

Tampereen ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma  
Timo Hainimo

Opinnäytetyö

## PIENTEN KUSTANNUSTEN KEVYET JA ULTRAKEVYET WWW-SISÄLLÖNHALLINTAOHJELMAT

Työn ohjaaja      Petri Heliniemi  
Tampere 5/2009

Tampereen ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma, Digimedia

Tekijä(t)	Timo Hainimo
Työn nimi	Pienten kustannusten kevyet ja ultrakevyet WWW-sisällönhallintaohjelmat
Sivumäärä	38
Valmistumisaika	toukokuu 2009
Työn ohjaaja	Petri Heliniemi

---

## TIIVISTELMÄ

Työllä ei ole toimeksiantajaa, koska yhteistyöni Katrium OÜ:n kanssa oli alkanut jo paljon ennen loppu-työprosessin aloittamista. Yritykselle tekemäni WWW-sivujen päivitystyön aikana syntyi kuitenkin ajatus oman WWW-sisällönhallinnan tietämyksen syventämisestä, ja päätin kirjoittaa myös loppu-työn aiheeseen liittyen.

Työn tavoitteena on ollut hankkia kattava yleiskuva WWW-sisällönhallintaohjelmista, joista koituu käyttäjälleen erityisen vähän kustannuksia. Näkökulma on siis lähinnä pienyrityksille ja yksityishenkilöille sopiva. Aihealue keskittyy vain kevyihin ja ultrakevyihin WWW-sisällönhallinnan ohjelmiin. Ohjelmia tutkitaan uuden käyttäjän silmin, pitkälti käyttöönoton ja käytön opiskelun kannalta.

Aloittaessani työn tekemistä tietoni WWW-sisällönhallintaohjelmista olivat todella vähäiset, vaikka minulla olikin kattavat taidot esimerkiksi XHTML:stä ja CSS-tyylitiedostoista. Tarvittava tietous hankittiin pääosin ohjelmien käyttöä harjoittelemalla ja online-ohjemateriaaleihin tutustumalla. Yleiskuva WWW-sisällönhallinnan ohjelmiin on pyritty hankkimaan neljää esimerkkiohjelmaa käyttäen, jotka edustavat alansa suosituimpia ohjelmia. Lähdekritiikistä sanottakoon, että lähteinä on käytetty pääosin ohjelmien valmistajien itse WWW-ympäristössä julkaisemia dokumentteja. Muussakin lähdemateriaalissa, ja etenkin online-materiaalissa, käytäntö on yleensä jälkikäteen varmistanut tietojen paikkansapitävyyden.

Työn päätuloksena syntyi yleiskuva kevyistä ja ultrakevyistä WWW-sisällönhallintaohjelmista. Työn lopputulokset ovat hyödynnettävissä eräänlaisena tutustumisena WWW-sisällönhallinta- eli CMS-ohjelmien maailmaan. Lisäksi työhön liittyen syntyivät Katrium OÜ:n uudet WWW-sivut ja heille suunniteltu ohjeistus WWW-sivujen päivittämisestä.

---

Avainsanat: WWW-sisällönhallinta, sisällönhallintaohjelmat, CMS

TAMK University of Applied Sciences  
Business Information Systems, Digital Media

Writer(s)	Timo Hainimo
Thesis	Small-cost light- and ultralight WWW-content management systems
Pages	38
Graduation time	May 2009
Thesis Supervisor	Petri Heliniemi

---

## ABSTRACT

This degree work has no principal, because my cooperation with Katrium OÜ had started a lot earlier than this degree work's process. Anyway, the idea of enhancing my own knowledge in WWW content management systems was born while doing the updating process with Katrium's WWW content.

The goal of the work has been to collect an exhaustive view on WWW content management systems, which cause really small costs, or no costs at all, to the developer. The point of view is mainly suitable for small companies and private persons. Theme of this work focuses only in light- and ultralightweight WWW content management programs. The programs are examined from a new user's view, mainly focusing in installation and practising the basic features.

When starting this process, my own knowledge in WWW content management systems was really minor, even though I had comprehensive skills with, for example, XHTML and CSS (Cascading style sheets). The needed knowledge was mostly collected by practising the use of selected programs and reading online tutorials. Overall picture of WWW content management programs was created by using four example programs, which represent the most popular programs in their own categories. About the criticism on source material, I would like to mention that most of the source material are the programs' developers own tutorials, published in WWW format. Also with other source material, and especially with online material, practical experience often proved the arguments true later on.

As a main result of the work, an overall picture of light- and ultralight WWW content management systems was created. The outcome of the work can be used as an introduction to the world of WWW content management systems (CMS). As a by-product of this work were created Katrium OÜ's new WWW-pages and a guide to updating WWW content with FTP connection for them.

---

Keywords: WWW content management, content management systems, CMS

# Sisällysluettelo

1 Johdanto	5
2 Peruskäsitteitä	7
2.1 Ohjelman ja järjestelmän ero	7
2.2 Sisällönhallinta- ja julkaisujärjestelmä	7
2.3 CMS	8
2.4 FTP	8
2.5 PHP	8
3 Sisällönhallinta- eli CMS-ohjelmat	9
3.1 Sisällönhallintaohjelmista yleisesti	9
3.2 Ultrakevyet sisällönhallintaohjelmat	11
3.2.1 Weebly	11
3.2.2 Doodlekit	15
3.3 Kevyet sisällönhallintaohjelmat	18
3.3.1 Joomla!	19
3.3.2 ModX	22
4 Muita tapoja helpottaa sivuston päivitettävyyttä	25
4.1 FTP-synkronointi	25
4.2 Yrityskohtaisesti suunniteltavat, CMS-elementtejä sisältävät ratkaisut	26
5 Oma sivustoprojekti	27
6 Yhteenveto ja johtopäätökset	30

# 1 Johdanto

Alun perin tarkoitukseni oli tässä työssä keskittyä vain yhteen CMS-sisällönhallintaohjelmaan, jota olisin käyttänyt omassa WWW-sivustoprojektissa. Tavoitteet kuitenkin muuttuivat sivuston luonnin yhteydessä, ja tuli selväksi, että ainakaan vielä en ajan puutteen vuoksi päässyt käyttämään CMS-järjestelmää kyseisen sivun yhteydessä, vaan turvauduin toistaiseksi paljon yksinkertaisempiin ratkaisuihin.

WWW-julkaisuun tarkoitetut sisällönhallintajärjestelmät kuitenkin kiinnostivat, koska itselläni on pitkä henkilöhistoria Internet-sivujen luonnin ja päivittämisen parissa. Taidot olivat jääneet kohtuullisen pitkältä ajalta päivittämättä, ja vaikka olin hyvin sinut esimerkiksi XHTML:n ja CSS:n kanssa, olivat CMS-järjestelmät minulle paljon vieraampi aihealue.

Päätinkin pitää ajankohtaiset ja jatkuvasti kehittyvät CMS-järjestelmät työni aiheena ja ottaa niihin laajemman näkökulman, jossa vertailen joitakin alan sovelluksia ja arvioin niiden etuja toisiinsa nähden. Päätin myös keskittyä kevyemmän luokan järjestelmiin, joita on mahdollista käyttää yritysmaailmassakin ilmaiseksi tai erittäin huokeaan hintaan verrattuna raskaiden kaupallisten sovellusten kustannuksiin.

Työssä vertaillaan ensin kahta ultrakevyttä CMS-vaihtoehtoa ja niiden jälkeen kahta kevyttä open source -vaihtoehtoa. Yhdessä luvussa kerron myös omasta sivustoprojektistani, ja siitä, mikä johti sen osalta toistaiseksi jättämään CMS-järjestelmän käyttöönoton myöhemmäksi, mahdollisesti tämän työn tulosten pohjalta tehtäväksi toimenpiteeksi. Lisäksi käsitellään lyhyesti joitakin muita tapoja helpottaa sivuston päivitettävyyttä.

Neljän sisällönhallintajärjestelmän käsitteleminen voi tuntua suppealta, kun sitä vertaa ohjelmien tarjontaan, joka on erittäin laajaa. Tarjonnan laajuutta kuvaa hyvin se, että yksi lähteistäniikin löytyvä Internet-sivu listaa 50 parasta ultrakevyttä CMS-sovellusta. Siinä on siis vain 50 parasta, puhumattakaan koko tarjonnasta tai kaupallisten, raskaampien sovellusten määrästä (*OpenJason: 50 Content Management Systems: 2008*). Tässä työssä on siis kuitenkin tarkoitus keskittyä kevyihin ja erittäin matalat kustannukset tuoviin järjestelmiin. Neljä ohjelmaa on valittu pitkälti luetun käyttäjäpalautteen perusteella

sekä eri Internet-arvosteluissa saadun tunnustuksen perusteella. Nämä neljä järjestelmää edustavat ehdottomasti oman kategoriansa käytetyimpiä ja parhaita ohjelmia, ja kuten monissa muissakin ohjelmatyypeissä, myös CMS-järjestelmissä pätee pitkälti se sääntö, että kun oppii yhden ohjelman käyttölogiikan, on helppo omaksua muitakin alan ohjelmia. Luonnollisesti yhtä järjestelmää käsitellään työssä vain rajalliselta syvyydeltä.

## 2 Peruskäsitteitä

Tässä luvussa käyn läpi työn ymmärtämisen kannalta tärkeimpiä ja yleisimmin käytettyjä käsitteitä. Selitän myös joitakin toisiinsa helposti sekoittuvia termejä.

### 2.1 Ohjelman ja järjestelmän ero

Järjestelmä tarkoittaa lähtökohtaisesti kokonaisuutta, joka hallinnoi resursseja. Perusperiaate ohjelmalle sen sijaan on se, että ohjelma voi olla mikä tahansa käskyjen kokoelma, jonka tietokone osaa suorittaa jonkinlaista logiikkaa käyttäen. Järjestelmä tuo väistämättä mieleen ohjelmaa suuremman kokonaisuuden, ja tälle oletukselle on olemassa vankka totuus pohja. Järjestelmät ovat hallinnointilähtökohdistaan johtuen lähes poikkeuksetta suuria kokonaisuuksia. Ohjelmat sen sijaan voivat olla kooltaan oikeastaan missä tahansa mittasuhteissa. Sisällönhallinnan maailmassa puhutaan yleisemmin järjestelmistä kuin ohjelmista. Syykin tälle on selvä, onhan sisällönhallinta nimenomaan sisältökokonaisuuden hallintaa, huolimatta siitä, minkälaisesta sisällöstä on kyse. Tämän työn WWW-sisällönhallinnan maailman puitteissa olen ottanut vapauden käyttää sekä ohjelma- että järjestelmä-termejä mielestäni niille sopivilla paikoilla.

### 2.2 Sisällönhallinta- ja julkaisujärjestelmä

Tässä työssä käytän pääosin termiä WWW-sisällönhallintajärjestelmä, joka on usein myös synonyymi yrityksen julkaisujärjestelmälle. Sisällönhallintajärjestelmä on itsessään paljon laajempi termi, mutta puhuttaessa WWW-sisällönhallinnasta, tarkoitetaan useimmiten lähinnä sitä materiaalia, jonka yritys tai yksityishenkilö julkaisee WWW-ympäristössä. Sisällönhallintaa hahmottaessa kannattaa aina muistaa, että myös rakenne on sisällön käytettävyyden kannalta ratkaiseva tekijä, ja WWW-sisällönhallinta sisältää paljolti rakenteen suunnittelua (Boiko 2005: 65).

## 2.3 CMS

CMS on lyhenne sanoista Content Management System, ja se on kattoterminä kaikille sisällönhallinnan tietojärjestelmille. CMS-termiä voidaan siis käyttää niin WWW-sisällönhallinnan yhteydessä, kuin vaikka dokumenttienhallintaan tai muuhun aineistonhallintaan liittyen. CMS-nimikkeeseen alle mahtuu todella monenlaisia tietojärjestelmiä niin koon kuin ominaisuuksienkin puolesta.

## 2.4 FTP

FTP on lyhenne sanoista File Transfer Protocol. Se on TCP-protokollaa käyttävä tiedonsiirtomenetelmä kahden tietokoneen välille muodostettavia yhteyksiä varten. FTP:llä yhteys muodostetaan yleensä siten, että toinen kone on asiakas ja toinen palvelin.

Perinteinen FTP-yhteyden muoto on salaamaton, mutta uudet palvelin- ja asiakasohjelmistot osaavat hyödyntää salauksia sekä kontrolli- että siirtoyhteyksissä. Yleisin tapa FTP-yhteyden salaamiseen on FTP-yhteyden tunnelointi SSH-yhteyden avulla.

## 2.5 PHP

PHP on lyhenne sanoista PHP: Hypertext Preprocessor, ja se on erityisesti WWW-ympäristöjä varten suunniteltu ohjelmointikieli. Kielenä PHP on alustariippumaton.

Yleisimmin PHP-koodia käytetään HTML/XHTML-koodiin upotettuna. PHP:n päätarjoitus on auttaa dynaamisen WWW-sisällön luomisessa. (*PHP: Manual: 2009*).



## 3 Sisällönhallinta- eli CMS-ohjelmat

Perehtymiseni ja käyttökokemukseni CMS-ohjelmista sisältyvät tähän lukuun. Ensin käsittelen sisällönhallintaohjelmia yleisesti, ja sen jälkeen keskitytään vuorotellen testauksessa olleisiin esimerkkiohjelmiin.

### 3.1 Sisällönhallintaohjelmista yleisesti

Sisällönhallintaohjelmia on olemassa monenlaisia sisältöjä varten, mutta tässä työssä keskitytään WWW-sisällön hallitsemista, eli käytännössä WWW-sivujen luomista ja päivittämistä helpottaviin sovelluksiin. Ennen CMS-ohjelmien tuloa WWW-sisällön tekemiseen oli vain kaksi tapaa; käsin koodaamalla tai jotakin sivujen tekoon tarkoitettua ohjelmaa (esimerkiksi Frontpage) käyttäen. Yksi suurimmista CMS-ohjelmien tuomista hyödyistä on se, että työvaihe, jossa tiedostot siirretään palvelimelle, puuttuu kokonaan sivujen päivittäjän näkökulmasta. Tiedostojen siirtovaiheen puuttuessa monet sellaiset ihmiset, joille muuten tuottaisi suuria vaikeuksia tuottaa sisältöä internetiin, pääsevät helpommin aloittamaan WWW-ympäristön käytön jakelukanavana. Yksi hyvä esimerkki tämän asian hyödyntämisestä on opetusympäristö ja vaikkapa Moodle, joka on pesunkestävä CMS-sovellus.

CMS-ohjelmat voidaan jakaa niiden käyttötarkoitusten mukaan kategorioihin, ja yleisin tapa jakaa ohjelmat luokkiin on niiden koko, joka käytännössä tarkoittaa sitä mittakaavaa, kuinka laajassa, vaativassa ja suuressa ympäristössä ohjelman tulee pystyä toimimaan ongelmitta. Tässä jaossa ohjelmat jaetaan luokkiin ultrakevyt, kevyt, keskiraskas ja raskas.

Ultrakevyen ja kevyen sisällönhallintajärjestelmän suurimpana erona on se, että ultrakevyt järjestelmä tarjoaa yleensä kehittäjänsä puolelta ylläpitopalvelun, jolloin käyttäjän ei tarvitse ymmärtää palvelinpuolen asioita. Kevyt sisällönhallintajärjestelmä pystyy ultrakevyen lailla käsittelemään vain pienehköjen yritysten tai muiden instanssien tietomääriä, mutta ultrakevyen sijaan kevyt ohjelma täytyy ainakin pystyä asentamaan palveli

melle. Kevyellä ohjelmalla voi olla, ja luultavasti myös on, muitakin vaatimuksia toimintaympäristöstään (esimerkiksi tietokannat), joiden hallitseminen on perusedellytys järjestelmän toiminnalle.

Mitä pienemmän mittakaavan sisällönhallintajärjestelmästä on kyse, sitä enemmän se perustuu yleensä WWW-julkaisutarpeeseen. Niinpä keskiraskaat ja etenkin raskaat sisällönhallintajärjestelmät usein nivoutuvat enemmän yhteen suurten organisaatioiden muiden tietojärjestelmien (usein esimerkiksi toiminnanohjausjärjestelmä) kanssa. Näissä järjestelmissä WWW-ulosanti on todellakin vain pala kokonaisuutta, ja niitä muokataan paljon sopimaan suuryrityksen tarpeisiin. Myös kustannukset ovat aivan eri luokkaa kuin kevyissä ja ultrakevyissä järjestelmissä. Keskiraskaita ja raskaita järjestelmiä ei kuitenkaan ole nyt tarkoitus käsitellä laajemmin, vaan tässä työssä keskitytään vain ultrakevyisiin ja kevyisiin open source -pohjaisiin järjestelmiin.

On olemassa myös muita tapoja jakaa sisällönhallintaohjelmia luokkiin. Yksi yleinen luokkajako on seuraava: flatfile-CMS, tietokanta-CMS ja portaali-CMS. Tietyllä tapaa tämäkin jako muistuttaa paljon yllä mainittua luokkajakoa, mutta tässä jako tehdään puhtaasti sen perusteella, millä tavalla sisällönhallintaohjelma käsittelee tietoa. Flatfile tarkoittaa sanana tiedostomuotoa, johon on tallennettu vain yksinkertaista tekstiä. Flatfile-cms-järjestelmässä kaikki tieto (esimerkiksi asiasisältö ja layout-elementit) on tallennettu tekstitiedostoihin, jotka CMS-ohjelma yhdistää kokonaisuudeksi. Nimensä mukaisesti tietokanta-CMS:t käyttävät tietojen tallentamiseen tietokantaa. Portaali-CMS taas tarkoittaa nimenomaan palveluntarjoajan portaalin kautta toimivia sovelluksia. Tämä jako ei ehkä ole aivan yhtä kattava ja käyttökelpoinen kuin ensin mainittu (*CMS Learning: 2009*).

Näiden luokkien ulkopuolella voidaan järjestelmistä erotella avoimen lähdekoodin vaihtoehdot, eli englanninkielistä termiä käyttäen open source -järjestelmät. Raskautensa puolesta avoimen lähdekoodin järjestelmät sijoittuvat enimmäkseen kevyiden järjestelmien tasolle. Avoin lähdekoodi ei oikein sovellu ultrakevyiden järjestelmien konseptiin selkeän ristiriitaisuutensa takia. Vaikka ultrakevyiden järjestelmien tuotot valmistajalleen perustuvatkin yleensä ylläpitoon, voisi silti olla outoa laskuttaa ohjelmasta, jota käyttäjät vapaasti kehittävät. Raskaiden järjestelmien vaatimukseen avoimen lähdekoo

din ohjelmat taas eivät yleensä pysty vastaamaan, mutta osaavissa käsissä parhaat open source -järjestelmät pystyvät tapauskohtaisesti korvaamaan keskiraskaita järjestelmiä.

## 3.2 Ultrakevyet sisällönhallintaohjelmat

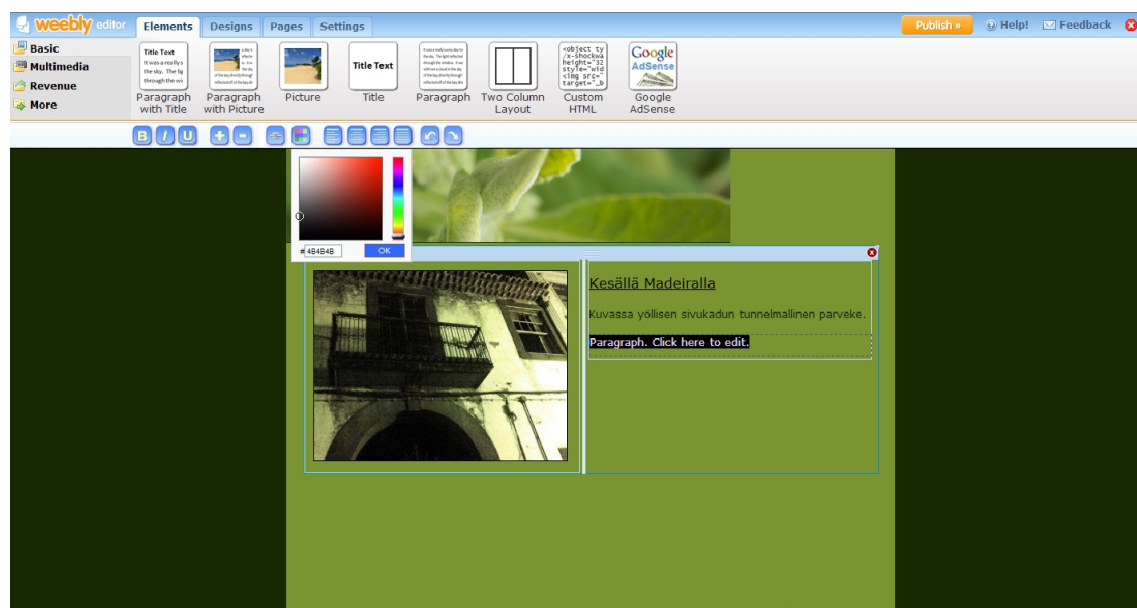
Ensin käydään läpi kaksi ultrakevyttä WWW-sisällönhallintaohjelmaa. Lähtökohtaisesti ultrakevyet sisällönhallintaohjelmat soveltuvat parhaiten esimerkiksi niille yrityksille, joille on hyötyä näkyvyydestä Internetissä, mutta mahdollista sisältöä on vähän ja asia halutaan hoitaa niin, ettei siihen kulu paljon omia resursseja. Ultrakevyetkin ohjelmat onnistuvat silti välillä hämmästyttämään monipuolisuudellaan.

### 3.2.1 Weebly

Weebly on toinen arvioitavaksi valitsemistani helppokäyttöisimmistä eli ultrakevyistä CMS-ohjelmista. Valitsin ohjelman pitkälti sen saaman hyvän palautteen ansiosta. Weebly on täysin web-pohjainen, joten mitään ohjelmia tai niiden osia ei tarvitse ladata omalle koneelle. Weeblyn käyttö alkaa nopeasti ja vaivattomasti. Heti ohjelman etusivulla voi keksiä oman käyttäjänimen ja salasanan, joiden lisäksi tarvitaan vain toimiva sähköpostiosoite. Lisäksi ohjelma kysyy otsikkoa, jolla nimetään ensimmäinen luotava oma sivusto.

Sisäänkirjautumisen jälkeen päästään omalle sivulle (MyWeebly). Omalta sivulta pääsee luomaan omaa sivustoa, tai jos sellainen on jo luotu, pääsee sitä muokkaamaan. Sivustoja voi myös olla useita, jolloin ne löytyvät omalta sivulta listattuina.

Siirryttäessä itse sivuston muokkaustilaan, näkyy yläpalkissa neljä välilehteä: elements, designs, pages ja settings. Näillä kaikilla voi tehdä muutoksia vapaassa aikajärjestyksessä, eli esimerkiksi muuten valmiin sivun ulkoasun voi käydä vaihtamassa toiseen designs-välilehdellä. Nämä välilehdet vievät vain pienen osan selainikkunasta, muu osa on muokattavaa sivua. Kuvassa 1 on nähtävillä sivun muokkaustila.



**Kuva 1: WWW-sivun sisällön rakentamista Weeblyssä**

Elements-välilehti sisältää nimensä mukaisesti kaikki sivun elementit sekä myös niiden asettelun eli sivun ulkoasullisen rakenteen. Peruselementtejä ovat tekstikehys, kuva ja otsikko, sekä näiden yhdistelmät. Lisäksi peruselementeistä löytyy mahdollisuus muuttaa layout yhdestä kolumnista kaksijakoiseksi, custom html (eli oman HTML-koodin tuominen) ja Google AdSense. AdSense on ohjelma, joka tuo sivustolle ulkopuolisia mainoksia (tosin hillittyjä), joiden avulla ylläpitäjä voi ansaita. Elementit tuodaan sivulle yläpalkista raahaamalla, ja oletuksena niitä laitetaan sivulle allekkain. Elementtien laittaminen allekkain tuntuu aluksi rajoittavan mahdollisuuksia sivun rakenteessa, mutta esimerkiksi kahden sarakkeen käyttö ja oman koodin tuominen sivuille laajentavat mahdollisuuksia.

Elements-välilehdeltä löytyy myös muita elementtejä kuin peruselementit (basic). Seuraava kategoria on multimedia, joka sisältää kuvan ja tekstikehysen lisäksi valmiin kuvagallerian lisäämisen ja ladattavan tiedoston lisäämisen (isommat tiedostokoot vaativat maksulliseen ohjelmaversioon siirtymisen). Erilaisia soittimia löytyy myös. Valmis mahdollisuus on audiosoitimen lisäämiseen (taustamusiikki sivuille, vaatii maksullisen version). Valmiit liitäntämahdollisuudet löytyvät myös Youtuben ja Googlen videotiedostojen toistamiseen, Flickr-palvelun kuvasarjoille ja Google Maps -kartoille.

Revenue-osa on tarkoitettu rahaa tuottaville elementeille, joten se sisältää uudestaan Google AdSense -palvelun käyttömahdollisuuden, joka onkin palvelun ainoa valmistyökalu rahan ansaitsemiseen. Viimeinen eli More-välilehti sisältää huomattavasti enemmän elementtejä, osa on tarpeellisia ja osa vähemmän tarpeellisia. Uusia elementtejä ovat layout-puoleen liittyen Divider, joka on tarkoitettu tilan jakamiseen, Fun-osiosta löytyy ristinantehtäviä sekä sivustolle upotettavia pelejä, Misc-osiosta uusia ovat yhteydenottolomake ja Feed Reader, jonka avulla sivustolle voi tuoda sisältöä blogeista.

Design-välilehdellä pääsee valitsemaan sivun tyylin ja värimaailman, eli aika pitkälti sitä, mitä käsin XHTML-sivua kirjoitettaessa tehdään CSS-tiedostolla, poislukien elementtien sijoittelu sivulla. Designin voi valita kymmenistä valmisvaihtoehdoista, joita voi vielä muokata Advanced Editing -tilassa, jos XHTML- ja CSS-aidot ovat kunnossa. Valmisvaihtoehtojen kirjo on laaja; löytyy aikuismaisempia ja ammattimaisempia, toisaalta värikkäämpiä ja hauskenpia, eri aloihin ja tarkoituksiin sopivia. Lopullisen muokausmahdollisuuden tuovat Custom-otsikon alta löytyvät vaihtoehdot, Import ja New Theme. Import antaa mahdollisuuden tuoda zip-paketti, jonka tiedostoja käytetään tyyliissä, kuten esimerkiksi taustakuva ja painikekuvat. New Theme:ssä pääsee luomaan tyylin perinteisesti käsin koodaamalla.

Pages-välilehti sisältää Manage Pages -osion, joka on käytännössä yksinkertainen tila sivuston eri sivujen hallintaa varten. Sivuja ja blogeja voi lisätä ja poistaa. Lisäksi niiden nimiä voi muuttaa, jolloin linkkien nimet muuttuvat navigoinnissa, ja päättää, näkyvätkö ne ohjelman itse luomassa navigaatiossa, sekä suojata sivuja salasanoilla. Salasanasuojaus on jälleen vain maksullisen pro-version ominaisuus.

Settings-välilehti sisältää nimensä mukaisesti asetuksia. Osoitteen voi Weeblyssä määrittellä kolmella tavalla. Oletusosoite on muotoa omaosoite.weebly.com, tämän lisäksi ohjelma auttaa tarvittaessa oman domainin hankkimisessa. Lisäksi on mahdollista käyttää jo valmiina olevaa omaa domain-osoitetta. Muita perusasetuksia ovat Weeblyn automaattisesti luoma sivun yläosan otsikko, salasanat ja Footer Message, joka oletuksena näyttää sivun alaosassa tiedon siitä, millä ohjelmalla sivut on luotu.

Settings-välilehden alta löytyy myös Weeblyn valmismahdollisuudet tehdä hakukoneoptimointia. Aihetta käsitellään SEO-otsikon alla (Search Engine Optimization, standarditermi). Valmiit tekstikentät löytyvät kuvauksen tekemiselle (Description), asiasanoille (Meta Keywords) ja Footer Codes -tiedoille. Perinteisessä merkityksessään termi Footer Code tarkoittaa informaatiota, joka näytetään IE:llä internet-sivun tulostetussa versiossa alatunnisteessa. Weebly käyttää Footer Code -termiä kuvaamaan sitä, että kyseiseen tekstiosioon voi lisätä erilaisten seurantaohjelmien vaatimia koodinpätkiä, esimerkiksi Google Analytics:in käyttöön tarvittavaa koodia. Viimeisenä asetuksissa on Archive/UnPublish-vaihtoehto, joka mahdollistaa luodun sivuston lataamisen itselle .zip-paketina.

Blogitoiminnallisuus on Weeblyssä tehty erittäin helpoksi. Blogin kirjoittamisen voi aloittaa suoraan, ja helpot työkalut löytyvät viestien arkistointiin ja kategorisointiin. Luonnollisesti myös valmis toiminnallisuus löytyy viestien kommentointiin.

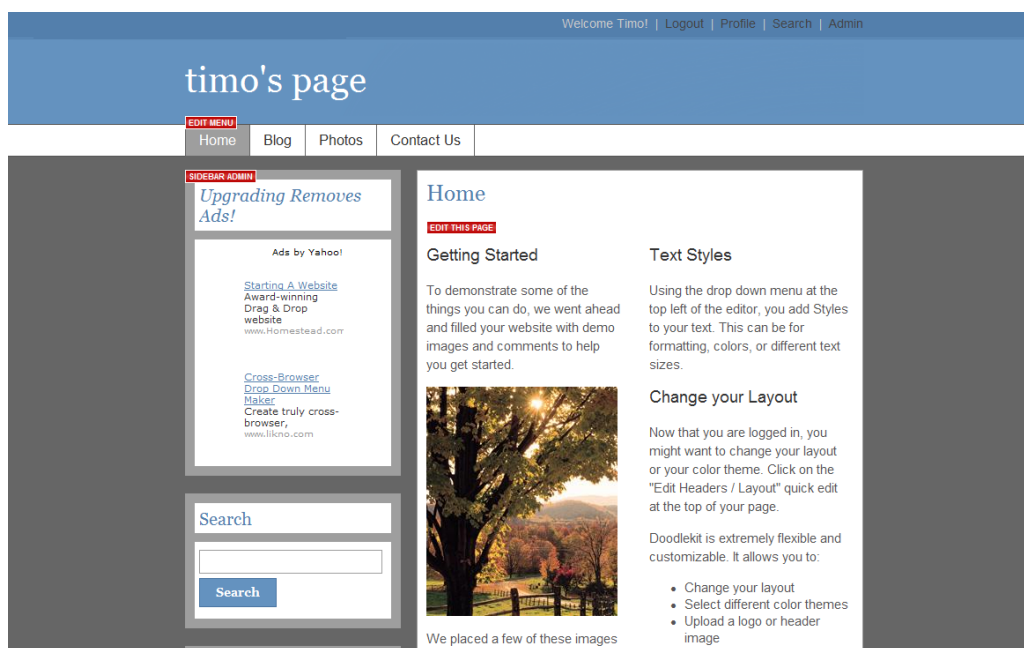
Weebly onnistuu yhdistämään hyvin täysin web-pohjaisen mahdollisimman helppokäyttöisen sivujenluontiohjelman, ja toisaalta myös hienompia ominaisuuksia, jotka mahdollistavat ohjelman käytön vähän useammassa tapauksissa. Täysin virheettömästi ohjelma ei tekemieni testien aikana toiminut, mutta silti niin hyvin, että sen käyttö oli mielekästä ja jopa hauskaa. Joitakin kertoja ohjelma jumittui täysin, ilmeisesti jonkinlaisen serveriongelman takia. Muiden luomia sivuja katsellessa näin myös tapauksia, joissa tyylitiedosto (CSS) ei selvästi toiminut aivan kuten sen olisi ollut tarkoitus, mutta se saattaa tietysti olla myös käyttäjän omaa syytä. Weeblyllä luotuja sivuja selaillessa näkee paljon juuri sen vaatimustason sivuja, joihin ohjelma parhaiten sopii; erilaisten järjestöjen ja pienyritysten kotisivuja, jotka eivät vaadi mitään raskaamman luokan ominaisuuksia, eivätkä ole laajuudeltaan kovin suuria.

### **3.2.2 Doodlekit**

Doodlekit-tilin luominen on hyvin yksinkertaista, WWW-sivulla kysytään vain käyttäjän sähköpostiosoite ja nimi, tulevan sivun nimi sekä salasana tiedot. Mentäessä tästä tilasta eteenpäin lataa Doodlekit automaattisesti esimerkkisivun, jota muokkaamalla oma

sivusto luodaan. Ensisilmäyksellä Doodlekitin käyttäminen ei tunnu aivan yhtä loogiselta kuin Weeblyn, mutta pidemmässä käytössä tulee selväksi, että ohjelmien välillä on loppujen lopuksi aika vähän eroa. Kekseliäs käyttäjä pääsee varmasti molemmilla ohjelmilla ulkoisesti yhtä hienoon lopputulokseen.

Doodlekitin automaattisesti luoma esimerkkisivusto sisältää neljä sivua. Etusivulle on laitettu perustoimintoja, kuten tekstinpätkiä, kuvia ja otsikoita. Valokuvat-sivulla on valmis toiminto kuvagallerian tekemiseen. Blogi-sivulla on hyvin samankaltainen valmis blogisovellus kuin Weeblyssä. Blogin toiminta on tehty juuri niin yksinkertaiseksi kuin se voikin olla, ja hyvä niin. Blogin kirjoittamisessa ja päivittämisessä yksinkertaisuus on valttia, jotta päivityksiä jaksaa tehdä. Contact Us -sivulla on valmis yhteydenottolomake. Kuvassa 2 on Doodlekitin luoma aloitussivusto, ja nähtävillä ovat myös ohjelman automaattisesti luomat 4 sivua.



**Kuva 2: Doodlekitin käyttäjälle automaattisesti luoma esimerkkisivusto**

Kaikilla sivuilla on yhteinen yläpalkki (header) ja alatunniste (footer), jotka ovat luonnollisesti myös muokattavissa. Jokaisen sivun oikeassa laidassa on nähtävissä myös sidebar eli sivupalkki, johon voi lisätä erilaisia sivun käyttöön liittyviä aputoimintoja ja muita lisäominaisuuksia. Esimerkkejä sidebarin mahdollisesta sisällöstä ovat esimerkiksi mainokset, mahdollisuus suorittaa hakuja sivuston sisällöstä, erilaiset muistutukset tai

vaikkapa oman yrityksen yhteystiedot. Sidebar ei suinkaan ole Doodlekitin kehittäjien luoma idea, mutta on todella hyvä asia, että se valmiina toimintona löytyy. Sidebarin avulla on todella helppoa erotella toimintoja tai vaikkapa oheistietoa, joka muuten hukkuisi leipätekstin sekaan.

Sivuston muokkaustilassa oltaessa löytyy ylälaidasta linkki Admin, jonka takaa voi hallita omaa sivustoa. Idea on siinä, että sivulla oltaessa voi sen sisältöä päivittää (kaikki sisältö on eräänlaisissa soluissa), ja Admin-linkin takaa pääsee tekemään kaikki mahdolliset muut muutokset, jotka liittyvät esimerkiksi sivuston rakenteeseen, layoutiin ja asetuksiin. Admin-linkin takaa löytyvät alisivut Settings, Pages/Menus, Sidebar, Add-ons, Users, Statistics ja Account.

Settings-välilehdeltä löytyy paljon perusasetuksia, kuten sivuston nimi, käytettävä kieli, tekijän sähköpostiosoite, ja tieto siitä, mitä sivua käytetään index-tiedostona, muutamia mainitakseni. Settings sisältää myös oikeastaan kaikki puhtaasti sivun ulkoasuun liittyvät valinnat. Layoutin pääsee vaihtamaan monista valmisvaihtoehdoista. Väriteeman vaihto onnistuu erikseen, samoin yläpalkin (header) kuvan, ja sivun otsikon fontin vaihto on mahdollista. Taustakuvan vaihto hoidetaan myös tätä kautta.

Pages/Menus-välilehden kautta hallitaan HTML-tiedostojen määrää (kuva 3). Välilehdelle tultaessa näkyvissä ovat esimerkksisivun luomat 4 yksittäistä sivua. Sivuja voi lisätä tai poistaa, niiden keskinäisiä suhteita voi muuttaa, ja ne voi määrittää näkyviksi tai näkymättömiksi. Lisäksi tätä kautta voi myös lisätä omaan navigointi-menuun linkkejä ulkoisille sivuille.





**Kuva 3: Sivujen määrän hallintaa Doodlekitissä**

Add-ons sisältää nimensä mukaisesti lisäpalveluita, joiden täysimittainen käyttö kuitenkin vaatii maksulliseen ohjelmaversioon siirtymistä. Ilmaiset demoversiot saa käyttöön lomakkeiden tekotyökalusta, forumisovelluksesta ja nettikaupan luontiin tarkoitettusta ostoskorista. Ainoastaan maksullisissa versioissa saa käyttöön työkalut File Uploader, jonka avulla sivuston käyttäjät voivat ladata sivulta tiedostoja, Advertising, joka on tarkoitettu rahaa tuottavan mainoskampanjan luontiin, ja Advanced Security, jolla on mahdollista luoda sivuja, jotka ovat näkyvissä vain tietyille sisäänkirjautuneille käyttäjille.

Users-välilehti on tarkoitettu oman sivuston käyttäjätilien hallintaan. Täysin staattisillakin sivustoilla on ainakin yksi tunnistettu käyttäjä, nimittäin ylläpitäjä. Käyttäjien lisääminen tai poistaminen Users-välilehdellä on todella yksinkertaista. Jokaisesta käyttäjästä ylläpidetään seuraavia tietoja: e-mail-osoite, salasana, nimi ja käyttäjän rooli tietosuojan kannalta. Vaihtoehtoja rooleiksi ovat ylläpitäjä, etuoikeutettu käyttäjä ja peruskäyttäjä.

Statistics-välilehdellä ovat statistiikkaan eli sivuston kävijöiden seurantaan liittyvät valinnat. Doodlekitissä kaikki statistiikatiedot tosin ovat ainoastaan maksullisten versioiden ominaisuuksia. Maksullisten versioiden käyttäjille on tarjolla kaikkia perinteisiä

statistiikkatietoja; eri sivujen kävijämäärät, käytetyt selaimet, ip-osoitteiden seuranta, millä hakusanoilla sivuille on löydetty ja jopa reaaliajassa päivittyvä kävijäseuranta.

Account-välilehdellä ovat nähtävissä kaikki Doodlekitin tarjoamat palvelupaketit, joista siis vain yksi on ilmainen. Suurempi kuukausimaksu tuo mukanaan aina enemmän ominaisuuksia, tilaa tai toiminnan vapautta. Kalliimpia palvelupaketteja tarkastellessa huomaa, että Doodlekitin ylläpitäjät ovat suunnitelleet järjestelmän toimimaan myös niin, että suurempaankin käyttötarpeeseen valmistuvat sivut voidaan sillä tehdä. Hyvänä esimerkkinä käy se, että kalleimmissa palvelupaketeissa on mahdollista luoda omalle sivulle rajoittamaton määrä käyttäjiä, ja palvelimet ovat varautuneet jopa miljoonaan kävijään kuukaudessa.

Doodlekit tuntui käytössä mukavalta ja toimintavarmalta. Graafisen ulkoasun suunnittelijat eivät ole saaneet Doodlekitiin samanlaista ilmavuutta, kuin mitä Weeblyssä on, mutta lopputuloksen kannaltahan sillä ei ole merkitystä.

### 3.3 Kevyet sisällönhallintaohjelmat

Siirryttäessä ultrakevyistä ohjelmista kevyiden puolelle alkavat erot näkyä heti käyttöä aloittaessa. Oikeastaan suurimmat erot peruskäytössä löytyvätkin nimenomaan asennuksesta ja laitteistovaatimuksista, joista kummastakaan ei käyttäjän tarvitse ultrakevyissä ohjelmissa välittää. Tässä työssä esiteltävät kevyet ohjelmat ovat kaiken lisäksi open source -ohjelmia. Open source tuo mukanaan näissä tapauksissa laajan käyttäjä- ja kehittäjäverkoston, joilta kysyä apua tarvittaessa. Toisaalta jatkuvassa kehityksessä olevat ohjelmat vaativat aitoa kiinnostusta ja eräänlaista talkoohenkeä, joita ilman niiden käyttöä tuskin kannattaa vaivautua harjoittelemaan.

CMS-vaihtoehtojen tarjonnan ollessa niin suurta, jää vertailun ulkopuolelle paljon ohjelmia, jotka ansaitsisivat olla mukana. Yksi tällainen ohjelma on Drupal, joka olisi todella ansainnut päästä vertailuun Joomla!:n ja ModX:n kanssa.

### 3.3.1 Joomla!

Joomla! on toinen arviointiin valitsemistani kahdesta kevyestä open source -ohjelmasta. Joomla! on suuren kehittäjäverkostonsa ja lähes loputtomalta tuntuvien laajennusvaihtoehtojensa johdosta todellinen oma maailmansa. Tämän työn tarkoitus ei ole alkaa tutkia Joomla!:n mahdollisuuksia pidemmälle, vaan keskittyä käyttöönoton vaatimuksiin, perustoiminnallisuuteen ja hieman raottaa ohjelman tuomia sivuston kehitysmahdollisuuksia. Selvitin, millaisella panostuksella asiaan täysin syventymätön henkilö pääsee sisään Joomla!:n maailmaan ja aloittamaan sen käytön. Käytön aloittaminen ei luonnollisesti ole yhtä vaivatonta kuin ultrakevyissä ohjelmissa, mutta jos siihen on resursseja, se kannattaa.

Joomla!:n käyttöönottoa ja käyttöä on tämän työn puitteissa testattu käyttämällä apuna XAMPP-ohjelmistoa, jotta tarvittava palvelinympäristö saatiin pystytettyä kotikoneelle. XAMPP ei kuitenkaan ole monilta ominaisuuksiltaan, kuten esimerkiksi tietoturvan kannalta, sellainen ympäristö, että sitä voisi käyttää muussa kuin testausmielessä.

#### Laitteistovaatimukset ja käyttöönotto

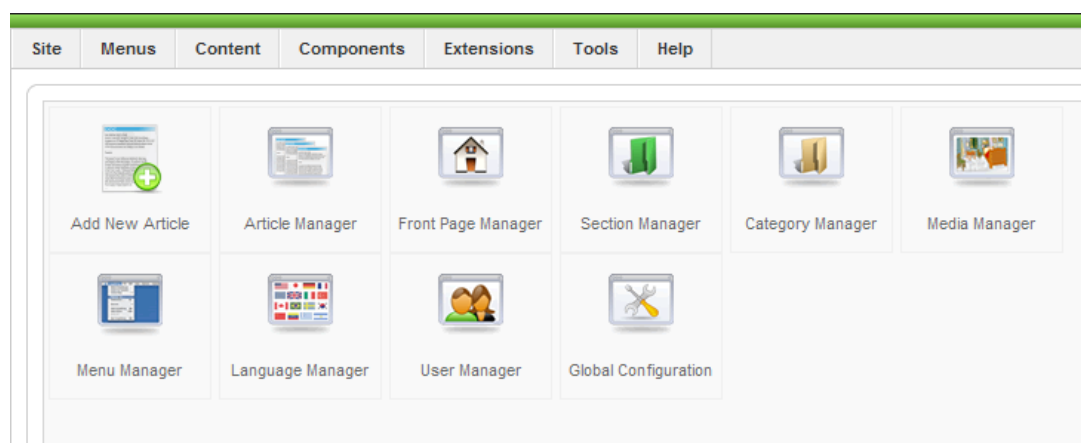
Joomla!:lla on tiettyjä vaatimuksia liittyen siihen palvelimeen tai ympäristöön, jolle käyttäjä haluaa ohjelmiston asentaa. Hyvin ymmärrettävästi jo esimerkiksi palvelimen kanssa toimiminen itsessään on asia, joka ei välttämättä ole tuttu niille käyttäjille, jotka valitsevat ennemmin jonkin ultrakevyen CMS-vaihtoehdon.

Joomla! voidaan asentaa kahdella tavalla; palvelinympäristöön tai paikalliselle tietokoneelle. Tällä hetkellä yleisimmän Joomla!-version tekniset vaatimukset ovat seuraavat: PHP, MySQL ja palvelinohjelmana Apache tai Microsoft IIS (Apachea kuitenkin suositellaan). Jos Joomla! halutaan asentaa yhdelle paikalliselle tietokoneelle, voi avuksi ottaa paketin, joka luo koneelle ympäristön, jossa ovat kaikki edellä mainitut vaadittavat palvelut. Tällaisen paketin käyttö nopeuttaa asennusta; niitä ovat jo aiemmin mainittu XAMPP, ja oikeaan käyttöön tarkoitetut WAMP (Windows-versio), LAMP (Linux-versio) sekä MAMP (Macintosh-versio) (*Joomla! 1.5: Quick Start Guide: 2009*).

Jos edellisissä kappaleissa mainitut asiat eivät kuitenkaan pelota käyttäjää pois, voidaan hyvällä syyllä sanoa, että Joomla!-n käyttöönotosta on tehty erittäin helppoa. Kun paikka, johon Joomla! halutaan asentaa, on tiedossa ja palvelut kunnossa, voi viimeisimmän version ladata pakettina ohjelman kotisivuilta (<http://www.joomla.org>). Kun paketti on purettu oikeaan paikkaan, pääsee konfiguraation tekemään WWW-pohjaisen käyttöliittymän avulla. Kaiken aikaa apuna voi pitää Joomla!-n kehittäjien tekemää opasta, jonka avulla on vaikea tehdä asioita väärin. Kun asennus on valmis, käyttäjä pääsee ihailemaan Joomla!-n käyttöliittymää ja aloittamaan sivujen luonnin.

### Perustoiminnot

Perustoimintoja on Joomla!-ssa melko paljon enemmän kuin aiemmin käsitellyissä ultrakevyissä sovelluksissa, mutta niihin tutustunut huomaa heti yhteneväisyyksiä. Kuvassa 4 on nähtävissä kehittäjän etusivu, jota kutsutaan myös nimellä Control Panel. Siinä on nähtävillä perustoimintojen luokat.



**Kuva 4: Joomla!-n perustoiminnot ylläpitäjän etusivun käyttöliittymässä**

Joomla!-n käytön opiskelussa huomaa kyllä eroja ultrakevyisiin CMS-vaihtoehtoihin. Ohjelman toimintalogiikka on ymmärrettävissä melko helposti, mutta toimivan sivukonaisuuden luominen kestää varmasti pitkään, ellei vastaavista järjestelmistä ole paljon pohjatietoa.

Joomla!:n toiminta perustuu sille, että sivujen asiasisältö kootaan omiksi palasikseen, joita sitten hallinnoidaan erilaisia työkaluja käyttäen, ja asiasisällöstä kootaan kokonaisuuksia. Asiasisällön lisäksi on hallittava tietenkin paljon muitakin asioita, kuten esimerkiksi valikkoja (Menus), sivun layoutia, ja monimutkaisempiin sivustoihin siirryttäessä vaikkapa käyttäjien oikeuksia (*Joomla! Core Features: 2009*).

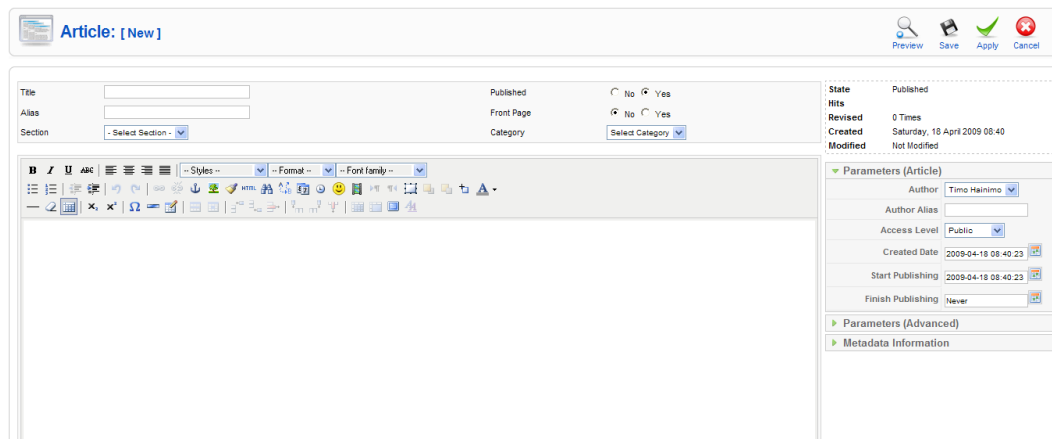
Perustan sivujen sisällölle luovat artikkelit (Articles), jotka voivat sisältää esimerkiksi tekstiä ja kuvia. Artikkelit sisältää todellakin pelkän asiasisällön, jota voidaan näyttää erikseen määritellyllä ulkoisella olemuksella ja sijainnilla. Sivulla voi tietysti olla paljon muunkinlaista sisältöä, kuten vaikka hakukenttiä, uutissyötteitä tai kyselylaatikoita.

Kaksi käytetyintä työkalua asiasisällön koostamiseksi sivulla näkyviksi kokonaisuudeksi ovat Section Manager ja Category Manager. Peruseriaatteena on, että yksi jaosto (Section) sisältää yhden tai useamman kategorian (Category), joihin taas on lajiteltu artikkelit (*Joomla! Documentation for administrators: 2009*).

Kun sisältö on hyvin kategorisoitu, on sitä helppoa hallita, vaikka sitä määrällisesti olisi enemmänkin. Kun artikkelit ovat listattuina eri managereissa, muutellaan vain niiden näkyvyysparametreja sisällön tuomiseksi eri sivuille. Layout on siis kokonaan sisällöstä erillinen oma kokonaisuutensa, jota ei sisältöä suunniteltaessa tarvitse huomioida.

### **Joomla!:n käytön etuja ja kehittymismahdollisuuksia**

Joomla!:n perustoiminnallisuudessakin riittää siis opiskeltavaa, mutta rajattomat kehittymismahdollisuudet tuovat ohjelmaan asennettavat plug-init ja moduulit, joita kehittäjät vapaasti suunnittelevat ympäri maailmaa. Esimerkki moduulista on vaikkapa juuri hakukenttä; se on aputyökalu, joka tuo lisää toiminnallisuutta sivustolle. Plug-init taas ovat enimmäkseen kehitystyön avuksi luotuja apusovelluksia. Yksinkertainen esimerkki plug-ineista on Joomla!:n sisäinen tekstieditori, tai vaikkapa Code Highlighter eli GeSHi, jonka avulla ohjelmoinnin tekijät saavat koodistaan helpommin luettavaa. Kuvassa 5 on nähtävissä Joomla!:n oletustekstieditori.



**Kuva 5: Joomla!-n oletustekstieditori**

### 3.3.2 MODx

MODx on jälkimmäinen kahdesta valitusta kevyestä open source -vaihtoehdosta.

MODx ei ole vielä aivan yhtä pitkälle kehittynyt kuin Joomla!, jolla on suuremman kehittäjäkuntansa ansiosta enemmän kehitettyjä apuohjelmia ja muita lisäominaisuuksia. Sen sijaan MODx antaa kehittyneet ja hyvin toimivat perusraamit, joiden avulla voi alkaa kehittää omaa sisältöään. Tehokkaan käytön opettelu vaatii MODx:n kanssa vielä enemmän aikaa kuin Joomla!-n, mutta toisaalta mielestäni MODx:n kehittäjät ovat saaneet ohjelman monelta osin todella loogiseksi, ja ainakin itselleni sen omaksuminen on tuntunut hyvin mielekkäältä.

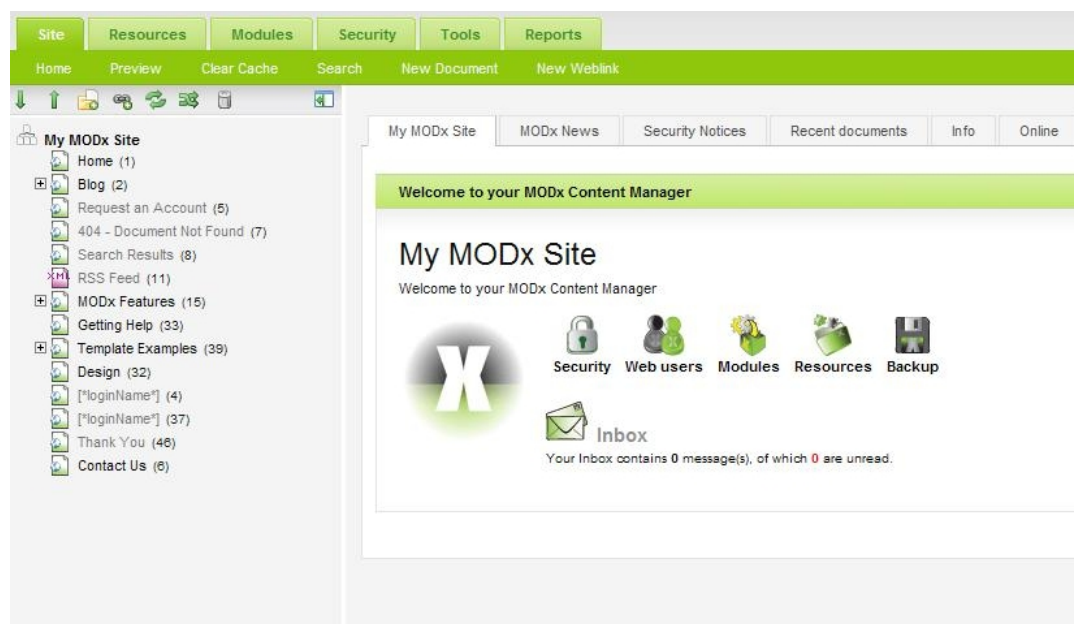
#### Laitteistovaatimukset ja käyttöönotto

Laitteiston vaatimukset ovat MODx:ssä hyvin samantapaiset kuin Joomla!-ssakin. Käyttöjärjestelmien osalta ohjelma toimii Windows-, Mac OS X- ja Linux-ympäristöissä. Palvelinohjelmistoista tuettuja ovat jälleen Apache ja IIS, sekä myös Lighttpd ja Zeus. Lisäksi vaatimuksista löytyy vielä PHP ja MySQL-tietokanta. Käyttöönoton jälkeen varsinainen sivuston hallinta voi tehdä käyttämällä tuettuja WWW-selaimia Internet Explorer, Firefox tai Safari (*MODx Documentation: Installation: 2009*).

Asennus on hyvin samantapainen kuin Joomla!-ssakin, eli ohjelman tiedostot siirretään palvelimelle WWW-kansion juureen. Jos kaikki palvelut ovat kunnossa, päästään avaamaan index.php-tiedosto, jonka jälkeen asetusten laittaminen kuntoon hoituu helposti askel-askeleelta-ohjeistetussa sovelluksessa. Uuden käyttäjän kannalta on sanottava, että WWW-sivuilta löytyvät ohjeet eivät ole yhtä kattavia kuin Joomla!-ssa.

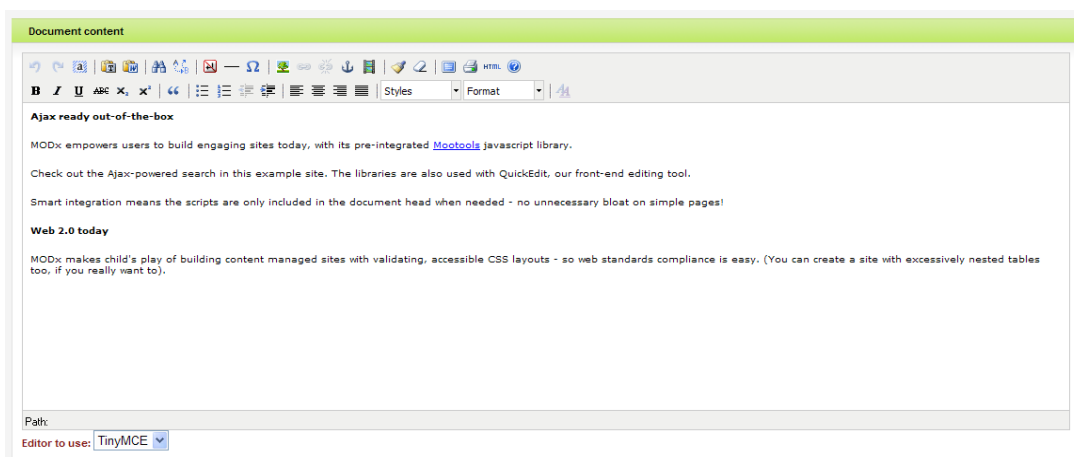
## Perustoiminnot

MODx:n käyttöliittymä (kuva 6), jossa sivuja muokataan, on looginen. Ylimpänä on kaksi riviä valikkoja, joista voi vaihtaa sivun tilaa; esimerkiksi perustilassa näkyy eri sivujen perustiedot ja sisältöä päästään muokkaamaan. Jos ylävalikosta sen sijaan on valittu esimerkiksi Security > Manage Users, näkyy samaa sivua tiedostopuussa klikkaamalla kyseisen sivun käyttäjätiedot, vaikkapa tieto siitä, kuka saa muokata sivun sisältöä. Ylävalikosta löytyy paljon muutakin perustoiminnallisuutta, kuten esimerkiksi uuden dokumentin lisääminen.



**Kuva 6:** MODx:n käyttöliittymä

Ylävalikoiden alla ja vasemmalla puolella on tiedostopuu, jossa on nähtävillä kaikki oman sivuston dokumentit. Päätoimintana tiedostopuusta tietysti valitaan dokumentit, joita halutaan muokata, siirtää tai poistaa. Oikealla on ikkuna, jossa varsinainen tietosisältö esitetään, olipa se sitten vaikka sivun tietoturvatietoja, itse muokattava dokumentti tai esimerkiksi jokin moduuli. Kuvassa 7 on nähtävissä oletustekstieditori, jossa muokattava dokumentti voidaan esimerkiksi avata. MODx:ssä on moduuleita, plug-ineja sekä muitakin rakennuspalikoita, joita voi itse kehittää ja tuoda niiden avulla lisätoiminnallisuutta. Esimerkki plug-inista on vaikkapa uudelleenohjaus, joka ohjaa sivujen käyttäjän haluttuun paikkaan sen dokumentin puuttuessa, johon hän yritti päästä. Moduuli voi olla mikä tahansa PHP:llä luotu apuohjelma, joka toimii sivun yhteydessä (*MODx Documentation: Content Editing: 2009*).



**Kuva 7: Tekstin muokkausta MODx:n oletustekstieditorilla**

## **MODx:n käytön etuja ja kehittymismahdollisuuksia**

MODx:n kehittäjät lupaavat ohjelmiston antavan erityisen hyvät SEO- eli hakukoneoptimointimahdollisuudet, ja niin asia näyttää olevankin. Kehittymismahdollisuudet ovat MODx:ssä varmasti aivan yhtä hyvät kuin Joomla!:ssakin, mutta kuten sanottu, MODx saattaa monelta osin vaatia vielä suurempaa osaamista, kun valmiita apupalikoita ei löydykään kaikkien asioiden tekemiseen. Taitojen riittäessä MODx antaa ehkä Joomla!:a enemmänkin toiminnan ja kehittämisen vapautta.



## 4 Muita tapoja helpottaa sivuston päivitettävyyttä

Tapoja helpottaa sivuston päivitettävyyttä on tietenkin olemassa lukemattoman paljon, mutta tässä luvussa esitellään niistä muutamia, tämän työn rajaukseen sopivia. Nämäkin mahdollisuudet lukeutuvat niihin, joista koituu käyttäjälle vähän tai ei ollenkaan kuluja.

Alkuperäisenä tarkoitukseni oli käsitellä tässä luvussa vielä Microsoft Frontpagen olemassa olevia lisäosia, jotka helpottavat sillä tehtyjen WWW-sisältöjen siirtämistä palvelimelle ja Internetiin. Frontpage on kuitenkin väistynyt ohjelma, jonka kehitys on jo lopetettu. Sen tilalla on Microsoft SharePoint, joka on suunniteltu suurempienkin sisältöjen hallintaa varten. SharePoint on todella suuri oma kokonaisuutensa, josta pitäisi kirjoittaa kokonainen opinnäytetyö, ja niitä on jo olemassakin. SharePoint ei myöskään sovellu esiteltäväksi tässä työssä, koska se kuuluu ehdottomasti raskaampien sisällönhallintaohjelmien joukkoon.

### 4.1 FTP-synkronointi

FTP-synkronointi on nimensä mukaisesti sitä, että ohjelma päivittää halutun sisällön automaattisesti, esimerkiksi käyttäjän tietokoneelta palvelimelle. Päivitys voidaan tehdä monien muidenkin tietovarastojen välillä, joita voivat olla vaikkapa kannettavat, ulkoiset tallennusasemat ja tietokannat.

Synkronointi voidaan yleensä hoitaa painikkeen painalluksella tai ennalta suunniteltua aikataulutusta käyttäen. Nämä ominaisuudet voivat tuoda korvaamatonta apua ylläpitäjälle, jolla on hallittavanaan suuria sisältömääriä, esimerkiksi WWW-pohjaisia valokuvagallerioita tai muuta vastaavaa.

Kaksi esimerkkiä sisältöjen väliseen synkronointiin erikoistuneista ohjelmista ovat FTP Synchronizer (<http://www.ftpsynchronizer.com>) ja GoodSync (<http://www.goodsync.com>), joista jälkimmäisestä on ladattavissa myös ilmainen, käyttömahdollisuuksiltaan rajatumpi versio (*GoodSync Tutorials: 2009*).

## 4.2 Yrityskohtaisesti suunniteltavat, CMS-elementtejä sisältävät ratkaisut

Yrityskohtaisista CMS-ratkaisuista tulevat ensimmäisenä mieleen raskaat järjestelmät ja niissä suoritettava implementointi sekä mahdolliset muut muutokset, joiden tavoitteena on sopeuttaa järjestelmä yrityksen omakohtaisiin tarpeisiin. Myös pienemmissä järjestelmissä tehdään kuitenkin yrityskohtaisia ratkaisuja. Monet osaavat WWW-suunnittelijat saattavat luoda itse WWW-sivuille toiminnallisuutta, joka korvaa ainakin osan CMS-järjestelmän mahdollisuuksista. Useimmiten PHP:ta ja muita WWW-käyttöön suunniteltuja ohjelmointikieliä kirjoitetaan itse käyttäjätunnistukseen tai vastaaviin, vielä melko helposti toteutettaviin palveluihin liittyen.

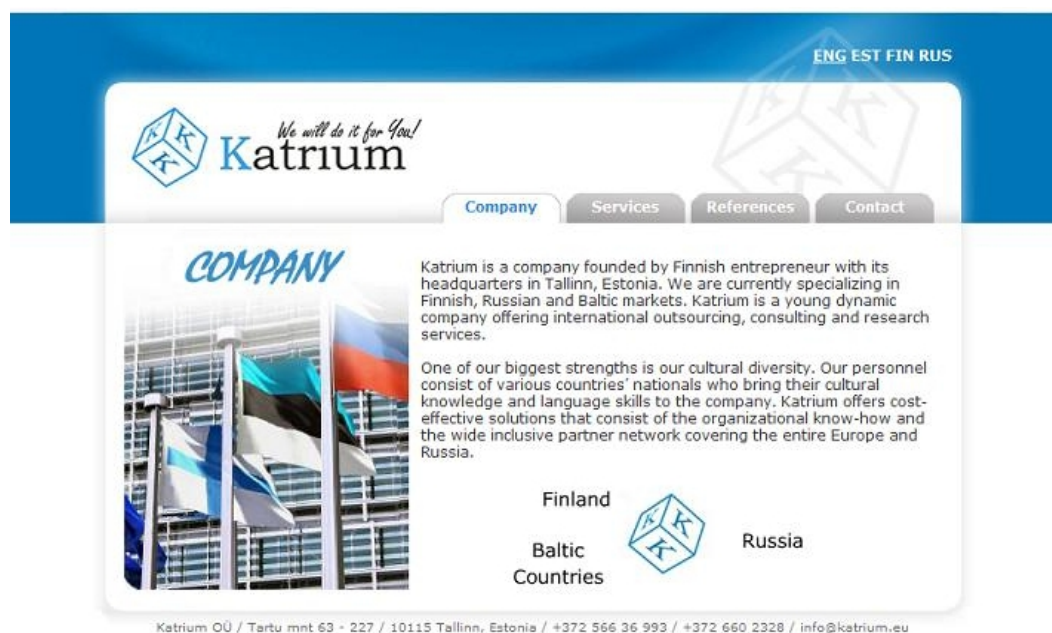
## 5 Oma sivustoprojekti

Syyskuussa 2008 aloitin WWW-sivujen uusimisen Tallinnassa, Virossa päämajaansa pitävälle Katrium OÜ:lle (<http://www.katrium.eu>). Tärkeimpänä tavoitteena projektilla oli saada sivuille nykyaikaisempi ilme, vanhojen sivujen jäätyä melko pahasti ajan hampaisiin. Myös CMS-ohjelman käyttöönotosta oli alussa puhetta, mutta se jäi ainakin vielä tällä erää tekemättä.

Projektin alussa käytimme liiankin paljon aikaa uuden ulkoasun peruselementtien suunnitteluun. Tämä söi aikaa projektilta, jonka oli määrä valmistua uuteenvuoteen mennessä. Ulkoasun suunnittelu oli omalta osaltaan toisaalta todella hauskaa; käytännössä toimimme alussa niin, että loin täysin staattisia sivujen layout-ehdotuksia ja siirsin ne palvelimelle, jonka jälkeen yrityksen henkilökunta esitti ulkoasuun kehittämissuunnitelmia. Ulkoasun suunnittelu on myös se osa-alue WWW-sivujen suunnittelussa, joka on ”ominta” aluetta. Jo alusta lähtien oli selvillä perusasiat, joille ulkoasun tulisi perustua. Värimaailman osalta vaikutteita haettiin luonnollisesti paljon yrityksen sinisestä logosta, johon liitettiin luottamusta herättäviä värejä, kuten harmaa ja valkoinen. Muuten ulkoasun tuli olla mahdollisimman hillitty ja arvokas, menettämättä silti kiinnostavuutta. Henkilökuntaa riittävästi miellyttävän ulkoasun löytämisen pitkittyessä saatiin apua aivan uudelta taholta; ulkoasun suunnitteluun osallistui yrityksen työharjoittelija. Hänen oli helpompi hakea kaikkia päättäviä tahoja miellyttävää linjaa työskennellessään itse paikalla Tallinnassa. Tämä tapaus opetti minulle lisää siitä paljon puhutusta asiasta, kuinka vaikeaa ohjelmistoprojekteissa on yhdistää tilaajan toiveet ja ohjelmiohjelmaajan käsitys siitä, mitä halutaan luotavan. Kun pelkkä WWW-sivujen ulkoasu voi tuottaa näin suuren työmäärän suunnittelussa, voi vain kuvitella, millaisia samat ongelmat ovat laajemmissa, ”oikeissa” ohjelmistojen suunniteluissa.

Lopulta hyväksyttävä ulkoasun linja löydettiin. Sisällöllisesti sivustosta tuli ihan perinteinen staattinen esittelysivu. Etusivulla kerrotaan yrityksen yleisiä tietoja, palvelut-välilehdellä syvennytään yrityksen tarjoamien palvelujen kirjoon, referenssit sisältää tiedot yrityksen tunnetuimmista yhteistyökumppaneista, ja viimeisenä löytyy vielä yhteystiedot-osio, jonka kautta saa yhteyden kaikkiin yrityksen avainhenkilöihin. Sama sisältö löytyy englanniksi, eestiksi, suomeksi sekä venäjäksi. Sisältöä elävöitettiin jokaiselta si

vulta löytyvällä kuvalla, ja etusivun Flash-mainoksella. Flash-mainos on mielestäni juuri sopivan yksinkertainen ja hillitty; se ei varasta liikaa huomiota muulta ulkoasulta. Mainoksessa kerrataan yrityksen tärkeimmät toimialueet. Kuvassa 8 on nähtävissä sivuston lopullinen, uusittu graafinen ilme. Kuva on englanninkielisen version etusivulta.



**Kuva 8: Katrium OÜ:n uusittu graafinen ilme**

Ulkoasuvaatimusten tarkennuttua erittäin tarkoiksi ja ajan käydessä vähiin tajusin, ettei aikataulu venyisi sellaisen CMS-vaihtoehdon löytämiseen, jolla ehtisin opetella ulkoasun muokkaamisen juuri halutun kaltaiseksi. Tämä on toki mahdollista ultrakevyillä CMS-vaihtoehdoilla, jotka myös usein näyttävät tukevan oman XHTML-koodin upottamista ohjelman luomiin sivuihin. Tiukan aikataulun lisäksi syynä perinteisten sivujen luontiin oli se, että sivuille ei ainakaan lähitulevaisuudessa ole tulossa enempää interaktiivisuutta vaativaa materiaalia. Pelkästään asiasisältöön mahdollisesti tehtävät pienet muutokset on helppoa tehdä suoraan koodiin, koska muokattavissa oleva tekstiosa on XHTML-tiedostossa merkitty erityisillä ”<!--TEKSTIOSIO ALKAA-->” ja ”<!--TEKSTIOSIO LOPPUU-->” -tageilla. Varsin yksinkertainen ratkaisu, jonka jälkeen tarvittiin vain asiasisältöä päivittävälle henkilölle ohjeistus FTP-yhteyden käyttöä varten, jotta varsinainen palvelimelle päivittäminen onnistuisi. Ohjeistus löytyy tämän opinnäytteen liitteistä (Liite 2).

Sivujen käyttötarkoitusten mahdollisesti laajentuessa tulevaisuudessa voi edessä kuitenkin olla CMS-järjestelmän käyttöönotto. Nykyisellään Katriumin sivut edustavat juuri niitä yrityssivuja, joiden vaatimuksiin myös ultrakevyiden CMS-ohjelmien edellytykset riittävät. Melko pienellä vaivalla voisi kuitenkin käyttää myös Joomla!:n tai ModX:n kaltaista open source -ohjelmistoa, joka tarvittaessa sitten tukisi suurempiakin laajentumistarpeita. Toisaalta sekin on totta, että kovin pitkälle tulevaisuuteen tuskin kannattaa tehdä valmiita suunnitelmia WWW-maailmassa. Kuitenkin 5-10 vuoden päästä kaikki mahdollisuudet ovat ehtineet kehittyä ja muuttua todella paljon. Jos CMS-järjestelmän käyttöönotto todella tulee ajankohtaiseksi, luultavasti ainakin itse alkaisin näillä pohjatiedoilla suunnitella projektia Joomla!:n tai ModX:n pohjalle.

## 6 Yhteenveto ja johtopäätökset

Yksi tämän työn tärkeimpiä tavoitteita oli läpikäydä valittujen ohjelmien käyttöönoton ja perustoimintojen harjoittelun helppoutta. Mielestäni onnistuin luomaan realistisen kuvan valittujen ohjelmien vaikeustasosta. Henkilökohtaisesti tärkeä tavoite oli oman WWW-sisällönhallinnan tietouden yleinen lisääminen, ja sekin toteutui. Lähtökohtaisesti näin laaja aihe ei ehkä ole optimaalinen lopputyölle, mutta CMS-ohjelmien vertailu oli näkökulmana uusi ja uskoakseni hyödyllinen.

Lopputyön aiheeseen liittyen toteutetut Katrium OÜ:n uudet WWW-sivut saivat nekin kiitosta, vaikka CMS-ohjelman käyttöön ei vielä siirrytty. Katriumin tilanteen tarkastelu toi esiin senkin seikan, ettei CMS-ohjelmaan siirtyminen välttämättä ole kannattavaa, ellei sisältö ole sen mukaista, että uudesta järjestelmästä saataisiin merkittävää hyötyä.

Se, millainen CMS-järjestelmä missäkin tilanteessa kannattaa ottaa käyttöön, riippuu todella monista tekijöistä. Ultrakevyet CMS-vaihtoehdot puolustavat paikkaansa etenkin sellaisten pienten yritysten käytössä, joille on hyötyä näkyvyydestä Internetissä, ja joilla on mahdollista WWW-sisältöä melko vähän. Tässä tapauksessa resursseja ei kannata hukata liikaa WWW-sisällönhallinnan miettimiseen. Jos osaamista tai kiinnostusta CMS-ohjelmiin kuitenkin löytyy vähänkin enemmän, tuovat Joomla!:n ja MODx:n kaltaiset ohjelmat kehittäjälle aivan erilaista vapautta.

Katriumin kanssa tehdyt aikataulut pitivät, mutta se söi loppuvaiheessa paljon aikaa opinnäytetyöprosessilta. Kaikki kuitenkin valmistui juuri ja juuri ajallaan. Lähes kaikki käyttämäni lähdemateriaali oli WWW-muodossa. Suurimmaksi osaksi käytin kuitenkin erittäin luotettavia lähteitä, kuten ohjelmien valmistajien itse tuottamia tutoriaaleja ja muita dokumentteja. Myös muut käyttämäni WWW-lähdetiedot suurimmalta osin todistivat paikkansapitävyyden käytännössä ohjelmien testausvaiheissa.

Kokonaisuutena voidaan siis sanoa, että tavoitteet saavutettiin. Hyvä yleiskuva pienten kustannusten CMS-järjestelmistä saavutettiin.

## Lähteet

Boiko, Bob. 2005. Content management bible. Yhdysvallat, Indianapolis.  
Wiley Publishing, Inc.

CMS Learning – Different types of CMS. [Online] [viitattu 12.2.2009].  
<http://cmslearning.blogspot.com/2007/10/different-types-of-cms.html>

GoodSync Tutorials: Basic Setup. [Online] [Viitattu 10.4.2009].  
<http://www.goodsync.com/tutorial.html>

Joomla! Core Features. [Online] [viitattu 21.3.2009].  
<http://help.joomla.org/ghop/feb2008/task020/Joomla%21%20Core%20Features%20V1.2.pdf>

Joomla! Documentation for administrators. [Online] [viitattu 21.3.2009].  
<http://docs.joomla.org/Administrators>

Joomla! 1.5: Quick Start Guide. [Online] [viitattu 20.3.2009].  
[http://help.joomla.org/ghop/feb2008/task048/joomla\\_15\\_quickstart.pdf](http://help.joomla.org/ghop/feb2008/task048/joomla_15_quickstart.pdf)

MODx Documentation: Content Editing. [Online] [Viitattu 6.4.2009].  
<http://svn.modxcms.com/docs/display/MODx096/Content+Editing>

MODx Documentation: Installation. [Online] [Viitattu 3.4.2009].  
<http://svn.modxcms.com/docs/display/MODx096/Installation>

OpenJason: 50 Content Management Systems. [Online] [viitattu 20.12.2008].  
<http://www.openjason.com/2008/02/23/50-content-management-systems/>

PHP: Manual. [Online] [viitattu 22.2.2009].  
<http://fi.php.net/FAQ.php>

## Liitteet

### Liite 1

#### Linkkejä CMS-ohjelmien valmistajien kotisivuille

##### **Ultrakevyitä ohjelmavaihtoehtoja**

Weebly

<http://www.weebly.com/>

Doodlekit

<http://doodlekit.com/home>

Wordpress

<http://wordpress.org/>

ClickCMS

<http://www.clickcms.co.uk/>

LightCMS

<http://www.speaklight.com/>

##### **Kevyitä open source -vaihtoehtoja**

Joomla!

<http://www.joomla.org/>

MODx

<http://modx.com/>

Drupal

<http://drupal.org/>

Moodle -oppimisympäristöihin erikoistunut vaihtoehto

<http://moodle.org/>

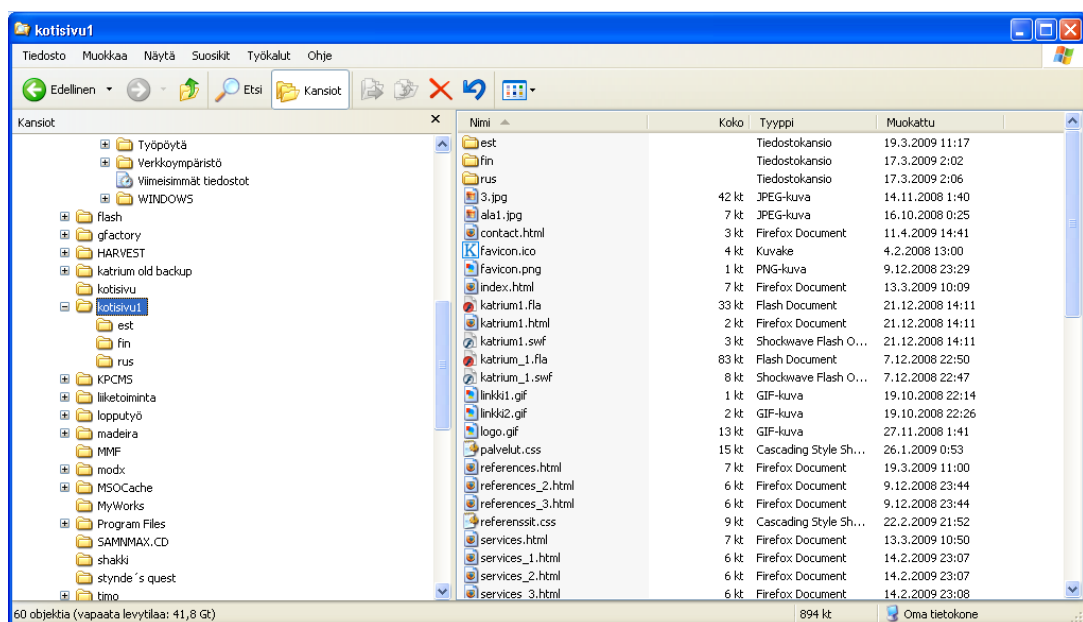


## Liite 2

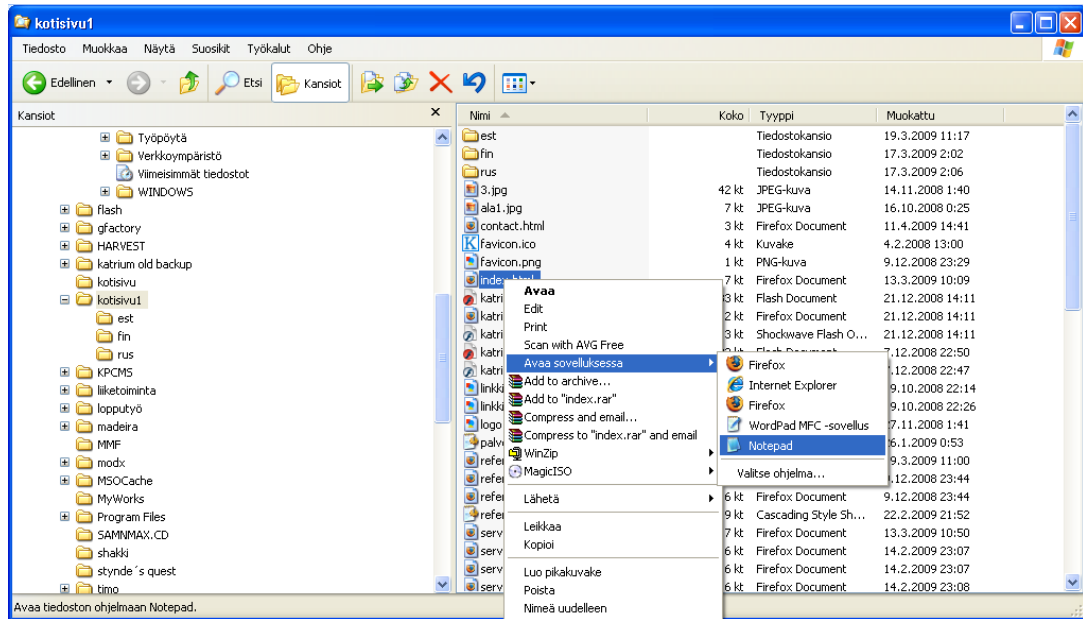
### Ohje WWW-sivujen päivittämiseen – Katrium OÜ

Tämän ohjeistuksen mukana tulee .zip-paketti, joka sisältää www-sivujen materiaalin kokonaisuudessaan. Tee esimerkiksi C-asetalle tai muuhun haluamaasi paikkaan uusi kansio ”Kotisivut”, johon purat paketin sisältämän materiaalin. Kun menet katsomaan kansion sisältöä, huomaat että siellä on paljon tiedostoja sekä alikansiot est, fin ja rus.

Ensimmäisenä näkyvissä ovat siis ENG-version tiedostot ja alikansioissa ovat muiden kieliversioiden tiedostot.



Tiedostot, joiden päätteenä on .html, ovat niitä joissa on sivujen asiasisältö. Oletetaan, että haluaisin muuttaa ENG-version etusivun tekstiä. Etusivu on index.html. Klikkaan sitä hiiren oikealla napilla ja valitsen ”Avaa sovelluksessa > Notepad”. Tässä kannattaa käyttää nimenomaan Notepadia (tunnetaan myös nimellä muistio).



Kun Html-tiedosto on avattu Notepadissa, löytyy muokattavissa oleva teksti ”tekstiosio alkaa” ja ”tekstiosio loppuu” –tagien välistä. Muuta tekstiä sen verran kuin haluat, ja tallenna tiedosto.



Nyt muokattu Html-tiedosto löytyy siis omalta koneeltasi. Seuraavaksi katsotaan, miten se saadaan päivitettyä palvelimelle.

## FTP-ohjelman hankkiminen

Ensimmäiseksi pitää tietysti hankkia ohjelma, jolla muodostaa FTP-yhteys palvelimelle, jotta sinne pääsee tiedostoja siirtämään. FTP-ohjelmia on kuin sieniä sateella, mutta olen päättänyt suositella seuraavaa:

Core FTP LE 2.1 build 1603

[http://download.cnet.com/Core-FTP-LE/3000-2160\\_4-10195752.html?tag=mncol;pop](http://download.cnet.com/Core-FTP-LE/3000-2160_4-10195752.html?tag=mncol;pop)

Core FTP Lite on ilmainen myös yrityskäytössä, ja erittäin suosittu sen toimivuuden ja monipuolisuuden vuoksi. Yllä oleva linkki vie download.com:in sivulle, josta ohjelman voi ladata pakattuna painamalla tekstiä download now.

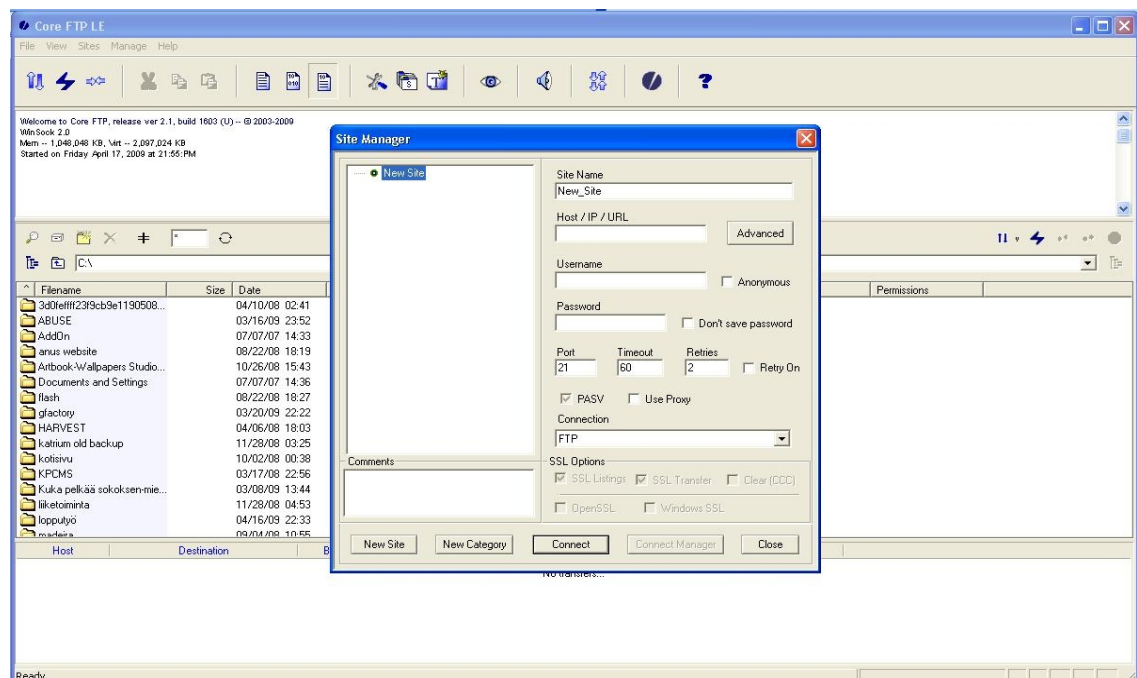
Kun olet ladannut tiedoston haluamaasi paikkaan, avaa se. Installointi käynnistyy, se on erittäin yksinkertainen, ei tarvitse painaa muuta kuin next-nappia siihen asti, että ohjelma tulee asennettua koneelle.

### FTP-ohjelman käyttäminen

#### Yhteyden muodostaminen palvelimelle

Kun ohjelma on asennettu, avaa se (luultavasti se teki pikakuvakkeen työpöydälle, jos et muuttanut perusasetuksia). Käynnistyksen yhteydessä ohjelma kysyy kysymyksen ”Make Core FTP your default client?” Koska en tiedä koneesta, jolla tämä suoritetaan, suosittelen valitsemaan vastauksen ”No”. Seuraavaksi ohjelma kysyy ”Would you like to use ’notepad.exe’ to edit all ascii files?”. Vastaa ”Kyllä”.

Tämän jälkeen ohjelma käynnistyy ja näyttää seuraavalta:



Heti auenneeseen Site Manager -ikkunaan voidaan täyttää yhteyden muodostamiseen tarvittavat tiedot.

Site Name: Vapaavalintainen nimi yhteydelle

Host / IP / URL: WWW-sivun osoite

Username: Käyttäjänimi

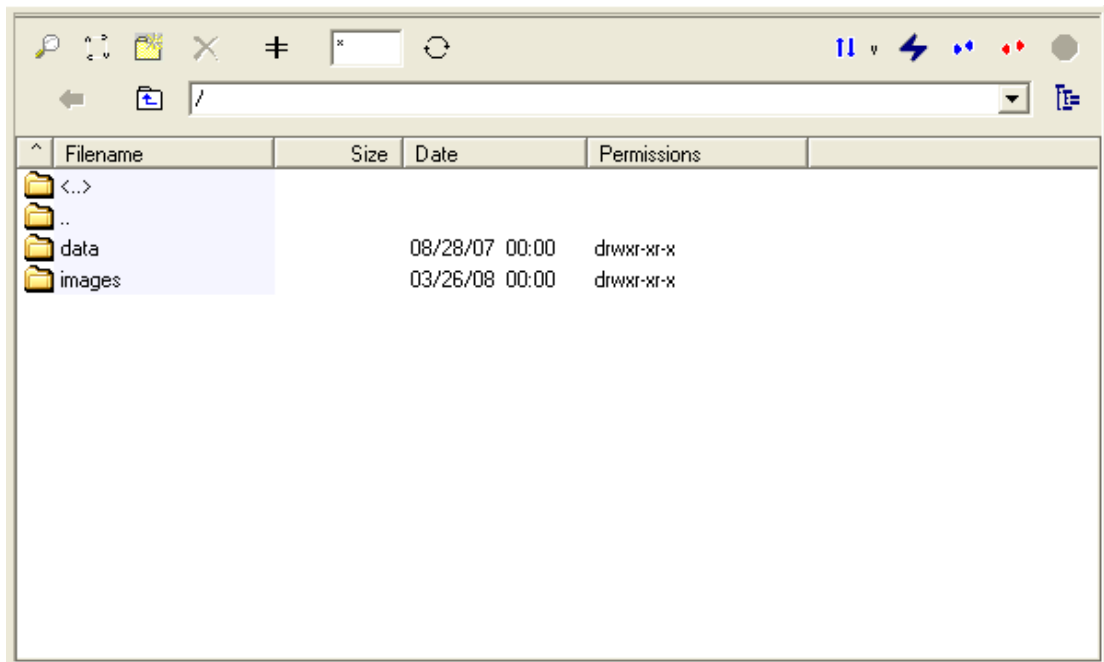
Password: Palveluntarjoajalta saatu salasana

Tarkista vielä, että Connection-vetovalikosta on valittu FTP.

Paina connect, ja yhteys muodostuu.

### Palvelimen oloihin tutustuminen

Ohjelmassa näkyy kaksi tiedostopuuta, joista vasemmanpuoleinen on oma tietokoneesi, ja oikeanpuoleinen on palvelin. Tultaessa palvelimelle, näyttää se tältä:



-tuplaklikkaa kansiota data

-tämän jälkeen tuplaklikkaa kansiota www

-nyt päästään näkemään itse sivuston tiedostot:

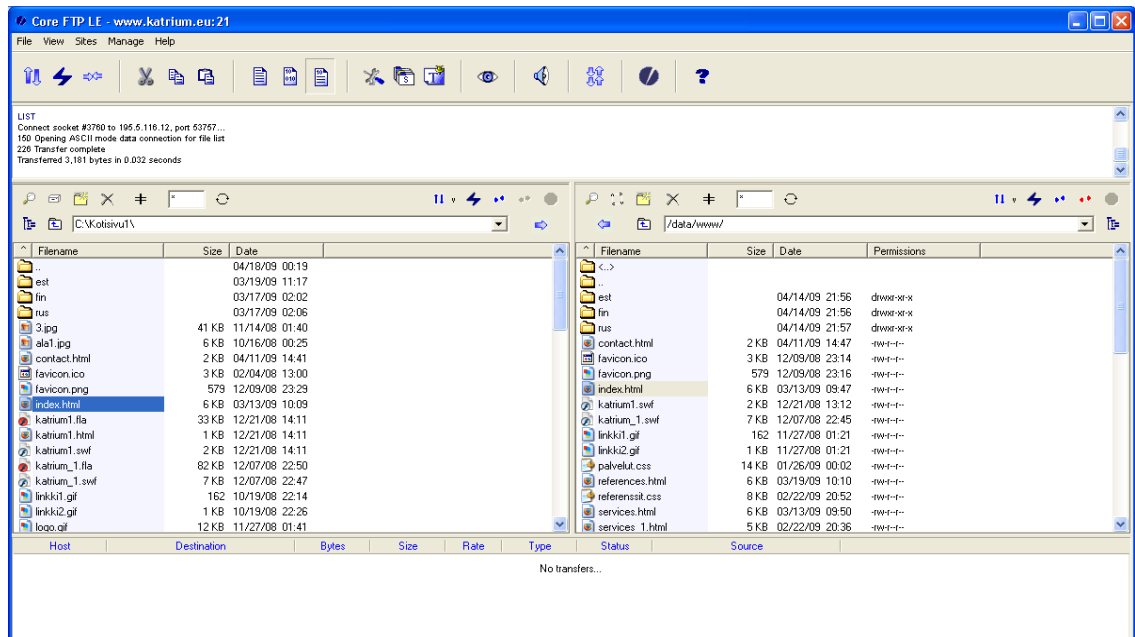
Filename	Size	Date	Permissions
<..>			
..			
est		04/14/09 21:56	drwxr-xr-x
fin		04/14/09 21:56	drwxr-xr-x
rus		04/14/09 21:57	drwxr-xr-x
contact.html	2 KB	04/11/09 14:47	-rw-r--r--
favicon.ico	3 KB	12/09/08 23:14	-rw-r--r--
favicon.png	579	12/09/08 23:16	-rw-r--r--
index.html	6 KB	03/13/09 09:47	-rw-r--r--
katrium1.swf	2 KB	12/21/08 13:12	-rw-r--r--
katrium_1.swf	7 KB	12/07/08 22:45	-rw-r--r--
linkki1.gif	162	11/27/08 01:21	-rw-r--r--
linkki2.gif	1 KB	11/27/08 01:21	-rw-r--r--
palvelut.css	14 KB	01/26/09 00:02	-rw-r--r--
references.html	6 KB	03/19/09 10:10	-rw-r--r--
referenssit.css	8 KB	02/22/09 20:52	-rw-r--r--
services.html	6 KB	03/13/09 09:50	-rw-r--r--
services_1.html	5 KB	02/22/09 20:36	-rw-r--r--

-näkyvissä olevat tiedostot ovat ENG-kieliversioiden tiedostot  
 -kansioista löytyy loppujen kieliversioiden tiedostot: est, fin ja rus

Nyt kun ohjelma on jo vähän tuttu, ei puutu enää muuta kuin alussa päivittämämme tiedoston siirto vanhan tilalle palvelimelle. Hae vasemmanpuoleiseen ikkunaan omalla koneellasi oleva kansio, josta löytyy päivitetty versio tiedostosta.

-Poista vastaava tiedosto palvelimelta (oikea puoli). Hiiren oikella napilla > Delete > varmista poistaminen.

-Siirrä vasemmalta puolelta uusi versio korvaamaan nyt kadonnutta tiedostoa. Voit siirtää tiedoston raahaamalla sen oikealle puolelle, tai käyttämällä vasemman puolen oikeassa reunassa olevaa nuolta (se joka näyttää oikealle).



Nyt päivitys on tehty, ja päivitetty versio tulee näkyviin internetiin.