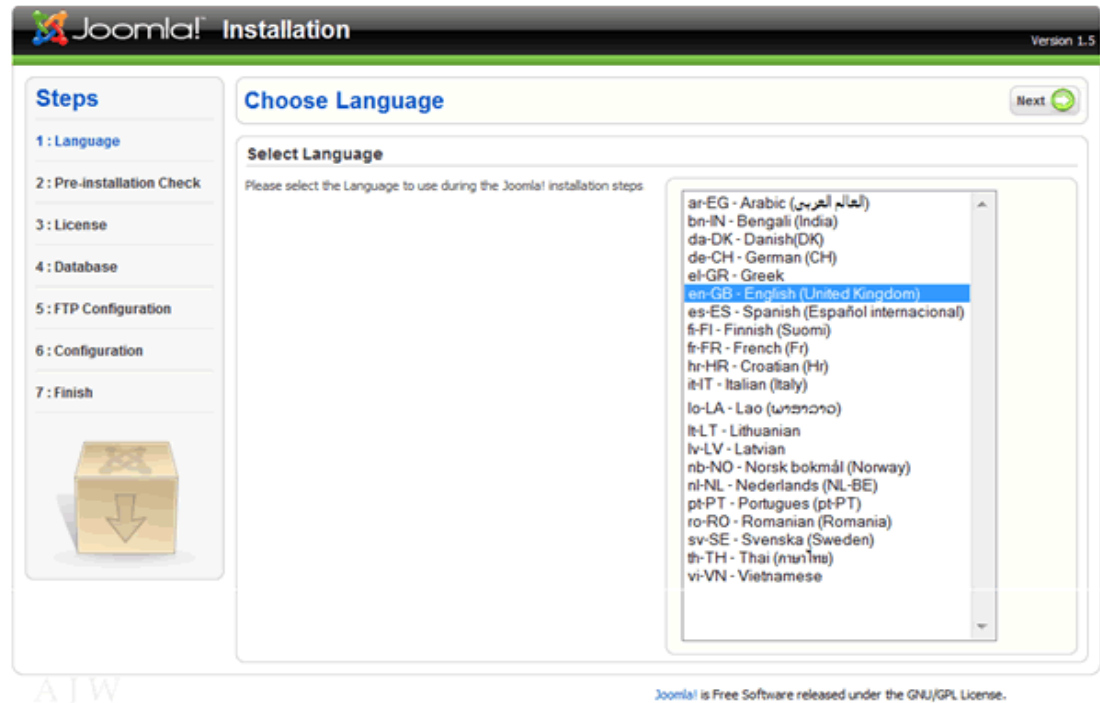
Joomla! soveltuu niin isojen kuin pienienkin internetsivujen julkaisuun. Valmiita sivupohjia (tyylejä) voi ladata netistä joko ilmaiseksi tai maksua vastaan. Yhdelle sivustolle voidaan määritellä monta eri sivupohjia, joten valmiin sivuston ulkoasun muuttaminen onnistuu nopeasti ilman, että sivuston sisältöön tarvitsee tehdä muutoksia. (Mikä on Joomla!?, 2005.)

3.1 ASENNUS JA KÄYTTÖÖNOTTO

Joomla!-n asentaminen on yksinkertainen prosessi. Uusimman Joomla!-n version saa ladattua osoitteesta

http://www.joomlaportal.fi/component/option,com_docman/Itemid,26/. Lataa uusimman version asennuspaketti (Stable full) ja pura se esim. 7zip -ohjelmalla, jonka saa ladattua ilmaiseksi osoitteesta <http://www.7-zip.org/>. Siirrä seuraavaksi tiedostot web -palvelimelle. Palvelimella täytyy olla PHP- sekä MySQL -tuki, jotta Joomla!-a voidaan siinä käyttää.

Siirry seuraavaksi internetiselaimella siihen osoitteeseen, johon siirsit Joomla!-n tiedostot , esim. <http://omapalvelin.com/joomla>. Jos kaikki on tähän asti mennyt oikein, aukeaa selaimen Joomla!-n asennuksen etusivu, jossa pyydetään valitsemaan asennuskieli.



KUVA 12. Joomla! asennusnäky

Vasemmalla näkyvät koko asennuksen ajan asennuksen eri vaiheet. Kielen valitsemisen jälkeen Joomla! tarkistaa palvelintiedot, jonka jälkeen asennus pyytää käyttäjää hyväksymään lisenssiehdot. Seuraavaksi käyttäjän täytyy syöttää tietokantatiedot, jonka jälkeen käyttäjä voi halutessaan määrittää ftp-yhteystiedot palvelimelle. Viimeiseksi asennus pyytää sivuston nimen ja antaa mahdollisuuden esimerkkiartikkelien asennukseen. Asennuksen lopuksi käyttäjää pyydetään poistamaan "installation" -kansio palvelimelta.

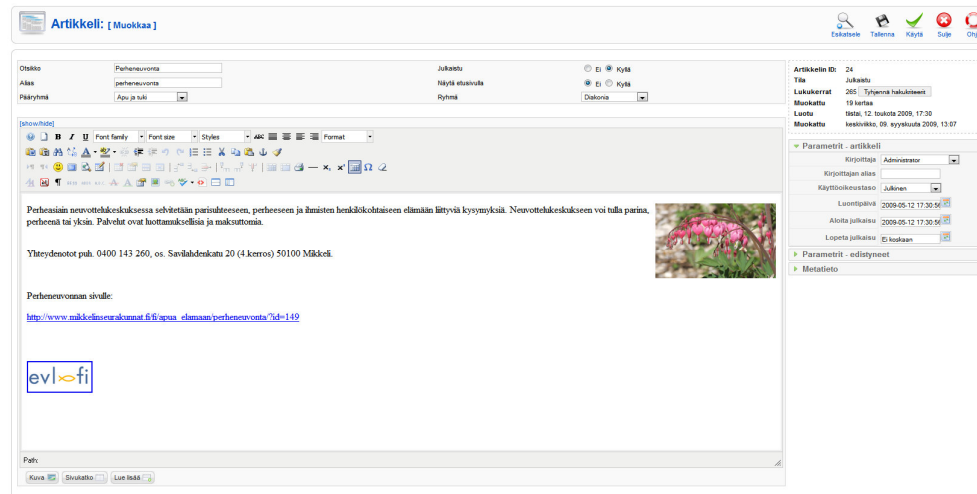
Joomla!:n hallintapaneelin osoite on muotoa

<http://tahanpalvelimesiosoite/administrator>. Joomla!:n hallintapaneelinäkymä on varsin selkeä, ja sen saa vaihdettua suomenkieliseksi netistä ladattavalla kielipaketilla.

3.1.1 Artikkelien hallinta

Nettisivujen sisältö Joomla!:ssa kirjoitetaan artikkeleiksi. Artikkelit kuuluvat ryhmiin, jotka kuuluvat pääryhmiin.

Artikkelien hallintaan Joomla!-n mukana tulee ihan hyvätasoinen peruseditori, jolla saa tuotettua sisältöä nettisivuille niin tekstinä kuin kuvinakin. Suosittelen kuitenkin asentaman JCE WYSIWYG editorin Joomla!-an. Siinä on Joomla!-n perusversioon verrattuna reilusti lisää ominaisuuksia, esimerkiksi tekstin muotoilu onnistuu huomattavasti monipuolisemmin JCE:llä.



KUVA 13. Näkymä JCE -editorin artikkelin muokkauksesta

Artikkelin hallintanäkymä on varsin selkeä. Yläosassa määritellään artikkelin otsikko, onko artikkeli julkaistu vai ei, mihin ryhmään artikkeli kuuluu sekä näytetäänkö artikkeli etusivulla. Näkymän oikealla puolella Määritellään yksityiskohtaisesti artikkelin parametrit ja metatiedot. Oletusarvoisesti artikkelilla on ne parametrit, mitä yleisistä asetuksista on asetettu artikkelille, mutta jokaiselle artikkelille voidaan määritellä myös omakohtaiset parametrit. Esimerkiksi julkaisun aloitus- ja loppumispäivän voi määritellä, sekä mitä tietoja (esim. kirjoittaja, luontipäivä, artikkelin nimi) artikkelista näytetään käyttäjälle.

3.1.2 Valikkojen hallinta

Joomla!-lla tehtyjen nettisivujen navigointivalikoita hallitaan Joomla!-n Menu Managerilla. Valikot otetaan käyttöön Joomla!-n valikkomodulilla. Menu Managerilla määritellään valikoiden linkitykset, valikkomodulin asetuksista määritellään mihin valikko tulee ja miltä se näyttää.

Menu Item Manager: [mainmenu]

#	Menu Item	Default	Published	Order	Access Level	Type	ItemID
1	Etusivu	★	✓	6	Public	Articles » Front Page	1
2	Tuotteet		✓	7	Public	Articles » Article	27
3	Maatalousneuvos		✓	1	Public	Articles » Category	53
4	Kolmas taso		✓	1	Public	Articles » Article	54
5	Lisää otsikoita		✓	2	Public	Articles » Article	57
6	Agri-neuvos		✓	2	Public	Articles » Article	58
7	Metsä-neuvos		✓	3	Public	Articles » Article	71
8	Nautaneuvos		✓	4	Public	Articles » Article	72
9	Cashman		✓	5	Public	Articles » Article	73
10	Palvelut		✓	8	Public	Articles » Article	2
11	Myynti		✓	9	Public	Articles » Section	37
12	Nettitausta		✓	10	Public	Articles » Article	62
13	Demonstraus		✓	11	Public	Articles » Article	63
14	Yritys		✓	12	Public	Articles » Article	64

KUVA 14. Näkymä Menu Managerista

Module: [Edit]

Save Apply Close Help

Details

Module Type: mod_mainmenu
 Title: Main Menu
 Show Title: No Yes
 Enabled: No Yes
 Position: paanavi
 Order: 0>Main Menu
 Access Level: Public
 ID: 1
 Description: Displays a Menu.

Menu Assignment

Menus: All None Select Menu Item(s) from the List

Menu Selection: mainmenu
 Etusivu
 Tuotteet
 Maatalousneuvos
 -Kolmas taso
 -Lisää otsikoita
 Agri-neuvos
 Metsä-neuvos
 Nautaneuvos
 -Cashman
 Palvelut
 Myynti
 Nettaustausta
 Demonstraus
 Yritys

Parameters

Module Parameters

Menu Name: mainmenu
 Menu Style: List
 Start Level: 0
 End Level: 1
 Always show sub-menu items: No Yes
 Target Position:
 Advanced Parameters
 Other Parameters

KUVA 15. Näkymä valikon moduulin asetuksista

3.2 JOOMLA!N KUSTOMOINTI

Moduulit

Joomla! Koostuu erilaisista moduuleista. Moduulit mahdollistavat erilaisia toimintoja sivuille, esim. navigointivalikko, kyselyt ja uutisikkunat. Moduulit määritellään sivupohjaan eri paikkoihin. Moduulien paikat määritellään Joomla!n version 1.5 tyyliohjelmistoon seuraavasti:

```
<jdoc:include type="modules" name="valikko" />
```

(Joomla! sanasto, 2008.)

Komponentit

Toisin kuin moduulit, jotka ovat pieniä toiminnallisuuksien lisäyksiä Joomla!:an, komponentit ovat suurempia ohjelmistokokonaisuuksia. Esimerkiksi keskustelualue (foorumi) on komponentti.(Joomla! sanasto, 2008.)

Tyylipohjat

Käyttäjä voi kirjoittaa itse omat tyylipohjansa Joomla!:n sivuille. Tyylipohjan voi kirjoittaa myös Joomla!:n hallintaosiolle. Tyylipohjan tekeminen Joomla!:aan on muuten samanlaista kuin tavallisen internetsivuston tekeminen, mutta sisältöpaikkoihin (esim. päävalikko, asiatekstikenttä) määritellään moduulipaikat.

Tyylipohja muodostuu vähintään kahdesta tiedostosta: index.php ja templateDetails.xml. Index.php -tiedostoon tulee tyylipohjan X/HTML -koodi. templateDetails.xml-tiedostoon tulee tyylipohjan tiedot xml -muodossa, jossa määritellään kaikki tiedostot, joita tyylipohja käyttää.

```
<install version="1.5" type="template">
  <name>Tyylipohja</name>
  <version>1.0</version>
  <creationDate>11.12.2009</creationDate>
  <author>Joonas Manninen</author>
  <copyright>GNU/GPL</copyright>
  <authorEmail>joonas.manninen@mynoste.com</authorEmail>
  <authorUrl>http://mynoste.com</authorUrl>
  <description>Testaus tyylipohja</description>
  <files>
    <filename>index.php</filename>
    <filename>templateDetails.xml</filename>
    <filename>template_thumbnail.png</filename>
    <filename>css/tyyli.css</filename>
    <filename>images/kuva.jpg</filename>
  </files>
</install>
```

KUVA 16. Esimerkki templateDetails.xml -tiedostosta

Kuvassa 16 on esimerkki templateDetails.xml-tiedostosta. Ensimmäisellä rivillä määritellään Joomla!:n versio, johon tyylipohja on tehty. Seuraavaksi määritellään

tyylijohjan nimi, versio, luomispäivä, tekijä, tekijänoikeus lisenssin määrittely, tekijän yhteystiedot sekä lyhyt kuvaus tyylijohjasta. Lopuksi määritellään tiedostot, joita tyylijohja käyttää. Kuvasta selviää myös Joomla!:n käyttämä hakemistorakenne, jota on aina noudatettava. CSS -tiedostot tulevat tyylijohja/css -hakemistoon, Javascript tiedostot tyylijohja/js -hakemistoon ja kuvat tulevat tyylijohja/images -hakemistoon. Tyylijohja/html -hakemistoon voidaan määrittellä komponentti- tai moduulikohtaisia muokkauksia, jotka ovat kuitenkin jo edistyneempiä ominaisuuksia. templateDetails.xml-tiedosto kannattaa pitää aina ajan tasalla tyylijohjaa tehdessä, koska tyylijohjan asennus ei onnistu tai tyylijohja ei näy oikein, jos tiedostossa ei ole määritelty kaikkia käytössä olevia tiedostoja tai jos tiedostoon on määritelty sellaisia tiedostoja, joita ei ole olemassa.(Creating Joomla 1.5 Templates, 2009.)

Seuraavassa esimerkki yksinkertaisesta Joomla! -tyylijohjan index.php -tiedostosta:

```
<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted access' );
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="<?php echo $this->language; ?>" lang="<?php echo $this->language; ?>" >
<head>
<jdoc:include type="head" />
<link rel="stylesheet" href="<?php echo $this->baseurl
?>/templates/<?php echo $this->template ?>/css/tyyli.css" type="text/css" />
</head>
<body>
<div id="header">
<jdoc:include type="modules" name="paanavi" />
</div>
<div id="asiateksti">
<jdoc:include type="component" />
</div>
</body>
</html>
```

KUVA 17. Joomla! -tyylijohjan index.php -tiedosto

Aluksi määritellään, että tiedostoon ei pääse suoraan käsiksi, esimerkiksi kirjoittamalla osoitekenttään <http://palvelin.com/templates/tyylijohja/index.php>. <jdoc:include type="component" /> -koodi lisää sivulle sisällön. Siihen tulostuu esimerkiksi valittu artikkeli, komponentti tai sivukartta.

4 KANGASNIEMEN SEURAKUNNAN NETTISIVUT

Kangasniemen Seurakunnan internetsivut olivat uudistuksen tarpeessa. Miika Hakala teki opinnäytetyönään seurakunnan graafisen ohjeiston uusiksi, samoin ulkoasusuunnitelman uusista internetsivuista. Seurakunta katsoi, että ulkoasuehdotuksen pohjalta voisi toteuttaa uudet sivut. Toimeksiannon suoritti kesällä 2009 MyNoste Ky, jonka toinen omistaja on Miika ja toinen on allekirjoittanut.

Kangasniemen SEURAKUNTA

Etusivu Toiminta Apu ja tuki Toimipaikat Hallinto Kirkolliset toimitukset Yhteystiedot

Kangasniemen seurakunnan verkkosivusto Maanantai 15.2.2010

Matti Tolvanen
Kirkkoherra

Tervehdys
Blogi

Liitykirkkoon.fi

Jumala,
me kerromme temppelissä
sinun armollisista teoistasi.

Ps. 48 : 10

Lapset ja perheet Tytöt ja pojat NUORET AIKUISET

Viikko-ohjelma

15.2 MA 09:00 ▶ 5v. Päiväkerho

13:00 ▶ Suurojan ja Vihikankaan lähetspiirien LASKIAISTAPAHTUMA

13:00 ▶ Kirkonkylän lähetspiiri

13:30 ▶ Ystävänpäivän laulu ja hartaustuokio vanhustentalon SPR:n toimitiloissa

14:00 ▶ Kaiken kansan virsilaulajaiset

18:00 ▶ Naisten piiri

18:30 ▶ Jumppaa ja gospelia Kalliolan koululla

16.2 TI 09:00 ▶ 3-4v. Päiväkerho

Tapahtumat
Kaikki

Yksi nimi toivomme on -tilaisuus	21.2
Vanhemman väen virkistyspäivä	25.2
Rauhanyhdistyksen seurat	7.3
Vanhemman väen virkistyspäivä	11.3
Lähets tapahtuma	18.3
Vanhemman väen virkistyspäivä	25.3
Armolahjakkuri	27.3
Sanan ja sävelen ilta	9.4

Valitse ryhmä ?
- Kaikki -

Lähetä viesti

Sivukartta

Linkit

myNoste

KUVA 18. Kuvankaappaus valmiin sivuston etusivulta

Uuden sivuston vaatimusmäärittelyt ja käyttötapauskaaviot oli luonnollisesti tehty jo ulkoasun suunnittelun aikoihin, joten teknisen toteutuksen lähtökohdat olivat hyvällä pohjalla. Ensiksi piti valita tehtävään sopiva julkaisujärjestelmä. Vaatimuksina olivat julkaisujärjestelmälle seuraavat:

- Ilmainen, eli ilmaiseen lähdekoodiin perustuva
- Järjestelmän hallintakielenä suomi

- PHP:lla ja MySQL:lla toimiva

Loppujen lopuksi vaihtoehtoja ei ollut kuin Joomla!. Tämä siitä syystä että olin kyseiseen julkaisujärjestelmään tutustunut aikaisemminkin ja sille oli olemassa paljon netistä löytyviä oppaita. Vaihtoehtoja Joomla!:lle olivat lähinnä Wordpress sekä oman alustan koodaus alusta asti. Näiden vaihtoehtojen soveltaminen olisi omalla kohdallani tuonut vain lisää työtä ja täten vaatinut enemmän aikaa verrattuna Joomla!:an.

Wordpress

Wordpress on ilmainen, vapaaseen lähdekoodiin perustuva, alun perin blogien kirjoittamiseen tarkoitettu sisällönhallintajärjestelmä. Wordpress julkaistiin vuonna 2003, jonka jälkeen siitä on kehittynyt täysiverinen julkaisujärjestelmä, jonka avulla voi luoda vähän isommatkin web-sivut. Kuten Joomla!:aa, voidaan Wordpressiä laajentaa ilmaisilla laajennuksilla. Wordpress on toteutettu PHP- ja MySQL-tekniikoilla. (Wordpress > About.)

Oma alustan ohjelmointi

Yksi vaihtoehto olisi ollut luoda oma sisällönhallintajärjestelmä. Tällaisen luominen aivan perusominaisuuksilla olisi ollut suhteellisen yksinkertaista, kun ihan kaikkea ei lähde keksimään itse. Esimerkiksi tekstieditorin saa valmiina (esim. TinyMCE), jolloin käyttäjä voi muotoilla tekstiä haluamallaan tavalla. Itse tehdyllä sisällönhallintajärjestelmällä pitäisi olla seuraavat ominaisuudet:

- sisällön kirjoittaminen, muokkaus ja poistaminen
- kirjoitettujen artikkelien listaus käyttäjälle
 - artikkelit pitäisi pystyä listaamaan eri määrittelyjen mukaisesti, jotta artikkelien hallinta onnistuisi jos artikkeleita on esimerkiksi satoja
- kirjoitettujen artikkeleiden organisointi eri ryhmiin
- kirjoitettu artikkeli pitäisi pystyä määrittelemään joko julkaistavaksi, tai piilotetuksi
- tietoturvallinen.

Oman arvioni mukaan tällaisen systeemin testikäyttöön luominen veisi noin kaksi kuukautta. Kolmannen kuukauden voisi varata järjestelmän testaamiseen ja testeissä

löytyvien bugien (virheiden) korjaamiseen. Tästä syystä ei mielestäni ollut kannattavaa lähteä omaa järjestelmää tekemään, koska aikaa seurakunnan sivuston tekemiseen oli varattu kaksi kuukautta, ja tuo arvioimani aikataulu sisällönhallintajärjestelmän tekemiselle ei juurikaan anna marginaalia viivästymisille.

Joomla!-ssa on myös mahdollisuus vaihtaa ulkoasua lennosta, joten jos Kangasniemen seurakunta haluaa tulevaisuudessa sivuilleen uuden ilmeen, täytyy sivuille tehdä vain ulkoasuun kohdistuvia muutoksia, itse sisältöön ei tarvitse koskea millään tavalla.

Sivujen tekninen toteutus lähti liikkeelle perinteisellä kynä ja paperi tekniikalla, jossa paperille tulostettiin kuva sivujen ulkoasusta, ja paperille vedettiin viivoja viivoittimen avulla, osoittamaan miten kuvat leikataan Adoben PSD –tiedostosta. Viivoitettua kuvaa katsomalla saadaan selville myös sivujen runko. Tässä vaiheessa kuva viivoitettiin vasta osoittamaan sivuston yleispiirteet, esim. valikkojen nappilinkkejä ei viivoitettu.

Sivuston teossa koodin kirjoittamiseen käytin ilmaista Notepad++ -tekstieditoria, kuvankäsittely hoitui Adoben Photoshop CS4:lla.

4.1 Sivurunko

Kangasniemen Seurakunnan sivuston runko on suhteellisen yksinkertainen.

- taustavarjot
 - ulkoasun nitoja
 - sivuston yläosa, "header".
 - päävalikko
 - "headerin" alla oleva osuus, johon tulee napit tulostukselle ja fontin koon muuttamiselle, sekä osoitepolku (breadcrumbs).
 - keskiosa
 - vasen alavalikko
 - asiasisältö, johon tulee esille artikkelit
 - asiasisällön "header"
 - itse asiasisällön teksti
 - tapahtumalistaus

- oikea alavalikko
- sivuston alaosa, "footer".

```

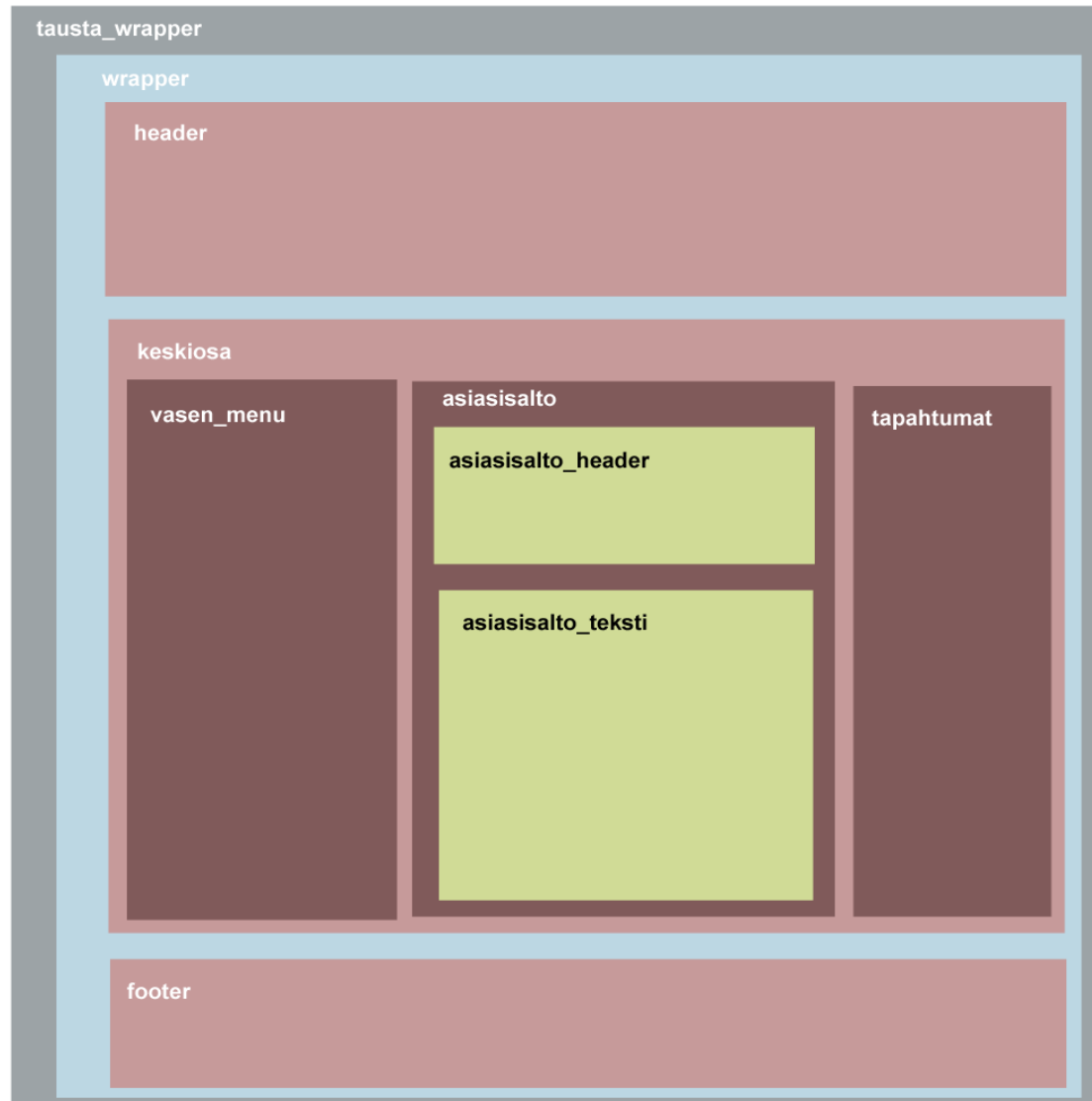
<html>
  <body>
    <div id="tausta_wrapper">
      <div id="wrapper">
        <div id="header">
          <div id="menu">
          </div>
        </div>
        <div id='below_header'>
        </div>
        <div id="keskiosa">
          <div id="vasen_menu">
          </div>
          <div id="asiasisalto">
            <div id='asiasisalto_header'>
            </div>
            <div id="asiasisalto_teksti">
            </div>
            <div id="tapahtumat">
              <div id="tapahtumalistaus">
              </div>
              <div id="oikeavalikko-erotin">
              </div>
            </div>
            <div id="oikea_navi">
            </div>
          </div>
        </div>
        <div id="footer">
        </div>
      </div>
    </div>
  </body>
</html>

```

Kuva 19. Seurakunnan sivurunko XHTML -koodina

Sivuston koodirunko rakennetaan <div> -elementeillä, joiden tunnisteet (id:t) nimetään mahdollisimman kuvaaviksi. Jotta taustavarjot saadaan sivuston rungon sivulle, täytyy sivuston runko upottaa "tausta_wrapper" -nimisen <div> -elementin sisälle. Itse "wrapper" -niminen elementti pitää kasassa sivuston sisältöosion, jossa "header" -nimiseen elementtiin sijoitetaan sivuston yläosan vaihtuva kuva, sekä päänavigointi. "keskiosa" -niminen elementti pitää kasassa sivuston keskiosan, jossa "vasen_menu" -elementin sisään tulee päävalikon alavalikko. "asiasisalto" -elementin

sisälle tulee "asiasialto_header", joka pitää sisällään artikkelin otsikon, tulostusnapin sekä fontin koon muuttamisnapit. "asiasialto_teksti" -elementtiin tulee itse artikkelin teksti. "tapahtumat" -elementin sisälle tulee tapahtumalistaus sekä oikea pikavalikko, "footer" -elementin sisään tulee Kangasniemen Seurakunnan yhteystiedot.



KUVA 20. Seurakunnan sivuston sivurunko

Kun kuvat oli saatu leikattua, koodirungon kirjoitus oli nopeaa. Koodirungon kirjoituksen jälkeen tein CSS –määrittelyt. Kun XHTML sekä CSS –koodit oli kirjoitettu, piti sivuston ulkoasun toimivuus testata sekä Internet Explorer 6:lla, 7:lla, 8:lla että Firefox 3:lla. Kun sivusto näytti yhtenäiseltä kaikilla eri selaimilla, oli edessä itse sisällön lisääminen sivuston niihin osiin, joissa se pysyy vakiona kaikilla sivuilla paitsi etusivulla. Tähän kuului linkki header-kuvaan ”Kangasniemen Seurakunta” –

tekstin kohdalle, osoitepolku (breadcrumbs) headerin alla olevaan tilaan, päivämäärä samaan kohtaan, oikean puolen pikalinkit sekä sivuston alaosan (footerin) tekstit ja linkit. Breadcrumbit tulevat sivustolle suoraan Joomla!n omana moduulina, joten se ei tarvinnut muuta kuin pistää oikeaan kohtaan sivustoa ja tehdä tarvittavat määrittelyt tyylitiedostoon. Päivämäärän tulostukselle kirjoitin pienen php-scriptin.

4.2 Sivuston valikkorakenteet

Sivustolla on kaksi valikkoa, päävalikko ja alavalikko. Päävalikko on näkyvissä joka sivulla sivuston yläosassa vaakasuorassa. Alavalikko on näkyvissä niillä sivuilla, joihin pääsee käsiksi päävalikosta ja itse alavalikosta. Alavalikko sijaitsee sivuston vasemmassa laidassa pystysuorassa.

Valikkorakenteen luominen sujui Joomla!lla kätevästi niin, että ensin rakennettiin yksi yhtenäinen valikko alavalikkoineen, jonka jälkeen ensimmäisen tason valikko sijoitettiin oikealla kohdalle ja alavalikko sivuston omalle paikalleen vasempaan laitaan. Jotta alavalikko saadaan piiloon etusivulla ja näkyviin alasivuille, tarvittiin sitä varten oma moduuli. Tämä tehtiin kopioimalla päävalikon moduuli, ja määriteltiin kopioitu moduuli esiintymään vain kun päävalikosta siirrytään toisen tason sivuille. Kumpaankin valikkoon määriteltiin näkymään aktiivinen sivu erilaisella grafiikalla. Kun valikot toimivat käytännössä oikein, otin kynän, paperin ja viivoittimen jälleen käteeni, ja suunnittelin valikkokuvien leikkaukset. Sen jälkeen leikkasin kuvat ja tein tarvittavat määrittelyt tyylitiedostoon.

4.3 Artikkelit

Kun valikot näyttivät kaikilla selaimilla yhtenäisiltä, kirjoitin materiaalit artikkeleiksi. Pelkän tekstin lisäksi artikkeleihin tuli jokaiseen oma kuvansa, sekä linkit eri yhteisöjen sivuille. Eri artikkeleissa oli eri yhteisöjä, joten tätä linkitystä ei voinut tehdä itse sivurungon koodiin. Artikkelien asiasisällön luominen sujui nopeasti, mutta oli yllättävän vaikeaa saada kuvat ja linkit näkymään oikein. Tämä piti loppupelissä tehdä Joomla!n omalla artikkelin html –editorilla, koska graafinen puoli toimi aika kömpelösti.

4.4 Artikkelin tulostaminen ja fontin koon muuttaminen

Seuraavaksi oli vuorossa fontin koon muuttamiset sekä tulostusnappi. Tulostusnappi toimii Joomla!:n oman "Luo PDF" –linkin avulla, eli tämä osuus hoitui helposti lähes pelkän css:n avulla. Fontin koon muuttamiseen Joomla:ssa ei ole omaa moduulia, vaan koodi piti itse kirjoittaa Javascriptillä. Tein ensin Javascript –koodin, jonka jälkeen sijoitin linkit oikeille kohdilleen. Scripti vaikuttaa ainoastaan fontin kokoon itse artikkeleissa, muun sivuston fontti on vakio. Fontin koon muuttamiset ja artikkelin tulostus eivät ole näkyvissä etusivulla.

4.5 Tapahtumat -moduuli

Sitten ryhdyin tapahtumat –moduulin työstöön. Tarkoituksena oli saada tehtyä sellainen systeemi, johon seurakuntalaiset pystyvät lisäämään, muokkaamaan ja poistamaan tapahtumia. Tapahtumat piti pystyä luokittelemaan eri luokkiin (esim. aikuiset, diakonia jne). Otin tähän tarkoitukseen käyttöön Joomla!:n kirjoitetun eventlist –moduulin. Tällöin tapahtumien hallinta hoitui helposti Joomla!:n oman admin –paneelin kautta, eikä omaa tarvitsisi kirjoittaa. Tapahtumien listaus käyttäjälle olikin tehtävä itse, koska moduulin oman listauksen ulkoasu on suoraan kivikaudelta.

Tapahtumat

YKSI NIMI TOIVOMME ON -TILAISUUS

Päivämäärä: 21.02.2010

Aika: 16:00 -

Paikka: Kirkko

KUVAUS

Mukana Sari Behm, Heikki Mikkonen ja Pirkka Sieviläinen.

Tapahtumat
Kaikki

Yksi nimi toivomme on -tilaisuus	21.2
Vanhemman väen virkistyspäivä	25.2
Rauhanyhdistyksen seurat	7.3
Vanhemman väen virkistyspäivä	11.3
Lähetystapahtuma	18.3
Vanhemman väen virkistyspäivä	25.3
Armolahjakurssi	27.3
Sanan ja sävelen ilta	9.4

Valitse ryhmä ?

- Kaikki -

Lähetä viesti

Sivukartta

KUVA 21. Tapahtumalistaus ja tapahtuman tarkempi kuvaus

Kirjoitin moduulin niin, että se listaa kahdeksan tulevaa tapahtumaa aikajärjestyksessä. Listauksen alla on alavetolaatikko, josta voi valita mitkä tapahtumat tulostetaan. Moduuli tulostaa etusivulla oletusarvoisesti kaikkien luokkien tapahtumat, ja mentäessä esim. Aikuiset –sivulle moduuli tulostaa oletusarvoisesti Aikuiset–luokan tapahtumat. Tulostuksessa tapahtumista luodaan linkit, joita klikkaamalla avautuu tarkemmat tiedot tapahtumista. Tapahtumien tarkempien tietojen tulostus tapahtuu moduulin omalla systeemillä, siihen vaihdoin ainoastaan hieman ulkoasua tyylitiedostoa muokkaamalla.

4.6 Kirkkoherran blogi ja kirkkoherra-valikko

Seuraavaksi oli edessä kirkkoherran blogi-järjestelmän tekeminen sekä vasemman puolen ”kirkkoherra”-valikon rakentaminen. Valikko esiintyy etu-, Kirkkoherran tervehdys -, tapahtuman tarkempia tietoja luettaessa, lähetä viesti-, sivukartta- sekä linkit –sivuilla. Kirkkoherran blogi rakentuu suoraan Joomlaan omalla moduulilla. Kirkkoherran blogi –linkistä päästään sivulle, jossa listataan blogit –ryhmästä artikkelit blogimuodossa. Neljä viimeisintä bloggausta tulostetaan kokonaan, seuraavista muutama rivi ja lopuista pelkät otsikot. Blogiin liittyy kommentointimahdollisuus, jonka toteutin ulkoisen moduulin, !JoomlaCommentin, avulla. Tämä moduuli on siitä kätevä, että sen voi liittää mihin tahansa artikkeleihin ja komponentteihin, joten tarvittaessa kaikki sisältö sivuilla on kommentoitavissa, jos niin halutaan.



KUVA 22. Kirkkoherran blogin kommentointinäkömä

4.7 Viikko-ohjelma -moduuli

Seuraavaksi piti sivustolle rakentaa viikko-ohjelma moduuli. Tämä oli helpointa tehdä samalla eventlist –moduulilla, kuin tapahtumatkin. Eventlistiin tein omat ryhmänsä, erottaen ne vo–etuliitteellä tapahtumista. Etusivulla tulostetaan kaikkien luokkien viikko-ohjelmat, ja jokaisen luokan (esim. musiikki) alasivulle mentäessä tulostetaan kyseisen luokan viikko-ohjelma. Otin tulostukseen mukaan jQuery–kirjaston ja siihen tehdyn jExpand –lisäosan. Tämän avulla tulostukseen saadaan lisäulottuvuutta niin, että ensin tulostetaan viikko-ohjelmaan tapahtumien päivämäärä, viikonpäivä, kellonaika sekä otsikko. Tätä tulostusriviä klikattaessa tuodaan esille tapahtuman tarkemmat tiedot. Näin ei tarvitse käyttäjää ohjata erilliselle sivulle tarkempia tietoja varten, eikä oletusarvoisesti tarvitse tulostaa kaikista tapahtumista kaikkia tietoja näytölle kerralla. Ikävä kyllä ominaisuus ei Internet Explorer 8:ssa toimi kunnolla. Ongelmana on se, että kun käyttäjä klikkaa tarkemmat tiedot esiin, niitä ei saada enää piiloon ilman, että sivu ladataan uusiksi.




15.2.	MA	14:00	▶ Kaiken kansan virsilaulajaiset
17.2.	KE	17:30	▶ EI LAPSIKUORON HARJOITUKSIA
18.2.	TO	17:30	▶ Kirkkokuoron harjoitukset
24.2.	KE	17:30	▶ Lapsikuoron harjoitukset
3.3.	KE		▶ Ei lapsikuoron harjoituksia
4.3.	TO		▶ Ei kirkkokuoron harjoituksia
10.3.	KE	17:30	▶ Lapsikuoron harjoitukset

KUVA 23. Viikko-ohjelma näkymä Musiikkiosiosta

4.8 Lähetä viesti-, Sivukartta- sekä Linkit-sivut

Lähetä viesti -sivun toteutin Joomla!n omana artikkelina syöttäen lomakkeen koodin artikkeliin Joomla!n omalla html -editorilla. Viesti voidaan lähettää kerralla vain yhdelle vastaanottajalle. Vastaanottajavaihtoehtoja on useita ja vastaanottaja valitaan pudotusvalikosta. Käyttäjä voi valita, haluaako viestiinsä vastauksen vai ei. Lomakkeessa on myös "Tyhjennä" -toiminto, jolloin kaikki lomakkeen kentät tyhjennetään.

Lähetä viesti

Valitse viestin vastaanottaja tai palautteen aihe:

Haluan viestiin vastauksen (sähköpostiosoite annettava)
 En halua viestiin vastausta

Lähettäjän nimi

Lähettäjän sähköpostiosoite

Viestin aihe:

Viesti:

KUVA 24. Lähetä viesti -sivu

Sivukartta toteutettiin Xmap -komponentilla. Komponentin asetuksista voidaan valita, mistä ryhmästä tulostetaan artikkelit. Komponentin ulkoasua ei voi suoraan hallintajärjestelmän kautta juurikaan muokata, joten jouduin ulkoasumuutokset tekemään suoraan Xmapin omaan tyylitiedostoon.

Linkit sivustolle koottiin linkejä erinäisille sivustoille. Lista toteutettiin Joomla!n omalla artikkelieditorilla.

4.9 Miete-moduuli

Seuraavaksi sivustolle piti rakentaa etusivulle mietemoduuli. Moduulin tarkoituksena on, että seurakuntalaiset kirjoittavat joka viikko mietteen. Mietteet tulostetaan diaesitys-tyyliin etusivulla, oletusarvoisesti esillä on viimeisin miete. Mietteestä

näytetään etusivulla muutama eka rivi ja linkki, josta mietteen pääsee lukemaan kokonaan. Sekä etusivun mietetulosuksessa että mietteen tarkemmassa tulostuksessa on seuraava- ja edellinen –linkit, joilla käyttäjä voi selata mietteitä. Mietteet kirjoitetaan aivan normaaleina Joomla!-n artikkeleina.

Viimeisenä oli vuorossa etusivulle pikalinkkien lisäys, sekä vaihtuva kuva, jossa on lainaus raamatusta. Kuvan vaihtumiseen tein pienen php -scriptin, joka näyttää sivulla kuvan kuukauden perusteella. Pikalinkit tulevat suoraan kuvina viikko-ohjelman ylle.

5 PÄÄTÄNTÖ

Nykyiset yritysten, yhteisöjen ja jopa yksityisten henkilöidenkin web-sivut tehdään useasti julkaisujärjestelmään, jotta he itse voivat päivitellä tietoja sivuillensa. Kangasniemen seurakunnan tapauksessa jopa päivittäistä päivittämistä sivuille tuovat seurakunnan viikko-ohjelman ja tulevien tapahtumien listaus sivuilla. Nykyaikana ihmiset haluavat internet-sivuilta vuorovaikutusta, tätä nälkää ruokkii Kangasniemen seurakunnan kirkkoherran blogi, jota kirkkoherra päivittää kuukausittain ja jota sivuilla kävijät voivat kommentoida.

Sivujen toteutus oli iso projekti. Kun sivuston ulkoasu oli saatu toimimaan Mozilla Firefoxilla, piti sivusto vielä saada näyttämään samalta Microsoftin selainvanhuksella, Internet Explorer 6:lla, jota vielä yllättävän iso osa ihmisistä käyttää. Tietotekniikan nopea kehitys on johtanut myös siihen, että web-sivuja selataan niin pieniresoluutioisilla kämmentietokoneilla kuin suuri-resoluutioisilla näyttöillä. Kaikilla näillä laitteilla sivuston pitäisi näyttää samalta, ja tämä tuo mukanaan haasteita tekniseen toteutukseen, koska websivustot suunnitellaan yleensä mainostoimistojen toimesta, joilla ei välttämättä ole silmää web-sivujen teknisen toteutuksen rajoituksille.

Jälkikäteen on myös huomattu, että vaikka Joomla! tarjoaa hyvän tekstieditorin sisältöjen tuottamiseen, saattaa sillä tuotettu sisältö olla muotoilultaan aivan jotain muuta kuin tarkoitus on, jos teksti on suoraan kopioitu esim. Microsoft Wordista.

HTML:n rajoituksista johtuen web-sivujen sisällön muotoilua ei vielä ole saatu samalle tasolle kuin modernien tekstinkäsittelyohjelmien.

Joomla!:n valmiit komponentit, kuten Eventlist ja Xmap, mahdollistivat sen, että itse ei tarvinnut aivan joka asiaa lähteä koodaaman alusta asti. Riitti että muokkasi tulostusnäkyvän HTML -koodia sekä muutteli tyylitiedoston määrittelyksiä. Kun vertaa omaa tietämystäni Joomla!:sta projektin alussa ja lopussa, huomaa, että tietämykseni kyseisestä julkaisujärjestelmästä on aivan uudella tasolla. Projektin ansiosta voin soveltaa Joomla!:a myös tulevissa projekteissa.

LÄHTEET

Anton, Vasilios 2009. 11 Classic CSS Techniques Made Simple with CSS3. WWW-dokumentti. <http://net.tutsplus.com/tutorials/html-css-techniques/11-classic-css-techniques-made-simple-with-css3/>. 8.9.2009. Luettu 21.2.2010.

Creating Joomla! 1.5 Templates. Packt Publishing 2008. WWW-dokumentti. http://www.themeswiki.org/Creating_Joomla_1.5_Templates. 6.7.2009. Luettu 19.2.2010.

HTML DOM Nodes. Refsnes Data 2010. WWW-dokumentti. http://www.w3schools.com/html/dom/dom_nodes.asp. Ei päivitystietoa. Luettu 7.2.2010.

HTML DOM Properties and Methods. Refsnes Data 2010. WWW-dokumentti. http://www.w3schools.com/html/dom/dom_methods.asp. Ei päivitystietoa. Luettu 7.2.2010.

Introduction to HTML. Refsnes Data 2010. WWW-dokumentti. http://www.w3schools.com/html/html_intro.asp. Ei päivitystietoa. Luettu 5.2.2010.

Javascript Introduction. Refsnes Data 2010. WWW-dokumentti. http://www.w3schools.com/js/js_intro.asp. Ei päivitystietoa. Luettu 7.2.2010.

Joomla! sanasto - Joomla!portal.fi. Joomla!portal.fi 2008. WWW-dokumentti. <http://www.joomla!portal.fi/content/view/453/44/>. 29.1.2008. Luettu 14.2.2010.

Linjama, Tero 2001. XHTML. Gummerus Kirjapaino Oy.

Mikä on Joomla!? - Joomla!portal.fi. Joomla!portal.fi 2005. WWW-dokumentti. <http://www.joomla!portal.fi/content/view/93/39/>. 17.11.2007. Luettu 14.2.2010.

Perälä, Juha 2010. HTML5-video. WWW-dokumentti.
<http://juhaperala.net/blog/2010/02/01/html5-video/>. 1.2.2010. Luettu 21.2.2010.

PHP Introduction. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/php/php_intro.asp. Ei päivitystietoa. Luettu 11.2.2010.

PHP Syntax. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/php/php_syntax.asp. Ei päivitystietoa. Luettu 11.2.2010.

Rantala, Ari 2005. Web-ohjelmointi. Porvoo: WS Bookwell.

SQL Introduction. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/sql/sql_intro.asp. Ei päivitystietoa. Luettu 12.2.2010.

SQL Syntax. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/sql/sql_syntax.asp. Ei päivitystietoa. Luettu 12.2.2010.

XHTML DTD. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/xhtml/xhtml_dtd.asp. Ei päivitystietoa. Luettu 5.2.2010.

XHTML HowTo. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/Xhtml/xhtml_howto.asp. Ei päivitystietoa. Luettu 5.2.2010.

XHTML vs HTML. Refsnes Data 2010. WWW-dokumentti.
http://www.w3schools.com/xhtml/xhtml_html.asp. Ei päivitystietoa. Luettu 5.2.2010.

Tirkkonen, Tomi 2009. HTML5 - pnuk!. WWW-dokumentti.
<http://pnuk.net/arkisto/2009/07/10/html-5-2/>. 10.7.2009. Luettu 21.2.2010.

W3C, 2010. HTML5. WWW-dokumentti.
<http://dev.w3.org/html5/spec/Overview.html>. 18.2.2010. Luettu 21.2.2010.

Wordpress > About. WordPress.org 2010. WWW-dokumentti.
<http://wordpress.org/about/>. Ei päivitystietoa. Luettu 21.2.2010.

Liite 1. Kangasniemen seurakunnan web-sivujen käyttötapauskaavio.

