
**YHTEYSTIETOMODUULIN SUUNNITTELU JA
TOTEUTUS MODX-JÄRJESTELMÄSSÄ**



Ammattikorkeakoulun opinnäytetyö

Tietojenkäsittelyn koulutusohjelma

Visamäki, 24.11.2011

András Lehtinen



Tietojenkäsittelyn koulutusohjelma
Hämeenlinna

Työn nimi Yhteystietomodulin suunnittelu ja toteutus MODX-järjestelmässä

Tekijä András Lehtinen

Ohjaava opettaja Lasse Seppänen

Hyväksytty _____ . _____ .20 _____

Hyväksyjä

Visamäki
Tietojenkäsittelyn koulutusohjelma

Tekijä	András Lehtinen	Vuosi 2011
Työn nimi	Yhteystietomoduulin suunnittelu ja toteutus MODX-järjestelmässä	

TIIVISTELMÄ

Opinnäytetyön tavoitteena oli suunnitella ja toteuttaa yhteystietomoduuli MODX-julkaisujärjestelmään. Työ tehtiin Mainostoimisto Parooninmäen Moukarille opintoihin kuuluneen työharjoittelun aikana. Mainostoimisto Parooninmäen Moukari käyttää hyväkseen Internet-sivujen toteutuksessa avoimen lähdekoodin järjestelmää nimeltä MODX. Tarkoituksena oli tehdä yhteystietomoduulista yksi toimiva kokonaisuus, jota he voivat hyödyntää tulevaisuuden asiakasprojekteissaan.

Opinnäytetyön tarkoituksena oli myös tutustua MODX-julkaisujärjestelmään. Julkaisujärjestelmä eroaa yleisimmistä sisällönhallintajärjestelmistä, koska se on helposti muokattavissa sekä sillä on selkeä ja käyttäjäystävällinen hallintaympäristö. MODX-järjestelmän käyttämiseen tarvitaan XHTML-merkintäkielen sekä CSS-muotoilukielten osaamista. PHP-ohjelmointikielen ja MySQL-tietokantojen ymmärrys lisää järjestelmän käytettävyyttä ja antaa käyttäjälle enemmän mahdollisuuksia käyttää järjestelmää sen tarjoamille mahdollisuuksille.

Yhteystietomoduulin teko onnistui annetussa aikataulussa ja kaikki vaadittavat ominaisuudet toimivat. Moduulia on mahdollista jatkokehittää ja siihen voi lisätä uusia ominaisuuksia käyttäjän ja asiakkaan toiveiden mukaisesti. Yhteystietomoduulia voi mukauttaa helposti jokaisen uuden projektin vaadittaviin toiminnallisuuksiin tai lisätä vanhalle sivustolle. MODX-järjestelmä sopii hyvin PHP-ohjelmoinnista kiinnostuneille Internet-sivujen tekijöille. Se vaatii hieman tutustumista MODX-järjestelmään, mutta järjestelmän rakenne avautuu käyttäjälle käytön yhteydessä. Vastaalkajille ja yksinkertaisten Internet-sivujen tekijöille sopivat paremmin sisällönhallintajärjestelmät kuten Joomla! tai WordPress. Niihin on helpompi tutustua ja niistä löytyy enemmän valmiiksi tehtyjä komponentteja, joita käyttäjä voi hyödyntää omilla sivuillaan.

Avainsanat MODX, sisällönhallintajärjestelmä, CMS, moduuli

Sivut 28 s. + liitteet 34 s.

Visamäki
Degree Programme in Business Information Technology

Author	András Lehtinen	Year 2011
Subject of Bachelor's thesis	Planning and executing a contact information module for MODX software	

ABSTRACT

The purpose of the thesis was to plan and execute a contact information module for MODX software. The thesis was commissioned by Mainostoimisto Parooninmäen Moukari Oy. The client handles all their web pages through MODX software. MODX is an open source content management system (CMS). The aim was to build a new module for contact information which Parooninmäen Moukari can utilize in their future projects.

The purpose of the thesis was also to point out all the diverse properties and qualities MODX software has compared to other content management systems. MODX is a user-friendly and easy to learn software with numerous editing possibilities. MODX uses XHTML markup and CSS style sheet languages for creating web pages, as well as MODX software's own tags. It is recommended to have the basic knowledge of PHP programming language and MySQL database knowhow before starting out with MODX.

The development project was carried out in the given schedule and the client was satisfied with the product. The module is easy to improve and develop for further use in future projects. MODX software is an excellent content management system for experienced PHP programmers. For novice web page builders content management systems such as Joomla! and WordPress are recommended. Both of the above mentioned systems are easier to use and require less programming than MODX. Also Joomla! and WordPress have a wide range of components ready for use after the installation of the software. Whilst MODX system does not have so many ready to use components, its reputation is based on its widely user friendly editing mode.

Keywords MODX, content management system, CMS, module

Pages 28 p. + appendices 34 p.

Sanasto

Ajax	Tekniikka monipuolisten verkkosovellusten luomiseen.
CMS	Content management system eli sisällönhallintajärjestelmä. Internet-sivujen hallintatyökalu, jolla helpotetaan sivustojen teko- ja päivitysprosessia.
CSS	Muotoilukieli, jolla voi määrittää Internet-sivujen ulkoasua ja tyylejä. Käytetään hyväksi XHTML-kielen kanssa.
Flash	Ohjelmointikieli, jolla tehdään pääasiassa dynaamisia web- ja mobiilisovelluksia.
Java	Ohjelmointikieli, jolla tehdään myös web-sovelluksia.
JavaScript	Verkkosivuille interaktiivisuutta lisäävä ohjelmointikieli. Koodi on sisällytetty XHTML-koodin sisälle tai omaan tiedostoonsa.
Loppukäyttäjä	Tuotteen käyttäjä eli asiakas.
MySQL	Avoimen lähdekoodin perustuva tietokanta, jota esimerkiksi MODX ja Joomla! käyttävät hyväksi.
Open Source	Avoimen lähdekoodin ohjelma, jota käyttäjä voi käyttää ilmaiseksi.
PHP	Ohjelmointikieli, jota käytetään useasti web-ohjelmoinnissa sekä sisällönhallintajärjestelmissä. PHP-koodikielellä kommentti merkataan //, alla esimerkki kommentista. <pre><?php echo "tämä teksti tulostuu"; // tämä on kommentti ?></pre>
UTF-8	Merkintästandardi, jota tietokannat ja Internet-sivut käyttävät hyväkseen. UTF-8 merkintästandardin yhteydessä voi käyttää ääkkösiä hyväksi.
Ylläpitäjä	Tuotteesta vastaava henkilö, joka päivittää ja hallinnoi tuotteen oikeuksia.
(X)HTML	Muotoilukieli Internet-sivujen tekoon.

SISÄLLYS

1	JOHDANTO.....	1
2	INTERNETIN MAAILMA.....	2
2.1	Sisällönhallintajärjestelmät.....	2
2.1.1	Joomla!.....	2
2.1.2	Drupal.....	2
2.1.3	WordPress.....	3
3	MODX.....	3
3.1	MODX Evolution.....	4
3.2	MODX Revolution.....	4
3.3	Sivupohjat.....	5
3.4	HTML-palaset.....	6
3.5	PHP-palaset.....	6
3.6	Liitännäiset.....	7
3.7	Moduulit.....	7
4	YHTEYSTIETOMODUULI.....	8
4.1	Yhteystietomoduulin suunnittelu.....	8
4.1.1	Yhteystietomoduuli ylläpitäjän näkökulmasta.....	8
4.1.2	Yhteystietomoduuli loppukäyttäjän näkökulmasta.....	9
4.2	Testisivun suunnittelu.....	10
4.3	Toteutus.....	10
4.3.1	Yhteystietomoduulin teko ylläpitäjän näkökulmasta.....	11
4.3.2	Yhteystietomoduulin teko loppukäyttäjän näkökulmasta.....	18
4.4	Ongelmat.....	22
5	JATKOKEHITTÄMINEN.....	23
5.1	Näytä yhteystiedot.....	23
5.2	Lisää yhteystieto.....	23
5.3	Muokkaa yhteystieto.....	24
5.4	Asetukset.....	24
5.5	Hakuominaisuus loppukäyttäjän näkökulmasta.....	25
6	YHTEENVETO.....	25
	LÄHTEET.....	27
Liite 1	index.php	
Liite 2	yhteystieto.css	
Liite 3	nayta.php	
Liite 4	lisaa.php	
Liite 5	lisaa2.php	
Liite 6	muokkaa.php	
Liite 7	muokkaa2.php	
Liite 8	poista.php	

Liite 9	poista2.php
Liite 10	Sivupohja
Liite 11	css_tiedosto.css
Liite 12	[[haku]]
Liite 13	[[haku_toiminto]]
Liite 14	Sivupohja_kartta_sivulle
Liite 15	[[Haku_Etela]]
Liite 16	Ohjeet

1 JOHDANTO

Sisällönhallintajärjestelmät ja sovelluskehikset ovat nykyaikainen tapa tehdä Internet-sivuja. Näillä järjestelmillä varsinkin sivujen päivitys onnistuu helposti ja sivujen ylläpitoa voi jakaa myös muille kuin sivujen koostajalle. Nämä järjestelmät mahdollistavat sen, että asiakkaat voivat itse saada sivuille käyttöoikeudet ja he voivat myös itse päivittää ja ylläpitää sivujen eri osia. Olen työharjoittelussa Mainostoimisto Parooninmäen Moukarissa, jossa Internet-sivujen tekoon käytetään MODX-järjestelmää. MODX on erittäin monipuolinen avoimen lähdekoodin julkaisujärjestelmä, jossa yhdistyvät sisällönhallintajärjestelmän ja sovelluskehiksen ominaisuudet (content management framework).

Opinnäytetyön tavoitteena on saada Parooninmäen Moukarille toimiva yhteystietomoduuli MODX-järjestelmään, jota yritys voi käyttää hyväksi eri projekteissaan. Moduulilla asiakkaat voisivat itse lisätä, muokata, tallentaa ja poistaa yhteystietojaan MODX-järjestelmän kautta. Opinnäytetyö on kehittämisprojekti ja opinnäytetyössä on käsitelty työn suunnittelua ja toteutusta. Osa työhön käytetyistä ohjelmointikoodeista on liitteenä opinnäytetyön lopussa. Opinnäytetyössä perehdyn myös MODX:n maailmaan ja sen tarjoamiin mahdollisuuksiin sekä ominaisuuksiin. Pyrin löytämään opinnäytetyössäni vastaukset seuraaviin kysymyksiin: mikä on MODX-järjestelmä? Miten tehdään toimiva yhteystietomoduuli MODX-järjestelmään?

Työharjoittelupaikassani Parooninmäen Moukarissa esitettiin toive yhteystietomoduulista. Monilla heidän asiakkaillaan on sivuillaan yhteystietolistausta, joka vaatisi päivitystä ja helpompaa järjestämistä. Tämän hetken yhteystiedot tulevat suoraan tiedostosta, joten asiakkailla ei ole mahdollisuuksia itse päivittää niitä. Yhteystietomoduuli MODX-järjestelmässä antaisi asiakkaille mahdollisuuden itse hallinnoida omia yhteystietojaan, mutta silti ylläpito-oikeudet moduulista jäivät Parooninmäen Moukarille. Toimintatavalla korostettaisiin myös järjestelmän käytettävyyttä, sillä järjestelmien kuten MODX on tarkoitus antaa Internet-sivun asiakkaalle mahdollisuus hallinnoida omaa sivustoaan puuttumatta itse koodiin. Ratkaisujen pitää olla asiakkaan näkökulmasta helppoja ja selkeitä, jotta yhteistyö toimii asiakkaan ja sivun ylläpitäjän välillä.

Parooninmäen Moukari on Riihimäellä toimiva mainostoimisto. Yritys on perustettu vuonna 1995 ja sillä on nykyisin yhdeksän työntekijää. Yritys toimii laajalti mainosalalla, graafisesta ja mainonnan suunnittelusta Internet-sivujen tekoon. Moukarilla on lukuisia suuryrityksiä asiakkanaan ja Internet-sivujen suunnittelu, toteutus ja ylläpito muodostavat suuren osan yrityksen toiminnasta.

Opinnäytetyön aihe on mielenkiintoinen, sillä olen kiinnostunut web-ohjelmoinnista ja Internet-sivujen rakenteista. Olen innostunut työskentelemään MODX-järjestelmän kanssa ja varsinkin syventymään sen lukuisiin mahdollisuuksiin.

2 INTERNETIN MAAILMA

Internet-sivut ovat kokeneet mullistavia muutoksia sen historian aikana. Tällä hetkellä sivut ovat interaktiivisia ja moniulotteisia kokemuksia, joiden toteutukseen käytetään monia erilaisia koodikieliä ja ohjelmia. Internet-sivut joiden toteutukseen käytettiin vain HTML-muotoilukieltä, ovat jääneet menneisyyteen. Nykyisin Internetissä löytää HTML-kielen lisäksi CSS-muotoilukieltä, PHP-ohjelmointikieltä, Flashiä ja JavaScriptiä monien muiden joukossa. Internet on myös jatkuvan muutoksen kohde ja sivujen sisällön muokkauskeinot ovat koko ajan kehittyneissä.

2.1 Sisällönhallintajärjestelmät

Sisällönhallintajärjestelmät eli content management systems (CMS) ovat nykyaikainen tapa tehdä Internet-sivuja. Sisällönhallintajärjestelmällä sivujen hallinta on helpompaa ja hallinnan oikeuksia voi jakaa eri henkilöille. Markkinoilla on useita maksullisia kuin myös Open Source versioita sisällönhallintajärjestelmistä. Tämän hetken käytetyimpiin sisällönhallintajärjestelmiin kuuluvat Joomla, Drupal ja WordPress (Most Popular CMS 2011).

Sisällönhallintajärjestelmät mahdollistavat Internet-sivun tekijöille ja heidän asiakkailleen helpomman ympäristön päivittää sivuja. Sivuston laajuus ei ole ongelma ja sivuston materiaalit ovat siististi omilla paikoillaan, kuten kirjat kirjastoissa. Sivujen ylläpitäjä voi antaa tietyt rajatut oikeudet asiakkaalle, jolloin asiakas voi kirjautua sivuille omilla tunnuksillaan ja esimerkiksi päivittää sivujen uutisia. Sisällönhallintajärjestelmällä on usein oma tekstieditorinsa, jolla sivuja voi helposti päivittää. Editorin kautta sivulle voi laittaa tekstiä, kuvia, linkkejä tai videoita. Hyvän sisällönhallintajärjestelmän pitää olla tarpeeksi helppo käyttää vasta-alkajalle, mutta antaa silti rajattomat mahdollisuudet ammattilaiselle sivujen toteutuksesta.

2.1.1 Joomla!

Joomla! on yksi tämän hetken suosituimmista sisällönhallintajärjestelmistä. Sillä on vankka perusta sisällönhallintajärjestelmästä ja sillä on tällä hetkellä yli 200 000 jäsenen yhteisö, joka koostuu Joomla:n käyttäjistä ja sen kehittäjistä (Mikä on Joomla? 2011). Joomla! on Open Source-ohjelma. Perus web-ohjelmoinnin syntaksien XHTML:n ja CSS:n lisäksi Joomlaa ohjelmoidaan myös PHP:lla sekä tietokantana on MySQL. Joomla! on varmasti helppo omaksua käytettäväksi ja siitä löytyy paljon valmiita komponentteja asennettaviksi. Joomla!lle löytyy paljon apua Internetistä ja sillä on myös omat sivunsa suomen kielellä.

2.1.2 Drupal

Drupal on sisällönhallintajärjestelmä ja sovelluskehysalusta. Se käyttää hyväksi PHP-ohjelmointikieltä ja on avoimen lähdekoodin ohjelma. Drupalin historia ulottuu vuodelle 1999 ja tämän hetken uusin versio Drupalista on 7.9. Drupalissa on valmiiksi moduuleita ja teemoja mitä voi käyttää

halutessaan. Siihen voi myös asentaa muiden tekemiä lisämoduuleita ja teemoja, joita löytyy tuhansia. (About Drupal n.d.)

Suomessa esimerkiksi YLE, Nelonen ja Nokia käyttävät hyväkseen Drupalia (Drupal Suomi n.d.). Drupalista löytyy paljon kirjallisuutta englanniksi ja Internetistä löytyy paljon tutoriaaleja sekä ohjeita Drupalin käyttöön.

2.1.3 WordPress

WordPress on suosittu julkaisualusta, jota käytetään varsinkin blogien ja yksinkertaisten Internet-sivujen tekoon. WordPressin ideana on pitää peruskoodi ja julkaisualusta kevyenä ja nopeana käyttäjälle, mutta tarvittaessa sivustoa voi laajentaa omien toiveiden mukaan (Extend n.d.). WordPressin uusin versio on 3.2.1 (Viimeisin julkaisu 2011).

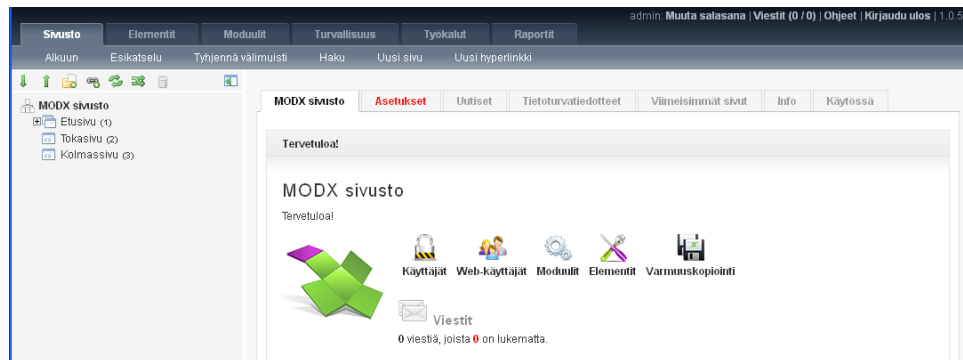
WordPressin suosio perustuu myös sen käyttäjien tekemiin komponenttien ja niiden saatavuuteen. WordPress jäsenet käyttävät hyväkseen Codex-kirjastoa, josta löytyy käyttäjien tekemiä koodeja ja artikkeleita. (Suomi 2010.)

3 MODX

MODX on samalla sisällönhallintajärjestelmä ja kehyssovellus. Se hyödyntää molempien järjestelmien ominaisuuksia ja tämä onkin MODX:n yksi valttikortti. Järjestelmä soveltuu käytettäväksi ammattilaisille kuin aloittelijoillekin. MODX:n vahvuutena voi pitää sen mahdollisuutta toteuttaa sivut juuri tekijän suunnitelmien mukaan. Järjestelmä on muokattavissa tekijän toiveiden mukaan, koska järjestelmän eri osia on helppo muokata eikä kaikkia palasia ole tehty valmiiksi upotettaviksi sivuille.

MODX:n perusteena on, että sen käyttäjä tuntee (X)HTML- ja CSS-muotoilukielet ja osaa käyttää niitä. PHP-ohjelmointikielen osaaminen on myös suotavaa ja varsinkin monipuolisissa sivuissa sen osaaminen johtaa parempaan lopputulokseen. Silti MODX:ssä pärjää ilman PHP-osaamista ja sitä voi oppia sivujen tekoprosessin aikana.

MODX:n hallinta tapahtuu sen käyttäjähallinnosta (Kuva 1). Käyttäjähallintaan kirjaututaan selaimen kautta ja osoitteeksi määritellään <http://sivunnimi.com/manager>. Ensin kirjaututaan sisään käyttäjätunnuksen ja salasanan kautta. Kirjautumisen jälkeen käyttäjähallinnossa aukeaa MODX-järjestelmän aloitussivu. Yläpalkista löytyvät kaikki tarvittavat työkalut, jotta sivuja pääsee rakentamaan ja muokkaamaan. Käyttöjärjestelmän rakenne on kattava ja sieltä käsin pääsee muokkaamaan kaikkia sivustoon liittyviä asetuksia. Vasemmalla laidassa on puolestaan sivuston rakenne, josta löytyy Internet-sivujen hakemistopuu. Jokaisen sivun perässä on sivulle oma yksilöivä ID-numero.



Kuva 1. Näkymä MODX-käyttäjähallinnon etusivusta.

MODX on saanut alkunsa vuonna 2004. Nykyisin MODX:stä on olemassa kaksi versiota: MODX Evolution ja MODX Revolution. Molemmat versiot on julkaistu vuonna 2009. (Ray 2011, 12-13.)

3.1 MODX Evolution

MODX Evolution eli MODX Evo, kuten sitä MODX piireissä kutsutaan, on julkaisusta vanhempi ja vakaampi. Uusin versio Evolutionista on 1.0.5. Tällä hetkellä se on vielä suuremmassa suosiossa kuin uudempi MODX Revolution. Se perustuu vanhaan koodipohjaan ja siitä löytyy huomattavasti enemmän ladattavia komponentteja. Monet sivustot Internetissä käyttävät Evolutionia ja sitä on vuosien aikana paljon testattu. (Ray 2011, 13.)

3.2 MODX Revolution

MODX Revolution eli MODX Revo on MODX:n tulevaisuus. Se on vakaampi ja teknologialtaan paremmin rakennettu kuin edeltäjänsä. Se toimii nopeammin kuin Evolution, mutta vie myös hieman enemmän palvelintilaa (32M). Molemmat järjestelmät ovat samankaltaisia, mutta suurimmat erot järjestelmistä löytyvät uusien snippettien asennuksista. Revolutionissa on mahdollista ladata uudet komponentit suoraan MODX-järjestelmän sisään. (Ray 2011, 13.)

Evolutionissa tämä toiminta tehtiin monimutkaisemmin. ZIP-tiedosto piti ladata koneelle, jonka jälkeen tiedostojen sisällöt piti kopioida MODX:n sisälle tarvittaviin kohtiin. Tämä tapa aiheutti enemmän virhetilanteita ja oli myös hitaampaa.

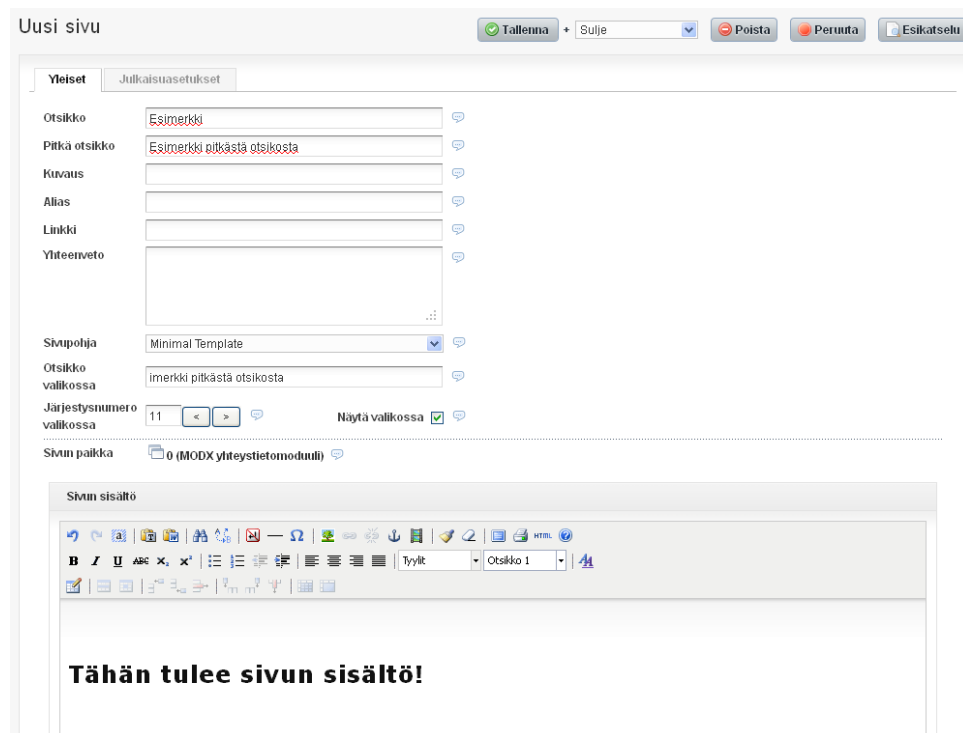
Uudessa järjestelmässä on myös helpompi ottaa yhteys MODX:n omaan tietokantaan. Evolutionin päivitys Revolutioniin pitäisi onnistua helposti, suurimmat ongelmat ovat nimenomaan komponenttien toiminnallisuudessa. Kaikki Evolutionin komponentit eivät vielä toimi Revolutionissa. Tämä on myös hidastanut käyttäjien kynnystä päivittää versioitaan. (Ray 2011, 14.)

3.3 Sivupohjat

MODX:ssä kuten muissakin sisällönhallintajärjestelmissä käytetään sivupohjia. Sivupohjia voi olla yhdellä Internet-sivustolla useita tai vain yksi. Se voi sisältää tekstiä, kuvia, ääntä tai melkein mitä vain (Antano 2011, 51). Sivupohjaa voi muokata jokaisen projektin tarpeisiin.

Sivupohja koostuu XHTML-koodista ja sisältää MODX:n omia tageja, joilla sisältöä kutsutaan sivuille. Nyrkkisääntönä on, että älä laita Internet-sivun sisältöä sivupohjaan. MODX:n tarkoituksena on, että sivupohjaa hallinnoidaan tagien avulla. (Ray 2011, 198.)

Internet-sivun sisältö kutsutaan [`*content*`] tagilla sivupohjassa. Sisältö tulee esille jokaisen oman sivun kohdalle (Kuva 2). Esimerkistä voi huomata kuinka sivulle voi tehdä otsikon, pitkän otsikon ja vielä erikseen otsikon, jotka näkyvät valikossa. Vaihtoehtona on myös tehdä sivu piiloon loppukäyttäjiltä, jolloin voi rastita ruutuun vaihtoehdon ”Näytä valikossa”. Sivun sisältöä voi kirjoittaa kohtaan, kuten esimerkissä näkyy.



Kuva 2. Sivun rakennenaikymä.

Sivun sisällön rakennetta pääsee muokkaamaan sivupohjien muuttujilla. Tämä on myös kätevä ominaisuus, jos sivun sisältöön haluaa erilaisia kohtia. Tämä mahdollistaa sivun sisältöosuuden pilkkomista osiin tai sitten lisäämällä uusia osia sisällön lisäksi. (Ray 2011, 226.)

Koodi esimerkistä voi nähdä miten sivupohjan aloitus eroaa normaalista XHTML-koodista. [`(base_url)`] on MODX:n oma tagi.

```
<link
href="[(base_url)]assets/templates/sivu/css_tiedosto.css"
rel="stylesheet" type="text/css" />
```

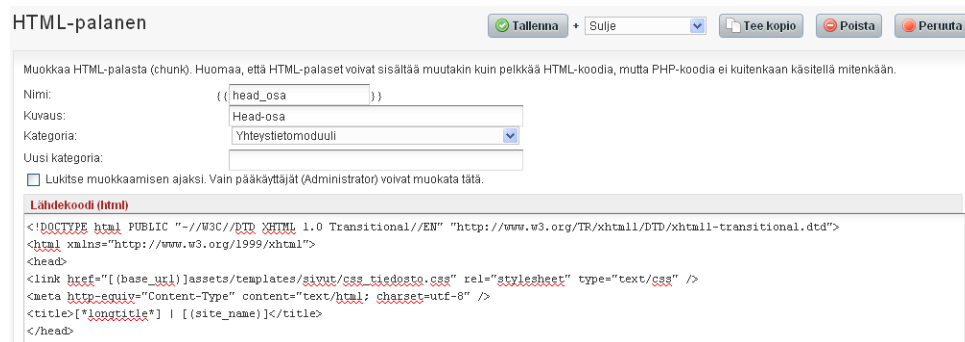
MODX:ssä voi myös hakea sivun otsikot näppärästi longtitle- ja site_name-tageilla kuten koodi esimerkissä näkyy.

```
<title>[*longtitle*] | [(site_name)]</title>
```

Tämä koodinpätkä näyttää Internet-sivun kävijälle jokaisen sivun pitkän otsikon ensiksi ja sitten vasta koko sivun nimen. Jokaiselle omalla sivulle MODX:ssä voi antaa oman pitkän otsikon.

3.4 HTML-palaset

Chunks eli HTML-palanen sisältää useimmiten sisältöä joka toistuu Internet-sivuilla. Se voi esimerkiksi sisältää jokaisella sivulla toistuvan koodinpätkän, esimerkiksi alavalikon. HTML-palaset voivat sisältää vain HTML-koodia ja tätä koodia kutsutaan sivuille omalla tyyllillä. Halutun sivun sivupohjaan voidaan merkitä koodia alla olevan kuvan mukaisesti. Esimerkissä on HTML-koodin head-osa tehty omaksi HTML-palaseksi, jolloin sitä ei tarvitse erikseen kirjoittaa muihin sivupohjiin. Head-osaa voi kutsua muiden sivujen koodissa tagilla `{{head_osa}}`.



Kuva 3. Esimerkki HTML-palasesta.

HTML-palasia voidaan siis kutsua tagien avulla sivupohjiin, mutta niiden käyttö on myös mahdollista linkittää PHP-palasiin tai liitännäisiin (Ray 2011, 229). Erityisesti sivupohjien kanssa HTML-palaset selventävät ja yksinkertaistavat koodia, koska eri sivun osat on jaoteltu omiin lokeroihin.

3.5 PHP-palaset

PHP-palanen voi nimensä mukaisesti sisältää vain PHP-koodia. Snippet-teiksi kutsuttavia paloja voi tehdä helposti itse ja tämä on juuri se ominaisuus joka on MODX:n valtti. PHP-palasia löytyy myös MODX:n sisään valmiiksi asennettuna. Näitä valmiita snippettejä voi halutessaan käyttää omilla sivuillaan. Valmiina sieltä löytyy myös esimerkiksi palautelomake tai kuvagalleria. Näitä snippettejä pääsee itsekin helposti muokkaamaan ja lisäämään omille sivuille. Lisää valmiita snippettejä löytyy myös MODX:n kotisivuilta. Evolution versioon löytyy vielä huomattavasti enemmän vaihtoehtoja, joka myös takaa Evolutionin jatkuvan suosion. Snippettejä kutsutaan tagilla `[[snippet_testi]]`.

Tagi hakee tällä hetkellä tiedot MODX:n välimuistista. Tämä voi jossain tilanteissa olla ongelmallista, jos tietoja päivitetään MODX:n käyttäjähallinnosta käsin. Tagin ominaisuuksia voi kuitenkin muuttaa ja esimerkiksi `[!snippet_testi!]` tagi haulla PHP-palanen päivittyy automaattisesti sivulla tapahtuvan päivityksen yhteydessä.

PHP-palasista löytyy myös valmiiksi asennettuna näppärä Wayfinder, jolla voidaan tehdä helposti navigointilinkit muille Internet-sivun sivuille. Sitä kutsutaan sivupohjaan alla olevalla koodilla.

```
[[Wayfinder? &startId=`0`]]
```

Tämä listaa kaikki Internet-sivun välilehtisivut yhteen navigoinnin helpottamiseksi. `&startId=`0`` tarkoittaa, että listaus alkaa sivulta, jonka ID-numero on nolla. ID-numeron arvoa on tietysti mahdollista muuttaa haluttuun numeroon. Wayfinder kannattaa laittaa omaan div-osaan sivupohjassa. Navigointilistauksen listaa voi muokata CSS-tyylitiedoston kautta. Wayfinder on erittäin monipuolinen PHP-palanen, jolla voi tehdä haluamansa valikon tutustumalla tarkemmin sen ominaisuuksiin.

3.6 Liitännäiset

Liitännäiset eli pluginit ovat myös PHP-koodia, mutta niitä ei kutsuta taageilla sivupohjien kautta. Sen sijaan liitännäinen käynnistyy kun sitä kutsutaan jonkun prosessin aikana. Liitännäisen koodi käynnistetään erilaisilla System Event-operaatiolla. Liitännäisiä varten MODX:ssä on valmiiksi asennettu System Eventit, jotka voi valita oman liitännäisensä käynnistämiseksi. Esimerkiksi `OnWebPagePredender` System Event käynnistää liitännäisen, kun MODX sivu lähetetään selaimelle. Pluginien käyttö vaatii hieman enemmän harjoitusta ja niiden käyttöä suositellaan kokeneimmille käyttäjille. (Ray 2011, 289-290.)

3.7 Moduulit

Moduuli tehdään MODX-järjestelmän sisään, josta sitä voi hallinnoida. MODX:n asennuksen jälkeen oletuksena on vain yksi valmis moduuli Doc Manager. Doc Managerilla voi hallinnoida MODX:n sisällä olevia dokumentteja. Valmiita moduuleja voi ladata myös MODX:n kotisivujen Extras-osiosta, josta löytyy moduuleita käyttäjän asennettavaksi.

Moduulia voi käyttää hyväksi, jos tarvitsee uutta toiminnallisuutta käyttäjähallintaan. Hyvä esimerkki voisi olla moduuli, joka hakee kaikkien sivupohjien tietoja ja näin käyttäjä voisi etsiä haluamansa uutisen sivupohjista. Moduulilla voi tehdä käyttöjärjestelmän, jossa tietoa voidaan tallentaa MODX:n omaan tietokantaan. Tästä moduulin toiminnasta on hyvä esimerkki kalenterimoduuli, jossa käyttäjä voi käydä lisäämässä moduulin kautta tietoa kalenteriin. Moduuli tallentaa tiedot tietokantaan, joita sitten snippettien tai liitännäisten kautta haetaan itse Internet-sivuille näkyviin.

Moduuleja ei ole enää MODX Revolutionissa vaan moodulit on korvattu Custom Manager Pagesilla. Toiminto pysyy pääasiallisesti samanlaisena eli MODX:n käyttäjähallinnan sisään tehdään uusi sivusto, jota hallinnoidaan MODX:n sisältä käsin. (Ray 2011, 712.)

4 YHTEYSTIETOMODUULI

Opinnäytetyön aiheena tehdään yhteystietomoduuli MODX-järjestelmään. MODX:n sisälle tallennetaan kokonaan uusi moduuli, joka koodataan PHP-koodikielellä sekä MySQL-tietokantaa käyttäen. MODX-järjestelmään pitää myös ohjelmoida PHP-paloja, jotta moduulia pystytään kutsumaan itse Internet-sivulle. Tämän lisäksi pitää koodata testisivu lopputuottajan näkökulmaa varten, joka tehdään HTML- ja CSS-muotoilukielillä. Testisivu upotetaan myös MODX:n sisään, jotta yhteystietomoduulia voi käyttää tässä ympäristössä. Testisivun toteutukseen käytetään hyväksi MODX:n sivupohjia, HTML-palasia, PHP-palasia ja lisäksi myös MODX:n omia tagi kutsuja.

4.1 Yhteystietomoduulin suunnittelu

Yhteystietomoduuli suunniteltiin yhdessä työpaikkaohjaajani kanssa. Aluksi listattiin kaikki tärkeät ominaisuudet, jotka yhteystietomoduulin tulisi pitää sisällään. Moduulin olisi tarkoitus toimia Parooninmäen Moukarin asiakkaiden eri projekteissa. Mallina yhteystietomoduulin tarvittavista yhteystietojen tietokentistä käytettäisiin Parooninmäen Moukarin asiakkaan kauppiasyhteystietoja. Tulen tekemään käyttöohjeet moduulista, jotta sen muokkaaminen eri asiakkaiden tarpeisiin on mahdollista tulevaisuudessa.

Moduulin tekovastuu tulee olemaan itselläni ja moduulista tulee jäämään Parooninmäen Moukarille pakattu ZIP-tiedosto, joka sisältää kaikki koodit sekä ohjeet moduulia varten.

4.1.1 Yhteystietomoduuli ylläpitäjän näkökulmasta

Ylläpitäjä voi moduulin kautta halutessaan listata kaikki yhteystiedot sekä lisätä, muokata ja poistaa yhteystietoja. Yhteystietojen lisäksi ylläpitäjällä on oikeus lisätä, muokata ja poistaa yrityskategorioita sekä paikkakuntia. Ylläpitäjä pääsee myös käsiksi moduulin asetuksiin, josta pystyy vaihtamaan yhteystietolomakkeen tietokenttiä. Yhteystietolomake moduulissa tulee sisältämään ainakin nämä välttämättömät tietokentät (Taulukko 1).

Taulukko 1. Yhteystietomodulin tietokentät ja niiden ominaisuudet.

Tietokenttä	Tietotyyppi	Julkinen
ID-numero	SMALLINT	Ei
Yrityskategoria	VARCHAR	Kyllä
Yrityksen nimi	VARCHAR	Kyllä
Katuosoite	VARCHAR	Kyllä
Postinumero	VARCHAR	Kyllä
Postitoimipaikka	VARCHAR	Kyllä
Paikkakunta	VARCHAR	Ei
Alue	VARCHAR	Ei
Puhelinnumero	VARCHAR	Kyllä
Fax-numero	VARCHAR	Kyllä
Sähköposti	VARCHAR	Kyllä
WWW-osoite	VARCHAR	Kyllä
Tuoteryhmä	VARCHAR	Kyllä
Tuoteryhmä2	VARCHAR	Kyllä
Tuoteryhmä3	VARCHAR	Kyllä
Tuoteryhmä4	VARCHAR	Kyllä
Julkinen	VARCHAR	Ei

MODX tallentaa tiedot omaan MySQL-tietokantaansa, jolloin ulkopuolista tietokantaa ei tarvita. Moduulia varten täytyy tehdä tietokantaan kolme eri taulua: yhteystieto, yrityskategoria sekä paikkakunta. Yhteystieto-taulu sisältää kaikki yllä mainitut tietokentät sekä jokaiselle yhteystiedolle yksilöllisen ID-numeron. Tuoteryhmätietokenttiä tulee tauluun yhteensä neljä. Jokainen tuoteryhmä on yksilöllisesti nimetty.

Jotta ylläpitäjä pääsee muokkaamaan yrityskategorioita ja paikkakuntia on nämä kaksi omassa taulussaan tietokannassa. Yrityskategoria-taulu sisältää kaikki yrityskategoriat ja paikkakunta-taulusta löytyvät paikkakunnat. Molemmissa tauluissa on myös yrityskategoria ja paikkakunta tietueille omat yksilölliset ID-numeronsa.

4.1.2 Yhteystietomoduli loppukäyttäjän näkökulmasta

Loppukäyttäjä tulee saamaan Internet-sivuilta kaikkien yhteystietojen listauksen valintalomakkeesta. Valintalomakkeessa tulee näkyä alue (esim. Etelä-Suomi, Itä-Suomi jne.) ja paikkakunta. Yhteystiedot on mahdollista näkyä sivuilla myös paikkakuntakohtaisesti. Jokaisesta yksittäisestä yhteystiedosta tulee näkyä kaikki tietotyypit, jotka on merkitty julkisiksi (Taulukko 1).

Valintalomakkeen lisäksi sivuille tulee myös kartta, josta voi klikata eri alueiden yhteystiedot näkyviin. Toivottava lisäominaisuus sivuille olisi myös yhteystietojen hakuominaisuus postinumeron tai yrityksen nimen perusteella.

4.2 Testisivun suunnittelu

Testisivuksi tulee Internet-sivu, jossa on kolme erilaista hakuvaihtoehtoa. Jokainen hakutoiminto tulee omaksi sivuksi Internet-sivuilla. Sivun pääsivulta on linkit jokaiseen sivuun. Ensimmäisenä on Haku-sivu, josta käyttäjä voi hakea alasvetovalikon kautta halutun yhteystiedon näkyville sivulle. Toisena hakuvaihtoehtona on Haku2-sivu, jossa on mahdollista hakea yhteystietoja tekstikentän avulla yrityksen nimen perusteella. Viimeinen hakuvaihtoehto on Kartta-sivu, jossa on Suomen kartta. Suomen kartta on jaettu kuuteen alueeseen Suomen läänien mukaisesti. Klikkaamalla haluttua lääninä sen alueen yritykset listautuvat esille. Jokainen hakutoiminto näyttää vain yhteystiedot, jotka on määritelty moduulin asetuksista julkisiksi. Sivulta löytyy myös Ohjeet-sivu, josta löytyy pakattu ZIP-tiedosto käyttäjälle. ZIP-tiedosto sisältää kaikki koodit ja ohjeet moduulin käyttöä varten.

Testi Internet-sivun on tarkoitus olla mahdollisimman selkeä ja yksinkertainen, kunhan yhteystietojen hakutoiminnot toimivat ja vastaukset tulevat selkeästi esille sivulle. Haut ovat kuitenkin vain suuntaa antavia ja Parooninmäen Moukarin pitää räätälöidä nämä toiminnot uudelleen kun he tarvitsevat yhteystietomoduulia.

4.3 Toteutus

Opinnäytetyö aloitettiin asentamalla MODX tietokoneelle. MODX asennus sujui helposti MODX:n kotisivuilta löytyviä ohjeita noudattaen. Tietokantayhteytenä toimii Parooninmäen Moukarin oma MySQL tietokanta, jossa modx_demo tietokanta sijaitsee. Yhteystietomoduulia varten tehtiin kolme taulua kantaan, joita tarvitaan työssä. Ensimmäinen taulu on nimeltä yhteystieto ja se sisältää kaikki tarvittavat tietotyypit suunnittelun mukaisesti (Taulukko 1).

Taulut luotiin SQL-komennolla `create table`. Taulun ID-numerosta tehtiin INT-tyyppinen tietotyyppi. Tämän jälkeen lisättiin kaikki tarvittavat kentät tauluun. ID-numero on yksilöllinen ja se kasvaa automaattisesti (`auto_increment`) jokaisen uuden tietueen tullessa tauluun. Tällä varmistetaan, että jokaisella uudella tiedolla on oma yksilöivä ID-numeronsa. Muut tietokentät taulussa ovat VARCHAR-tietotyyppisiä. VARCHAR on yksi yleisin käytetyimmistä tietotyypeistä ja sitä on helppo käyttää, jos esimerkiksi merkkijonojen koot vaihtelevat (Kolehmainen 2007, 285). Olen antanut sarakkeille eripituiset arvot riippuen siitä mitä halutaan kantaan tallentuvan. Tuoteryhmä- ja julkinen sarakkeet ovat kaikki vain yhden merkin pituisia, koska niihin tallentuu vain joko X-merkki tai tyhjä arvo. Tyhjällä arvolla tuoteryhmä ei ole käytettävissä ja X-merkillä se on käytössä. Sama pätee myös Julkinen-sarakkeeseen, tyhjällä arvolla se ei ole julkinen, kun taas X-merkillä siitä tulee julkinen. Muiden VARCHAR-tietotyyppi sarakkeiden arvot vaihtelevat viidestä kolmeen kymmeneen riippuen sarakkeesta ja siihen tallentuvasta tiedosta.

Oletusarvona kaikki tietotyyppien arvot ovat tyhjinä. Sarakkeita tehdessä ei ole käytetty ääkkösiä, jotta välttyttäisiin turhilta virhetilanteilta. Tietokanta on UTF-8 merkintästandardien mukainen.

Seuraava taulu luotiin yrityskategorioille nimeltä kategoria. Taulu sisältää kaksi saraketta eli ID-numeron ja yritys_kategorian. ID-numero on taas auto_increment ja INT-tyyppinen. Yritys_kategoria sarake on VARCHAR-tietotyyppinen ja pituudeltaan 30. Tämä taulu luotiin, jotta käyttäjä voi erikseen hallinnoida yrityskategorioita.

Viimeinen taulu on paikka. Se sisältää sarakkeet ID-numero ja paikkakunta. ID-numero on INT-tyyppinen ja sillä on auto_increment ominaisuus. Paikkakunta sarake on VARCHAR-tietotyyppinen ja pituudeltaan 20. Tämä taulu luotiin myös, jotta käyttäjällä on mahdollista hallinnoida paikkakuntia. Taulut kategoria ja paikka ovat molemmat UTF-8 merkintästandardien mukaisia. Taulujen kategoria- ja paikkatiedot tulevat näkyviin yhteystietomoduulissa ja niiden tiedot tallentuvat omiin tauluihin, kuin myös yhteystieto tauluun.

4.3.1 Yhteystietomoduulin teko ylläpitäjän näkökulmasta

Yhteystietomoduulin teko aloitettiin tekemällä uusi moduuli Yhteystietomoduuli MODX-järjestelmään (Kuva 4). Moduulia tehtäessä on mahdollista antaa kuvaus tai lisätä pieni kuvake. Näitä kaikkia ominaisuuksia voi myös muokata myöhemmin. Moduulin voi laittaa jo jonkun tehdyn kategorian alla tai sille voi tehdä kokonaan uuden kategorian. Tässä tapauksessa sillä ei ole merkitystä, mutta kategoria helpottaa lähinnä MODX:n rakenteen sisällä olevia komponentteja. Jos sivupohjat, PHP-palaset ja moduulit tallennetaan aina saman kategorian alle, niitä on helpompi etsiä MODX käyttäjähallinnosta.

Muokkaa moduulia

Tallenna + Sulje Poista Peruuta

Muokkaa moduulia

Muokkaa moduulia. Moduuli on kokoelma elementtejä (esim. liitännäisiä, PHP-palasia jne.).

Yleiset Asetukset Riippuvuudet

Moduulin nimi: Yhteystietomoduuli

Kuvaus:

Kuvake (32x32): Selaa

Kategoria:

Uusi kategoria:

Elementti:

Ota moduuli pois käytöstä

Lukitse muokkaamisen ajaksi. Vain pääkäyttäjät (Administrator) voivat muokata tätä.

Lähdekoodi (php) Automaattinen rivitys

```
include_once $modx->config['base_path'].'assets/modules/yhteystietomoduuli/index.php';
```

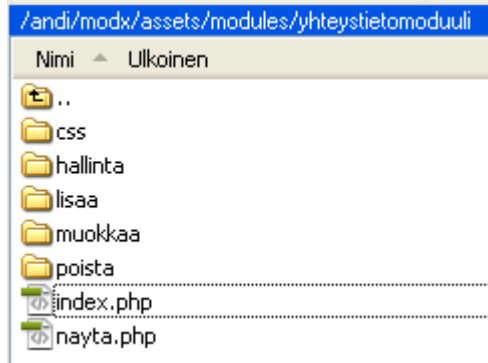
Kuva 4. Moduulin tekoa.

Moduulin lähdekoodin voi sisällyttää jo tähän sivulle, mutta on helpompaa jos kaikki moduulin liittyvät koodit ovat samassa kansiossa. Yhteystietomoduuliin tulee myös sen verran paljon koodia, että ne eivät mahtuisi yhteen tiedostoon. Laitoin linkin index.php tiedostoon, joka on moduulini aloitussivu. Komennolla include linkitin moduulin oikeaan tiedostoon.

```
include once $modx->config['base_path'].'assets/modules/yhteystietomoduuli/index.php';
```

Tämän jälkeen moduulin voi tallentaa ja suorittaa. Moduuli on nyt käytössä, kuitenkin ilman mitään toiminnallisuutta.

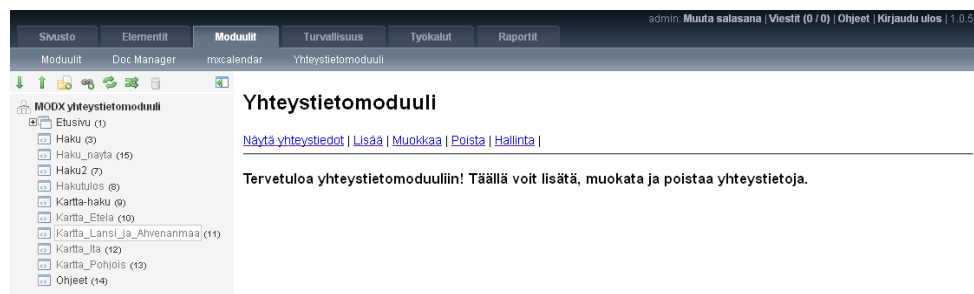
Tulen hallinnoimaan kaikkia moduulin tiedostoja FTP-yhteyden kautta ja tiedostot sijaitsevat /andi/modx/assets/modules/yhteystietomoduuli/ tiedostopuun alla (Kuva 5). Tiedostot on selkeästi jaettu omiin alikansioihin, jossa kyseisen toimenpiteen tiedostot löytyvät. Index.php ja nayta.php ovat ainoat tiedostot, jotka ovat suoraan yhteystietomoduuli-kansion alla.



Kuva 5. Yhteystiemoduuli-kansion sisältö ja rakenne.

Jos ylläpitäjällä on vielä tarvetta päästä muokkaamaan tallennettuja tietoja, hän pääsee niihin käsiksi Moduulit → Moduulit välilehdeltä, jonne on lisätty kaikki moduulit ja niitä pääsee helposti hallinnoimaan klikkaamalla valittua moduulia.

Moduulin aloitussivulta index.php (Liite 1) on ylläpitäjän tarkoituksena päästä valitsemaan haluaako hän katsomaan, lisäämään, muokkaamaan tai poistamaan yhteystietoja. Jokaiselle sivulle on oma linkkinsä, josta tarvittavaa toimenpitoa pääsee tekemään. Aloitussivun (Kuva 6) tarkoituksena on vain toivottaa ylläpitäjä tervetulleeksi moduulin ja siitä voi valita haluamansa toiminnon tarpeensa mukaan.



Kuva 6. Aloitussivun näkymä.

Sivut tehtiin XHTML-merkkikielen ja UTF-8 merkkistandardien mukaisesti. Sivujen ulkoasu tehtiin omaan CSS-tyylitiedostoon (Liite 2), johon viitataan jokaisessa koodissa. CSS-tiedostossa määritellään yhteystietomoduulissa käytetyt fontit ja niiden koot leipätekstille sekä otsikoille.

Ensimmäisenä linkki käyttäjälle on Näytä yhteystiedot-sivu. Klikkaamalla tätä sivua käyttäjälle tulee esille taulukkoon kaikki yhteystietomoduulin tallennetut yhteystiedot. Nayta.php (Liite 3) tiedosto hakee tietokannasta

kaikki tallennetut yhteystiedot ensiksi alueen ja sitten paikkakunnan aakosjärjestyksen mukaan nousevassa järjestyksessä. Taulukossa on kaikki tietotyypit omassa sarakkeessaan listattuna ja jokaisen yrityksen tiedot sekä sen tietotyypit ovat käyttäjälle nähtävissä. Soluihin tulevat tiedot haetaan tietokannasta `<? echo $rows['sarakkeen_nimi']; ?>` komennolla. Sivulla ei ole muuta tarkoitusta kuin näyttää yritysten tiedot listattuna.

PHP-koodin sisään on määritetty yhteys tietokantaan ja tietokannan osat on helposti selitetty koodia lukevalle. Koodi on myös kommentoitu, joten tietokantayhteyden muuttaminen on tarvittaessa helppoa. Jos tietokantayhteys ei toimi, tulee käyttäjälle viesti moduuliin ”Tietokantayhteys ei onnistunut!”.

```
$host=""; // Host nimi
$username=""; // Mysql käyttäjänimi
$password=""; // Mysql salasana
$db_name=""; // Tietokannan nimi
$table_name="yhteystieto"; // Taulun nimi

// Yhteys tietokantaan
mysql_connect("$host", $username, $password) or
die("Tietokantayhteys ei onnistunut!");
mysql_select_db("$db_name") or die("Tietokantayhteys ei onnistunut!");

$sql="SELECT * FROM $table_name ORDER BY alue, paikkakunta
ASC";
$result=mysql_query($sql);
```

SELECT-lauseessa on määritelty mitä tietoa haetaan taulusta \$table_name eli taulusta yhteystieto. Nämä tiedot jaotellaan ORDER BY komennolla alueen ja paikkakunnan mukaan nousevassa järjestyksessä (ASC). Jos haluttaisiin käyttää laskevaa järjestystä, olisi ASC:n tilalle pitänyt laittaa komento DESC.

Lisää yhteystieto-sivulla käyttäjä pystyy lisäämään yhteystiedon järjestelmään. Sivun koostuu kahdesta tiedostosta. Ensimmäinen tiedosto lisää.php (Liite 4) sisältää lomakkeen, jonka käyttäjä täyttää lomake-ehtojen mukaisesti. Toinen tiedosto lisää2.php (Liite 5) ottaa yhteyden tietokantaan ja lisää uuden yhteystiedon järjestelmään lomakkeeseen täytettyjen tietojen mukaisesti.

Lisää.php tiedoston lomake tallentaa käyttäjän syöttämät tiedot ja lähettää ne eteenpäin lisää2.php tiedostoon, joka suorittaa tallennuksen tietokantaan. Lomakkeessa tekstikenttinä ovat yrityksen nimi, osoite, postinumero, postitoimipaikka, puhelinnumero, fax-numero, sähköposti ja www-osoite. WWW-osoitteeksi on valmiiksi annettu käyttäjille selaimissa useimmiten käytetty http:// - aloitus. Alasvetovalikkoina ovat yrityskategoria, paikkakunta ja alue. Jos alasvetovalikosta ei löydy oikeaa yrityskategoriaa tai paikkakuntaa niiden lisäämiseen on linkki. Alueita ei ole mahdollista muokata, ne on tehty käyttäjälle valmiiksi.

Tuoteryhmät käyttäjä voi ruksata valitsemalla tuoteryhmän neljästä vaihtoehdosta. Tuoteryhmiä on neljä kappaletta ja ne on nimetty tietokantaan tuoteryhma, tuoteryhma2, tuoteryhma3 ja tuoteryhma4. Käyttäjälle lisää yhteystieto-sivulle tuoteryhmille on annettu eri nimet: aseet, varusteet, asusteet ja muut. Viimeisenä valintana lisää yhteystieto-sivulla käyttäjän pitää valita onko yhteystieto julkinen vai ei (Kuva 7).

Lisää yhteystieto:

Yritys kategoria:	Jälleenmyyjä	Lisää uusi yritys kategoria
Yrityksen nimi:	<input type="text"/>	
Osoite:	<input type="text"/>	
Postinumero:	<input type="text"/>	
Postitoimipaikka:	<input type="text"/>	
Paikkakunta:	Espoo	Lisää uusi paikkakunta
Alue:	Etelä-Suomi	
Puhelinnumero (esim.09-5555050):	<input type="text"/>	
Fax-numero (esim.09-5555055):	<input type="text"/>	
Email:	<input type="text"/>	
WWW-osoite:	http://	
Tuoteryhmä Aseet	<input type="checkbox"/>	
Tuoteryhmä Varusteet	<input type="checkbox"/>	
Tuoteryhmä Asusteet	<input type="checkbox"/>	
Tuoteryhmä Muut	<input type="checkbox"/>	
Julkinen:	<input checked="" type="radio"/> Kyllä <input type="radio"/> Ei	
<input type="button" value="Lähetä"/>		

Kuva 7. Lisää yhteystieto.

HTML-koodi lisää.php-tiedostossa sisältää lomakkeen eli formin, johon käyttäjä lisää tiedot tekstikenttiin. Yrityskategoria ja paikkakuntien arvot haetaan omista tietokannan tauluista. Alue-valikko tulee saamaan ennalta määritellyt arvot, kuten alla olevassa koodista näkyy. Valueksi eli arvoksi annettiin alueet ilman ääkkösiä, koska tallentamisessa ilmeni ongelmia, joita ei saatu ratkeamaan. Alueen arvot tallentuvat tietokantaan ilman ääkkösiä, mutta ne eivät haittaa loppukäyttäjän näkymää, koska alasetelissä ne kuitenkin näkyvät ääkkösten kera.

```
Alue:
<select name="alue">
<option value="Etelä-Suomi">Etelä-Suomi</option>
<option value="Itä-Suomi">Itä-Suomi</option>
<option value="Lansi-Suomi ja Ahvenanmaa">
Länsi-Suomi ja Ahvenanmaa</option>
<option value="Pohjois-Suomi">Pohjois-Suomi</option>
</select> </br>
```

Tuoteryhmät käyttäjä voi valita checkboxilla. Jos käyttäjä klikkaa checkboxia, tulee kyseiselle tuoteryhmälle merkintä "X". Jos käyttäjä ei klikkaa tuoteryhmää tulee kyseiseen tuoteryhmään tyhjä arvo (""). Jokaiselle checkboxille annettiin arvot "X" ja "". Tyhjä arvo annettiin hidden määri-

telmällä alla olevan esimerkin mukaisesti. Samalle tuoteryhmälle on siis kaksi arvoa, joista toinen tallentuu järjestelmään.

```
<input type="hidden" name="tuoteryhma" value="" />
<input type="checkbox" name="tuoteryhma" value="X" />
```

Viimeisenä Julkinen-sarakkeelle tehtiin samantyyppinen ratkaisu kuin checkboxeille, mutta checkboxien sijaan käytettiin radio buttonia. Tämä ratkaisu tehtiin käyttäjäystävällisestä näkökulmasta, jotta käyttäjä huomaa selkeästi kysymyksen olevan eri kuin edellinen. Molemmalla vaihtoehdolle on omat arvot, ”X” ja tyhjä arvo ””. ”X” vaihtoehto on automaattisesti valittuna. Asiakkaat lisätessään luultavasti haluavat yhteystiedon julkiseksi, joten jos sen oletusarvona on tyhjä, voi asiakas ihmetellä miksi yrityksen tiedot eivät näe loppukäyttäjälle.

Lisaa2.php tiedostossa on heti tietokantayhteys yhteystieto-kantaan, jonne käyttäjän lisäämät uudet tiedot tulevat tallentumaan. Tiedot tallennetaan INSERT-komennolla ja \$_POST-metodilla tiedosto hakee kaikki edellisen tiedoston lisaa.php kirjatut tiedot kantaan.

```
$sql="INSERT INTO yhteystieto (id, yritys_kategoria, yritys_nimi, osoite, postinumero, postitoimipaikka, paikkakunta, alue, puhelinnumero, fax, email, www, tuoteryhma, tuoteryhma2, tuoteryhma3, tuoteryhma4, julkinen)
VALUES
('$POST[id]', '$POST[yritys_kategoria]', '$POST[yritys_nimi]', '$POST[osoite]', '$POST[postinumero]', '$POST[postitoimipaikka]',
'$POST[paikkakunta]', '$POST[alue]', '$POST[puhelinnumero]', '$POST[fax]', '$POST[email]', '$POST[www]', '$POST[tuoteryhma]',
'$POST[tuoteryhma2]', '$POST[tuoteryhma3]', '$POST[tuoteryhma4]', '$POST[julkinen]');"
```

Lopuksi varmistetaan käyttäjälle, että toiminto on onnistunut. Jos tallennus on onnistunut käyttäjä saa siitä ilmoituksen. Jos tallentaessa tulee virhe käyttäjälle ilmoitetaan: ”Virhe yhteydessä ja virheen koodi”. Lopuksi MySQL-yhteys suljetaan mysql_close komennolla alla olevan esimerkin mukaisesti.

```
if (!mysql_query($sql, $con))
{
    die('Virhe yhteydessä: ' . mysql_error());
}
echo "Tiedot on tallennettu onnistuneesti!";

mysql_close($con)
```

Lisäyksen jälkeen yhteystiedon ID-numero tulee yhteystiedolle automaattisesti, koska se on jo määritelty tietokanta asetuksissa. Tämän takia ID:lle ei ole omaa kenttää lisäys sivulla.

Yhteystietojen muokkaaminen tapahtuu muokkaa-sivulta, joka on tehty muokkaa.php (Liite 6) tiedostoon. Muokkaa-sivulla kaikki yhteystiedot ovat listattuna allekkain ensin alueen ja sitten paikkakunnan aakkosjärjestyksen perusteella a:sta ö:hön. Yhteystiedoista näkyy sen ID-numero, alue,

paikkakunta yrityskategoria, yrityksen nimi, osoite, postinumero ja posti-toimipaikka. Jokaisen yhteystiedon perässä on muokkaa-painike, josta ky-seistä yhteystietoa pääsee muokkaamaan.

Muokkaustilassa käyttäjä näkee kaikki yrityksen tiedot alla olevan kuvan mukaisesti.

Muokkaa yhteystietoa:

ID:	185
Yritys kategoria:	Jälleenmyyjä
Yrityksen nimi:	<input type="text" value="Yritys X"/>
Osoite:	<input type="text" value="Keskustie 5 B 2"/>
Postinumero:	<input type="text" value="02345"/>
Postitoimipaikka:	<input type="text" value="Espoo"/>
Paikkakunta:	Espoo
Alue:	<input type="text" value="Etelä-Suomi"/>
Puhelinnumero (esim.09-5555050):	<input type="text" value="09-5555050"/>
Fax-numero (esim.09-5555055):	<input type="text" value="09-5555055"/>
Sähköposti:	<input type="text" value="x@myyja.fi"/>
WWW-osoite:	<input type="text" value="http://www.myyja.fi"/>
Tuoteryhmä Aseet	<input checked="" type="checkbox"/>
Tuoteryhmä Varusteet	<input type="checkbox"/>
Tuoteryhmä Asusteet	<input checked="" type="checkbox"/>
Tuoteryhmä Muut	<input type="checkbox"/>
Julkinen:	<input checked="" type="radio"/> Kyllä <input type="radio"/> Ei
<input type="button" value="Tallenna muutokset"/>	

Kuva 8. Muokkaustilan näkymä.

ID-numero näkyy muokkaustilassa, mutta sitä ei ole mahdollista muuttaa. Samoin yrityskategorian ja paikkakunnan tiedot näkyvät muokkaustilassa, mutta niitä ei pysty muuttamaan. Muokkaustoiminto perustuu samaan lomake muotoon kuin lisää-sivulla oleva yhteystietolomake. Muokkaus-sivun tiedot sisältyvät yhteen PHP-tiedostoon muokkaa2.php (Liite 7). Lomake lähetetään Tallenne muutokset-nappia painamalla ja sama tiedos-to tallentaa kyseiset muutokset. Muokkaa2.php tiedostossa otetaan yhteys tietokantaan ja MySQL UPDATE-lauseella päivitetään tietokannan tiedot. Lopuksi varmistetaan käyttäjälle, että onnistuiko päivitys vai ei, sekä lo-puksi lopetetaan MySQL-yhteys.

Poista-sivulta pääsee poistamaan yhteystietoja. Poista sivun koodit on jaet-tu kahteen tiedostoon. Poista.php-tiedostossa (Liite 8) yhteystiedot ovat listattuna samalla tavalla kuin moukkaa-sivulla. Yritykset ovat listattuna alueen ja paikkakunnan mukaan ääkkösjärjestyksessä ja kaikkien vieressä on checkbox. Yhteystietoja on mahdollista poistaa yksi tai useampia sa-malla kertaa checkboxien kanssa. Käyttäjä valitsee valitut yhteystiedot ja poista-napin painalluksella yhteystiedot voi poistaa. Tein JavaScriptillä tarkistuksen käyttäjälle, että haluaako käyttäjä varmasti poistaa kyseiset yhteystiedot. JavaScript-koodi on sisällytettyinä html-dokumentin <head> tagien sisään.

```
<script language=javascript>
function verify(){
msg = "Haluatko poistaa valitun / valitut yhteystiedot?";
return confirm(msg);
}
</script>
```

Kun käyttäjä painaa poista-nappia, ilmestyy hänelle ponnahdusikkuna, jossa varmistetaan toiminto. Lopuksi käyttäjä saa ilmoituksen onnistuneesta poistosta tai virheestä ponnahdusikkunan painalluksen jälkeen. Tiedot tallentuvat tiedostoon poista2.php, joka tekee lopullisen poiston yhteystieto-aulusta.

Hallinta-välilehdeltä pystyy hallitsemaan yrityskategorioita ja paikkakuntia. Hallintavälilehden teko oli periaatteessa samaa koodia kuin yhteystietomoduulin muille sivuille, mutta pienemmässä mittakaavassa. Tietotyyppejä oli vähemmän kategorian ja paikka tauluissa. Koodissa jouduttiin lähinnä vaihtamaan SQL-komentoja, jolla hallinnoidaan mitä tietoja haetaan, päivitetään, poistetaan tai lisätään.

Hallintavälilehdelle mentäessä pystyy heti tallentamaan uusia yrityskategorioita tai paikkakuntia (Kuva 9). Jos käyttäjä painaa Tallenna-nappia kirjoittamatta tekstikentän mitään, tulee hänelle ponnahdusikkunan kautta ilmoitus, että ”Sinulta puuttuu nimi yrityskategorialle/paikkakunnalle”. Tämä varmistus tehtiin myös JavaScriptiä käyttäen.

Yhteystietomoduuli

[Näytä yhteystiedot](#) | [Lisää](#) | [Muokkaa](#) | [Poista](#) | [Hallinta](#) |

Yhteystietomoduulin hallintapaneeli. Täällä voit lisätä, muokata sekä poistaa yritys kategorioita ja paikkakuntia.

Lisää uusi yritys kategoria:

[Muokkaa yritys kategorioita](#) | [Poista yritys kategorioita](#) |

Lisää uusi paikkakunta:

[Muokkaa paikkakuntia](#) | [Poista paikkakuntia](#) |

Kuva 9. Hallintavälilehden näkymä.

Yrityskategorian ja paikkakunnan arvoja pääsee myös muokkaamaan ja poistamaan. Muokatessa ja poistaessa molempien taulujen tiedot tulevat esille taulukon muodossa. Muokkaus tapahtuu samalla tavalla kuten muokkaa yhteystietoja välilehdellä (Kuva 10).

Yhteystietomoduuli

[Näytä yhteystiedot](#) | [Lisää](#) | [Muokkaa](#) | [Poista](#) | [Hallinta](#) |

Valitse muokattava yritys kategoria:

ID	Yritys kategoria	
34	Jälleenmyyjä	<input type="button" value="muokkaa"/>
40	Valtuutettu jälleenmyyjä	<input type="button" value="muokkaa"/>

Kuva 10. Hallinta-välilehden muokkaus mahdollisuus.

Poista yrityskategorioita tai paikkakuntia tapahtuu samalla tavoin kuten aiemmin, eli on mahdollista checkboxien avulla poistaa yksi tai useampi yrityskategoria tai paikkakunta samanaikaisesti (Kuva 11). JavaScriptin avulla käyttäjältä varmistetaan haluaako hän poistaa tietoja vai ei.

Yhteystietomoduuli

[Näytä yhteystiedot](#) | [Lisää](#) | [Muokkaa](#) | [Poista](#) | [Hallinta](#) |

Poista yritys kategoria. Voit poistaa yhden tai useamman kategorian samanaikaisesti.

ID	Yritys kategoria
<input type="checkbox"/>	Jälleenmyyjä
<input type="checkbox"/>	Valtuutettu jälleenmyyjä

Poista

Kuva 11. Hallinta-välilehden poisto-sivu.

4.3.2 Yhteystietomoduulin teko loppukäyttäjän näkökulmasta

Yhteystietomoduuli toimii nyt ylläpitäjälle, joka haluaa käydä lisäämässä, muokkaamassa tai poistamassa yhteystietoja. Mutta Internet-sivuille moduuli ei ole vielä yhteydessä. Tähän tarvitsemme tehdä testi Internet-sivut ja sen jälkeen käyttää MODX:n PHP-paloja, jolla voimme linkittää halutut tiedot tietokannasta loppukäyttäjälle. Internet-sivu koostuu pääsivusta sekä haku- ja ohjeet-sivuista (Kuva 12).



Kuva 12. Pääsivun näkymä.

Ensimmäisenä tehtiin Internet-sivulle uusi sivupohja nimeltä Sivupohja (Liite 10) ja uusi css-tiedosto (Liite 11). Tämä sivupohja tulee olemaan sivupohjana kaikille sivun sivuille paitsi Kartta_Haun-sivulle. Sivupohja koostuu HTML- ja PHP-palasia kuten alla olevasta koodista näkyy. HTML-dokumentin head- ja footer-osa ovat molemmat HTML-palasia. Wayfinder PHP-palanen hakee kaikki sivulle tulevat alisivut.

```
{{head_osa}}  
  
<body>  
  
<div id="kokosivu">  
  
<div id="logo">  
</div>  
  
<div id="linkit">  
<hr />  
[[Wayfinder? startId=`0` &firstClass=`first`  
&hereClass=`active` ]]  
<hr />  
</div>  
  
<div id="sisalto">  
  
[*content*]  
  
</div>
```

```
</div>

{{footer}}

</body>
</html>
```

Internet-sivulle tehtiin kolme esimerkkisivua, jolla testaamme yhteystietomoduulia loppukäyttäjän näkökulmasta. Ensimmäisenä tehtiin MODX-järjestelmään Haku-niminen sivu. Tälle sivulle tehtiin alasvetovalikko hakuvaihtoehto.

Haku-sivun sisältökenttään linkitettiin PHP-palanen nimeltä `[!haku!]`. PHP-palasen koodia muokattiin, jotta se hakee yhteystiedot alasvetovalikkoon yrityksen nimen perusteella (Liite 12).

Seuraavalla sivulla on hakuominaisuus, jossa loppukäyttäjä voi hakea yhteystietoa tekstikenttään syötettynä. Haku tapahtuu yrityksen nimen perusteella. Haku kentän ominaisuutta voi kuitenkin muuttaa helposti jatkossa, mikäli sille on tarvetta.

Ensin tehtiin kaksi uutta sivua ”Haku2” ja ”Hakutulos”. Hakutulos-sivua ei julkaista, joten se pysyy piilotettuna. Seuraavaksi tehtiin uusi HTML-palanen MODX-järjestelmään. Alla oleva lomakekoodi kuuluu HTML-palaseen ja sitä kutsutaan `{{haku}}` tagilla.

```
<form method="post" action="[~8~]">
<input name="haku" size="25" type="text" />
<input type="submit" value="Hae!" />
</form>
```

Hakutulos-sivua kutsutaan sen omalla ID-sivunumerolla. Tässä tapauksessa se näkyy lomakekoodissa action-kohdassa `[~8~]`, siinä lähetetään lomaketiedot Hakutulos-sivulle. Seuraavana luotiin uusi PHP-palanen nimeltä ”haku_toiminto” (Liite 13). Koodi ottaa yhteyden tietokantaan ja hakee hakuehtoihin liittyvät yhteystiedot. Lopuksi Hakutulos-sivulle tehtiin linkki PHP-palaseen `[!haku_toiminto!]` tagin kanssa. Alla olevassa kuvassa on näytetty haku kentän toiminnallisuus loppukäyttäjän näkökulmasta.



Kuva 13. Haku2 sivun näkymä.

Viimeisenä esimerkkinä tehtiin kartta haku omalle sivulle ”Kartta-haku”. Tällä sivulla voi hakea yhteystietoja alueiden mukaan. Kartta on jaettu neljään osaan: Etelä-Suomi, Itä-Suomi, Länsi-Suomi ja Ahvenanmaa sekä

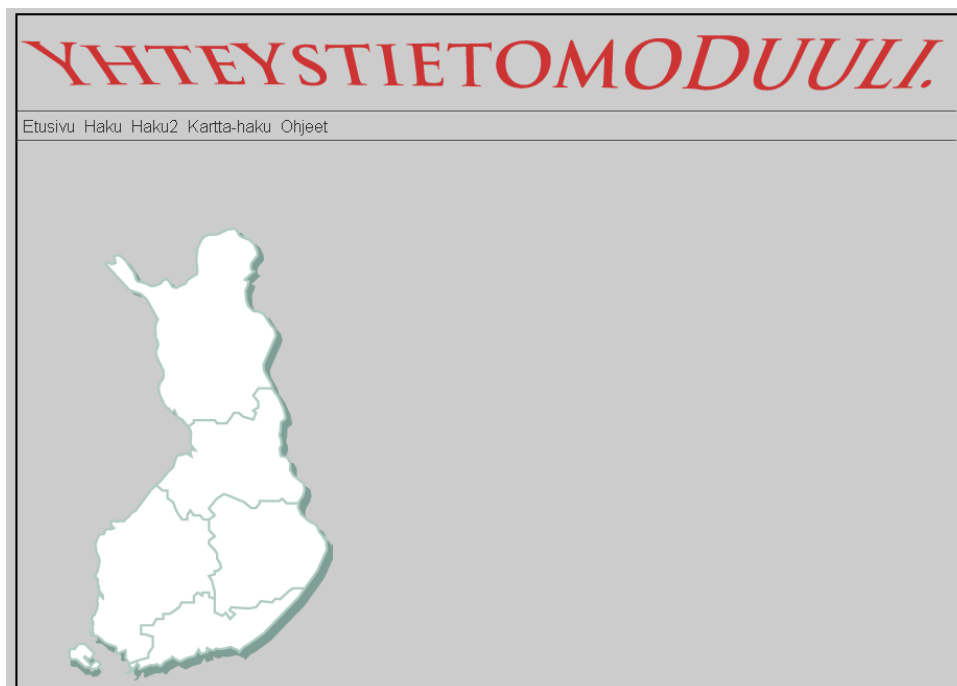
Pohjois-Suomi. Käyttäjä klikkaa aluetta ja alueen kaikki yhteystiedot tulevat näkyviin.

Ensimmäisenä luotiin uusi sivupohja nimeltä Sivupohja_kartta_sivulle (Liite 14). Sivupohja sisältää JavaScript-koodia, jolla on tehty Suomen kartan toiminnallisuus. Muuten uusi sivupohja noudattaa samaa kaavaa kuin vanha. Jokaista neljää aluetta varten tehtiin omat sivut oman alueen nimen mukaisesti (Kuva 14). Lihavoidut sivut tulevat loppukäyttäjälle esiin ja muut ovat piilotettuja sivuja.



Kuva 14. Sivuston rakenne.

Sivupohjaan on jokaiselle alueelle määritelty sen sivun oma ID-numero. Klikatessaan kartan aluetta Etelä-Suomi, sivupohja ottaa yhteyden Kartta_Etela sivuun. Kartta_Etela sivun sisällössä on linkki PHP-palaseen [Haku_Etela!], joka hakee kaikki julkiset yhteystiedot alueelta Etelä-Suomi (Liite 15) sekä näyttää ne käyttäjälle (Kuva 15).



Kuva 15. Kartta-haku.

Tämä toiminnallisuus on tehty myös Kartta_Lansi_ja_Ahvenanmaa, Kartta_Ita sekä Kartta_Pohjois – sivuille. Jokaiseen sivuun linkittyy oma PHP-palanen, joka hakee juuri sen alueen tiedot.

4.4 Ongelmat

Työn eteneminen oli ongelmallista. Aluksi oli vaikea saada yhteys tietokantaan, joka tuotti työn ensimmäisillä viikoilla ongelmia. Tietokantaa yhteystietoa yritettiin MODX:n omalla DBAPI-yhteyden kautta, mutta yhteys ei toiminut. Perinteisellä MySQL-yhteydellä yhdistäminen onnistui ja lopulta myös MODX:n oma DBAPI-yhteys on toiminut. Opinnäytetyössä käytetään kuitenkin perinteistä MySQL-yhteydenottoa, koska MODX:n DBAPI ei ole niin kehittynyt vielä Evolution versiossa.

Aluksi tietokannat olivat vaikeita hahmottaa ja en täysin tiennyt miten minun olisi pitänyt edetä niiden kanssa. Aluksi tein vain yhden taulun yhteystieto, jonka luulin riittävän. Kunnes ymmärsin, että suunnittelemani suunnitelmassa pitää päästä lisäämään, muokkaamaan ja poistamaan yrityskategorioita ja paikkakuntia. Tämän jälkeen tein kaksi uutta taulua kategoria sekä paikka. Aluksi oli ongelmallista siirtää kahden uuden taulun tiedot alasvetovalikkoon lisää-yhteystietoja sivulla. Muokkaa-yhteystietoja sivulle en saanut alasvetovalikkoon kategoria ja paikka taulujen tietoja, joten niitä ei pääse muokkaamaan kuin hallintavälilehden kautta.

Olen kohdannut ongelmia työn edetessä etenkin ääkkösten kanssa, koska ne ovat tallentuneet väärin joko tietokantaan tai itse moduuliin. Jokaisessa .php-tiedostossa on merkattu käyttöön UTF-8 merkintästandardi. Olen myös jossain MySQL komennoissa merkannut sen myös käyttöön komennolla.

```
mysql_set_charset("utf8")
```

Lopuksi merkkasin jo itse kooditasollakin ääkköset oikeilla merkintätavoilla.

```
ä = &#228; tai &auml;  
ö = &#246; tai &ouml;
```

Isoille ja pienille kirjaimille on eri merkintätavat, mutta Internetistä löytyy helposti listattuna kaikki oikeat merkkiviittaukset.

Hakuominaisuuden kanssa oli myös ongelmia. Toivottavaa oli, että käyttäjä voi muokkaa-sivulta alasvetovalikosta valita mitä yhteystietoa muokkaa. Tällä tavoin alasvetovalikkoon olisi tullut listattuna kaikki yhteystiedot. Samalla tavalla poista-sivulle olisi tarkoituksena tehdä hakukenttä, jolla käyttäjä voi hakea yhteystietoa nimen tai alueen perusteella ja sen jälkeen poistaa tämän.

5 JATKOKEHITTÄMINEN

Moduulia pystyy helposti räätälöimään sivujen tarpeitten mukaan ja sen laajentaminen on mahdollista. Vaikka Parooninmäen Moukari siirtyisi käyttämään MODX Revolutionia, yritys pystyy kuitenkin siirtämään yhteystietomoduulin uuteen järjestelmään. Koodikieli pysyy kuitenkin samana ja Revolutionissa moduuleilla on oma paikkansa Custom Manager Pagesissa.

Moduulin ohjeet ovat yksi osa kehittämisprojektia ja ohjeissa lukee miten mitään sivua pystyy muokkaamaan. Ohjeet kertovat yhteystietomoduulin asentajalle miten moduuli asennetaan ja miten se liitetään MODX-järjestelmään. Yhteystietomoduulin ohjeistus löytyy opinnäytetyön liitteenä (Liite 16). Tulevaisuudessa moduuliin voi kehittää uusia ominaisuuksia ja antaa käyttäjälle mahdollisuudet hallinnoida moduulia vieläkin enemmän.

Yhteystietomoduulia ja siihen liittyvää Internet-sivua on pääasiassa testannut tähän mennessä vain allekirjoittanut ja työpaikkaohjaajani. Jatkokehittämisen kannalta olisi hyvä jos testausta suoritettaisiin enemmän. Tällä tavoin virhetilanteet huomattaisiin ajoissa jo ennen kuin moduuli julkaistaan julkiseksi.

5.1 Näytä yhteystiedot

Näytä yhteystiedot-sivulle voisi laittaa esimerkiksi alasvetovalikon, josta käyttäjä voisi valita minkä sarakkeen perusteella hän haluaa listata yhteystiedot. Alasvetovalikkoon voisi laittaa vaihtoehdot ID-numero, yrityskategoria, alue tai paikkakunta. Näytä yhteystiedot-sivulla yhteystietolistaus voisi myös rajoittaa tietokenttiin ID-numero, yrityskategoria, yrityksen nimi, osoite, postinumero, postitoimipaikka, paikkakunta ja alue. Jos käyttäjä haluaa nähdä lisää tietoa jostain tietystä yhteystiedosta hän voi halutessaan klikata jokaisen yhteystiedon kohdalla olevaa lisätietoja painiketta, jolloin yhteystiedon kaikki tiedot tulevat esille joko omalle sivulle tai ne voisivat tulla käyttäjälle näkyviin lightboxin avulla. Lightbox tai shadowbox tuo Internetissä kuvia ponnahdusikkunan tapaisesti näkyville.

5.2 Lisää yhteystieto

Lisää yhteystieto-sivulle olisi hyvä tehdä lisää tarkistuksia ennen kuin käyttäjä voi lähettää lomakkeen. Sähköpostiosoitteen tarkistus olisi hyvä ominaisuus. Sähköposti-kenttä ei ole pakollinen kenttä lomakkeessa, mutta jos käyttäjä lisää siihen tietoa voisi tarkistus olla tarpeen. Tarkistuksessa voisi käyttää hyväksi JavaScriptiä ja se voisi tarkistaa onko sähköposti kentässä käytetty @-merkkiä ja onko kyseisen merkin jälkeisessä kahden sanan välissä pistettä. Tarkistukset on hyvä tapa varmistaa, että käyttäjälle ei tule turhia virheitä lomakkeen syöttö tilanteessa.

Lisää yhteystieto kohdassa voisi puhelinnumerolle ja fax-numerolle olla molemmille erikseen suuntanumero-tietokenttä, joka olisi lomakkeessa

erillisenä kenttänä. Tämä vaatisi yhteystieto-tauluun puhelinnumeron suuntanumeron ja fax-numeron suuntanumeron sarakkeiden lisäämistä. Tällä voisi välttää puhelin- ja fax-numeroiden väärinkirjoittamista sekä välttää eri tyylien kirjoittamista kenttiin. Tällä tavalla kenttiin voisi sisällyttää vain numeroita, eikä kenttiin saisi laitettua esimerkiksi mitään merkkejä suuntanumeron ja puhelinnumeron väliin.

Tällä hetkellä lisää yhteystieto-sivulla käyttäjä voi hakea alasvetovalikkoon tiedot yrityskategorioista, paikkakunnista sekä alueista. Alue kenttä luetaan suoraan HTML-tiedostosta ja yrityskategoria sekä paikkakunta ovat omia taulujaan tietokannassa. Jatkossa myös muille tietueille voi tehdä omat kannat esimerkiksi alue, postinumero ja postitoimipaikka tietueille.

MySQL-tietokantojen oikeaoppinen tallentaminen on yhteystietomoduulin käyttäjän vastuulla. Parooninmäen Moukarin henkilökunta saa itse päättää miten he suunnittelevat ja luovat jatkossa tietokannat, joita yhteystietomoduuli käyttää hyväkseen.

Tuoteryhmien tämän hetkisten arvojen sijaan tuoteryhmät näkyisivät loppukäyttäjälle kuvakkeina. Jokaisesta tuoteryhmästä olisi pieni kuvake, joka esittää tiettyä tuoteryhmää. Nämä valitut tuoteryhmät tulisivat näkyviin Internet-sivuille loppukäyttäjälle kuvakkeina.

5.3 Muokkaa yhteystieto

Muokkaa-sivulle pitäisi saada kaikki yhteystiedot alasvetovalikkoon, josta käyttäjä voisi valita mitä tiettyä yhteystietoa haluaa muokata. Muokkaa-sivulla ei pysty tällä hetkellä vaihtamaan yrityksen yrityskategoriaa tai sen paikkakuntaa. Nämä tiedot tulevat sivulle esille, kuten myös yrityksen ID-numero, mutta niitä ei voi muokata.

Poista-sivulle pitäisi myös saada hakukenttä, jolla voi hakea yhteystietoja nimen tai alueen perusteella.

5.4 Asetukset

Toivottava ominaisuus olisi, että ylläpitäjälle voisi tehdä kokonaan uuden välilehden Asetukset. Tällä voisi poistaa tämänhetkisen Hallintavälilehden ja hallinnoida yhteystietomoduulia laajemmin. Käyttäjä voisi lisätä, poistaa ja muokata yhteystietomoduulin sisältämiä tietokenttiä. Ulkoasua olisi mahdollista muokata asetuksista, esimerkiksi taustaväri olisi mahdollista vaihtaa sekä fontin väriä ja kokoa muokata. Asetuksetvälilehti antaisi käyttäjälle täydet oikeudet muokata yhteystietomoduulista omannäköisensä komponentin. Näitä ominaisuuksia pitäisi kuitenkin rajata ja antaa eritasoisia käyttöoikeuksia Parooninmäen Moukarille ja heidän asiakkaalleen.

5.5 Hakuominaisuus loppukäyttäjän näkökulmasta

Hakuominaisuuksia voisi myös muuttaa, että MODX-järjestelmässä ei tarvitsisi käyttää kahta tai useampia sivuja niiden ratkaisuun. Parempi tapa olisi, että koodi olisi tehty yhdellä sivulla toimivaksi. Tämä karsisi MODX sivujen määrää sen rakenteessa.

6 YHTEENVETO

Opinnäytetyön tutkimuskysymyksiin vastattiin onnistuneesti. Kehittämishanke onnistui ja yhteystietomoduuli tuli valmiiksi opinnäytetyön prosessin aikana. Opinnäytetyön tavoitteet tulivat toteen ja asiakas sai itselleen onnistuneen komponentin.

MODX-järjestelmä on mielestäni loistava ja suosittelen sitä kaikille web-ohjelmoinnista kiinnostuneille. Järjestelmänä se on käytännöllinen ja helppo ymmärtää. MODX ympäristön omaksuu helposti ja järjestelmän toiminnot ovat käytännönläheisiä. MODX:n suosio kasvaa varmasti lähivuosina myös Suomeen. Tämä voisi johtaa helpommin omaan MODX yhteisöön, jossa on helpompi vaihtaa ajatuksia suomen kielellä. MODX:n omilla kotisivuilla on Suomelle oma osionsa keskustelufoorumilla, mutta se on erittäin vähän käytetty ja sieltä löytyy vain muutamia keskusteluaiheita. Toisaalta monille englannin kieli on helppo ratkaisu Internet-sivuilla käymiseen, joten foorumeilla käynti ja niiden seuranta voi olla yhtä helppoa myös englanniksi. Yhteenvetona kuitenkin MODX:stä saisi olla Internetissä enemmän tutoriaaleja ja neuvoja sen käyttöön. Siitä on tällä hetkellä olemassa kaksi kirjaa, joista toinen ilmestyi juuri nyt marraskuussa 2011.

Sisällönhallintajärjestelmistä kiinnostuneille vasta-alkajille Internet-sivujen tekijöille Joomla! tai WordPress voi olla parempi ja helpompi vaihtoehto. Molempiin järjestelmiin on helppo asentaa jo valmiiksi tehtyjä komponentteja ja moduuleita. PHP-ohjelmoinnista kiinnostuneet käyttäjät saavat varmasti MODX-järjestelmästä irti jotain uutta sekä järjestelmän joka pystyy toteuttamaan käyttäjän omat toiveet. Drupal on järjestelmänä varmasti lähempänä MODX-järjestelmää kuin esimerkiksi edellä mainitut Joomla! tai WordPress. Sen käyttö voi olla aluksi hankalaa, mutta sen suosio ei voi kuitenkaan perustua järjestelmää, joka ei palvelisi sivuston tekijää. Drupalin kanssa työskentelevä kannattaa myös tutustua PHP-ohjelmointikieleen.

Yhteisöllisyys tuntuu olevan avain-asemassa eri järjestelmien käytössä. Selkeästi huomaa, että jokaisella järjestelmällä on hyvät kotisivut, jonka keskustelufoorumeille keskustelu järjestelmästä käy kiivaana. Eri komponenttien asennuksista ja käytöstä löytyy rutkasti informaatiota ja vinkkejä. MODX kalpenee vielä monille muille suosituille julkaisujärjestelmille komponenttien määrästä. Tämä oli myös yksi syy yhteystietomoduulin tekkoon, koska tämänlaista moduulia tai PHP-palasta ei ole ainakaan virallisesti missään Internetissä. Esimerkiksi Joomla!lle yhteystietomoduuli on valmiiksi olemassa jo oletuksena Joomla!n asennuksen jälkeen.

Yhteystietomoduuli on varmasti hyödyllinen osa Internet-sivuille, jota Paaroninmäen Moukari tulee jatkossa tekemään. Moduuli vaatii vielä hieman testausta ja esimerkiksi tietoturvaan liittyviin asioihin ei ole paneuduttu tässä opinnäytetyössä. Joka tapauksessa he voivat hyödyntää sitä tämän hetkisiin projekteihin ja päivittää yhteystietomoduulia sivujen tarpeen mukaan. Kuitenkin moduuli vaatii siihen perehtymistä ennen kuin sen voi sisällyttää uuteen sivustoon. Sen käyttöönotto on kuitenkin tehty mahdollisimman helpoksi.

Alusta alkaen olisi pitänyt tehdä laajempi suunnitelma itse toteutuksesta ja kohdistaa työ johonkin tiettyyn alueeseen sekä unohtaa pikkuseikat ja keskittyä kokonaisuuteen. Varsinkin tietokannat olisi pitänyt suunnitella alussa kunnolla ja vasta sitten lähteä työstämään niitä. Nyt rönsyilin välillä aiheesta ja keskityin liikaa yksityiskohtiin, enkä kokonaiskuvaan. Itselläni ei ole kuitenkaan hirveästi ole kokemusta PHP / MySQL ohjelmoinnista ja apua oli vaikea saada.

Moduulin teko osoittautui vaikeammaksi kuin odotin, mutta erittäin mielekkääksi ja opettavaiseksi kokemukseksi. Olen päässyt perehtymään uuteen MODX-järjestelmään ja myös mielenkiintoisten ohjelmointi tehtävien pariin. Itse MODX-järjestelmän ominaisuuksien perehtyminen jäi työssä hieman toisarvoiseksi, koska itse kehittämisprojekti vei suurimman osan ajan työn teosta. Nyt opinnäytetyössä on hieman tutustuttu järjestelmään, mutta syväluotoisempi perehtyminen olisi vaatinut yhteystietomoduulin ominaisuuksien karsimista. Toisaalta myös omat taitoni ovat vielä MODX:n suhteen kehittymässä, enkä olisi ehtinyt testata kaikkia uusia MODX:n tarjoamia mahdollisuuksia.

LÄHTEET

- About Drupal. About. drupal.org. Viitattu 10.11.2011. <http://drupal.org/about>
- Airaksinen, T. & Vilkka, H. 2003. Toiminnallinen opinnäytetyö. Helsinki. Tammi.
- Antano, J. 2011. MODx Web Development. Birmingham. PACKT publishing.
- Asleson, R. & Schutta, N. 2007. Ajax – Tehokas hallinta. Helsinki. Readme.fi.
- Drupal Suomi. Drupal.fi. Viitattu 10.11.2011. <http://drupal.fi/fi/drupal-suomi>
- Extend. wordpress.org. Viitattu 10.11.2011. <http://wordpress.org/extend/>
- Gilmore, W. 2005. PHP & MySQL – Tehokas hallinta. Helsinki. Readme.fi.
- Heinisuo, R. 2004. PHP ja MySQL Tietokantapohjaiset verkkopalvelut. Helsinki. Talemum Media Oy.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. 2008. Tutki ja kirjoita. Helsinki. Tammi.
- Kolehmainen, K. 2006. PHP & MySQL : Teoriasta käytäntöön. Helsinki. Readme.fi.
- Mikä on Joomla?. 2011. Joomla.fi. Viitattu 28.10.2011. <http://www.joomla.fi/mika-on-joomla>
- Most Popular CMS. 2011. PHP-developer.org. Viitattu 10.11.2011. <http://www.php-developer.org/most-popular-cms-content-management-system/>
- Negrino, T. & Smith, D. 2007. JavaScript – Tehokas hallinta. Helsinki. Readme.fi.
- Rantala, A. 2002. PHP : Web-ohjelmoinnin peruskirja. Jyväskylä. Docendo.
- Ray, B. 2011. MODX : The Official Guide. Dallas. MODX Press.
- Suomi, S. 2010. WordPress 3.0 sisällönhallintajärjestelmänä asiakaskäyttöön: case Road Dream. Tampereen ammattikorkeakoulu. Tietojenkäsittelyn koulutusohjelma. Opinnäytetyö.

Viimeisin julkaisu. 2011. Releases. fi.wordpress.org. Viitattu 10.11.2011.
<http://fi.wordpress.org/releases/>

Zeldman, J. 2003. Designing with web standards. New Riders.

index.php

```
<!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">
<head>
<link href="/assets/modules/yhteystietomoduuli/css/yhteystieto.css"
rel="stylesheet" type="text/css" />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Yhteystietomoduuli</title>
</head>
<body>

<h2>Yhteystietomoduuli</h2>

<!-- linkit muihin sivuihin -->
<a href="/assets/modules/yhteystietomoduuli/nayta.php" style="text-decoration: underline"><font color="#0000CC">N&auml;yt&auml;yhteystiedot</font></a> |
<a href="/assets/modules/yhteystietomoduuli/lisaa/lisaa.php" style="text-decoration: underline"><font color="#0000CC">Lis&auml;&auml;</font></a> |
<a href="/assets/modules/yhteystietomoduuli/muokkaa/muokkaa.php" style="text-decoration: underline"><font color="#0000CC">Muokkaa</font></a> |
<a href="/assets/modules/yhteystietomoduuli/poista/poista.php" style="text-decoration: underline"><font color="#0000CC">Poista</font></a> |
<a href="/assets/modules/yhteystietomoduuli/hallinta/hallinta.php" style="text-decoration: underline"><font color="#0000CC">Hallinta</font></a> |
<hr/>

<h3>Tervetuloa yhteystietomoduuliin! T&auml;&auml;ll&auml; voit lis&auml;t&auml;, muokata ja poistaa yhteystietoja.</h3>

</br>
</br>

</body>
</html>
```

yhteystieto.css

```
@charset "utf-8";

body, span, div, td, p, input, textarea, li, select, option {
    font-family: Arial, Helvetica, sans-serif;
    color: #000;
    font-size: 0.9em;
}

h2 {
    font-size: 1.7em;
    color: #000;
    font-family: Arial, Helvetica, sans-serif;
}

h3 {
    font-size: 1.2em;
    color: #000;
    font-family: Arial, Helvetica, sans-serif;
}
```

nayta.php

```
<h3>Kaikki yhteystiedot:</h3>

<?php

//tietokanta asetukset
$host=""; // Host nimi
$username=""; // Mysql käyttäjätunnus
$password=""; // Salasana
$db_name=""; // Tietokannan nimi
$table_name="yhteystieto"; // Taulun nimi

// yhteys tietokantaan
mysql_connect("$host", "$username", "$password")or die("Yhteys ei onnistunut!");
mysql_select_db("$db_name")or die("Virhe tietokanta yhteyden luomisessa!");

// valitaan kaikki taulusta yhteystieto, alueen ja paikkakunnan mukaan nousevassa järjestyksessä
$sql="SELECT * FROM $table_name ORDER BY alue, paikkakunta ASC";
$result=mysql_query($sql);

$count=mysql_num_rows($result);

?>

<form name="form1" method="post" action="">
<table width="30%" border="0" cellpadding="4" cellspacing="2" bgcolor="#CCCCCC">
<tr>
<td align="center" bgcolor="#FFFFFF"><strong>ID</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Alue</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Paikkakunta</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Yritys kategoria</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Yrityksen nimi</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Osoite</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Postinumero</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Postitoimipaikka</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Puhelinnumero</strong></td>
```

```

<td align="center" bgcolor="#FFFFFF"><strong>Fax-
numero</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>S&#228;hk&#246;posti</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>WWW-
osoite</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Tuoteryhm&#228;
Aseet</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Tuoteryhm&#228;
Varusteet</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Tuoteryhm&#228;
Asusteet</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Tuoteryhm&#228;
Muut</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Julkinen</strong></td>
</tr>
<?php
while($rows=mysql_fetch_array($result)){
?>

<!--haetaan tiedot tietokantataulusta taulukkoon -->
<tr>
<td bgcolor="#FFFFFF"><? echo $rows['id']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['alue']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['paikkakunta']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['yritys_kategoria']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['yritys_nimi']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['osoite']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['postinnumero']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['postitoimipaikka']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['puhelinnumero']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['fax']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['email']; ?></td>
<td bgcolor="#FFFFFF"><? echo $rows['www']; ?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['tuoteryhma'];
?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['tuoteryhma2'];
?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['tuoteryhma3'];
?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['tuoteryhma4'];
?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['julkinen'];
?></td>
</tr>

<?php
}
?>

```

</table>

</form>

</body>

</html>

lisaa.php

```
<title>Yhteystietomoduuli</title>

<!-- JavaScript tarkistus lisää nimi kohdalle -->
<script type="text/javascript">
function validateForm()
{
var x=document.forms["lisaaForm"]["yritys_nimi"].value;
if (x==null || x=="")
{
alert("Sinulta puuttuu nimi yhteystiedolle!");
return false;
}
}
</script>

</head>
<body>

<h2>Yhteystietomoduuli</h2>

<!-- linkit muihin sivuihin -->
<a href="/assets/modules/yhteystietomoduuli/nayta.php" style="text-decoration: underline"><font color="#0000CC">N&auml;yt&auml;yhteystiedot</font></a> |
<a href="/assets/modules/yhteystietomoduuli/lisaa/lisaa.php" style="text-decoration: underline"><font color="#0000CC">Lis&auml;&auml;</font></a> |
<a href="/assets/modules/yhteystietomoduuli/muokkaa/muokkaa.php" style="text-decoration: underline"><font color="#0000CC">Muokkaa</font></a> |
<a href="/assets/modules/yhteystietomoduuli/poista/poista.php" style="text-decoration: underline"><font color="#0000CC">Poista</font></a> |
<a href="/assets/modules/yhteystietomoduuli/hallinta/hallinta.php" style="text-decoration: underline"><font color="#0000CC">Hallinta</font></a> |
<hr/>
```

```
<h3>Lis&#228;&#228; yhteystieto: </h3>

<form          name="lisaaForm"          ac-
tion="/assets/modules/yhteystietomoduuli/lisaa/lisaa2.php"  onsub-
mit="return validateForm()" method="post">

<table width="500" border="0" cellspacing="1" cellpadding="1">
<tr><td>

<?php

//tietokanta asetukset
$host=""; // Host nimi
$username=""; // Mysql käyttäjätunnus
$password=""; // Salasana
$db_name=""; // Tietokannan nimi
$table_name="kategoria"; // Taulun nimi

// yhteys tietokantaan
mysql_connect("$host", "$username", "$password")or die("Yhteys ei on-
nistunut!");
mysql_select_db("$db_name")or die("Virhe tietokanta yhteyden luomises-
sa!");

// valitaan taulusta "kategoria" yrityskategoriat nousevassa järjestyksessä
$res = mysql_query("SELECT * FROM $table_name ORDER BY
yritys_kategoria ASC")
or die("Virhe yhteydess&#228;; " . mysql_query());
echo '<label>Yritys kategoria: </label> </td>';
echo '<td><select id="arvo" name="arvo">';

while ($row = mysql_fetch_assoc($res)) {
$va = $row['yritys_kategoria'];
echo "<option value='$va'$va</option>";
}
echo '</select>';
?>

&nbsp;
<a          href="/assets/modules/yhteystietomoduuli/hallinta/hallinta.php"
style="text-decoration: underline"><font color="#0000CC">
Lis&#228;&#228; uusi yritys kategoria</font></a></td>
</tr>

<tr>
<td>Yrityksen nimi:</td>
<td><input type="text" name="yritys_nimi" /></td>
</tr>
```



```
Lis&#228;&#228; uusi paikkakunta</font></a></td>
</tr>

<tr>
<td>Alue:</td>
<td>
<select name="alue">
<option value="Etel&auml;-Suomi">Etel&auml;-Suomi</option>
<option value="It&auml;-Suomi">It&auml;-Suomi</option>
<option value="Lansi-Suomi ja Ahvenanmaa">L&auml;nsi-Suomi ja Ah-
venanmaa</option>
<option value="Pohjois-Suomi">Pohjois-Suomi</option>
</select>
</td>
</tr>

<tr>
<td>Puhelinnumero (esim.09-5555050):</td>
<td><input type="text" name="puhelinnumero" /></td>
</tr>

<tr>
<td>Fax-numero (esim.09-5555055):</td>
<td><input type="text" name="fax" /></td>
</tr>

<tr>
<td>Email:</td>
<td><input type="text" name="email" /></td>
</tr>

<tr>
<td>WWW-osoite:</td>
<td><input type="text" name="www" value="http://" /></td>
</tr>

<tr>
<td>Tuoteryhm&#228; Aseet</td>
<td>
<input type="hidden" name="tuoteryhma" value="" />
<input type="checkbox" name="tuoteryhma" value="X" />
</td>
</tr>

<tr>
<td>Tuoteryhm&#228; Varusteet</td>
<td>
<input type="hidden" name="tuoteryhma2" value="" />
<input type="checkbox" name="tuoteryhma2" value="X" />
</td>
</tr>

<tr>
<td>Tuoteryhm&#228; Asusteet</td>
<td>
<input type="hidden" name="tuoteryhma3" value="" />
```

```

<input type="checkbox" name="tuoteryhma3" value="X" />
</td>
</tr>
<tr>
    <td>Tuoteryhm&#228; Muut</td>
    <td>
        <input type="hidden" name="tuoteryhma4" value="" />
        <input type="checkbox" name="tuoteryhma4" value="X" />
    </td>
</tr>
<tr>
    <td>Julkinen:</td>
    <td>
        <input name="julkinen" type="radio" value="X" checked/>
        Kyll&#228;;
        <input name="julkinen" type="radio" value="" /> Ei
    </td>
</tr>
<tr><td><input type="submit" value="L&#228;het&#228;"/></td></tr>
</table>
</form>

</body>
</html>

```

lisaa2.php

```
<?php

// yhteys tietokantaan
$con = mysql_connect("", "", "");
if (!$con)
{
    die('Yhteys ei onnistunut: ' . mysql_error());
}

mysql_select_db("", $con);

// lisää "yhteystieto" tauluun uudet tiedot
$sql="INSERT INTO yhteystieto (id, yritys_kategoria, yritys_nimi, osoite, postinumero, postitoimipaikka, paikkakunta, alue, puhelinnumero, fax, email, www, tuoteryhma, tuoteryhma2, tuoteryhma3, tuoteryhma4, julkinen)
VALUES
($_POST[id],$_POST[arvo],$_POST[yritys_nimi],$_POST[osoite],$_POST[postinumero],$_POST[postitoimipaikka],
$_POST[arvo1],$_POST[alue],$_POST[puhelinnumero],$_POST[fax],
$_POST[email],$_POST[www],$_POST[tuoteryhma],
$_POST[tuoteryhma2],$_POST[tuoteryhma3], $_POST[tuoteryhma4],
$_POST[julkinen])";

if (!mysql_query($sql,$con))
{
    die('Virhe yhteydess&#228;;: ' . mysql_error());
}
echo "Yhteystieto '$_POST[yritys_nimi]' on tallennettu onnistuneesti!";

//sulje yhteys
mysql_close($con)
?>

</body>
</html>
```

muokkaa.php

```
<h3>Valitse muokattava yhteystieto:</h3>

<?

// varmista utf-8 merkintä
mysql_query("SET NAMES 'utf8'", $conn);

// yhteys tietokantaan
$yhteys = mysql_connect("", "", "");

if (!$yhteys)
{
    echo '<p>Yhteys ei onnistunut!</p>';
    exit();
}

if (!mysql_select_db("")) {
    exit('<p>Yhteys ei onnistunut.</p>');
}

$rad      =      "<form      method      =      'post'      action      =
'/assets/modules/yhteystietomoduuli/muokkaa/muokkaa2.php'><input type
= submit name = 'x' value = 'muokkaa'>";

// valitse "yhteystieto" taulusta alueen ja paikkakunnan mukaan nousevas-
sa järjestyksessä
$result = @mysql_query('SELECT * FROM yhteystieto ORDER BY
alue, paikkakunta ASC');
?>

<!-- taulukon tiedot -->
<table width="55%" border="1" cellspacing="1" cellpadding="1">
<tr>
<td align="center" bgcolor="#FFFFFF"><strong>ID</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Alue</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Paikkakunta</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Yritys katego-
ria</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Yrityksen ni-
mi</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Osoite</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Postinumero</strong></td>
<td align="center" bgcolor="#FFFFFF"><strong>Postitoimipaikka</strong></td>
</tr>
```

```
<?php
while ( $row = mysql_fetch_array( $result ) )
{
    print "<TR><TD>$row[0] </td><TD>$row[7] </td><TD>$row[6]
</td><TD>$row[1] </td><TD>$row[2] </td><TD>$row[3]
</td><TD>$row[4] </td>";
    print "<TD>$row[5] </td><TD>$rad <input type = hidden name = 'y'
value = '$row[0]'></form></td></TR>";
}
print "</TABLE>\n";
?>

</body>
</html>
```

muokkaa2.php

```
<h3>Muokkaa yhteystietoa:</h3>

<form action="<? echo $softa; ?>" method="post">
<table width="30%" border="0" cellspacing="1" cellpadding="1">

<?php
$softa = $_SERVER['PHP_SELF'];

// yhteys MySQL-palvelimelle
$yhteys = mysql_connect("", "", "");

// yhteyden varmistaminen
if (!$yhteys)
{
echo '<p>Yhteys ei onnistu!</p>';
exit();
}

// valitse tietokanta
if (!mysql_select_db("")) {
exit('<p>Tietokannan valitseminen ei onnistu!</p>');
}

if (isset($_POST['y']))
{ $my = $_POST['y'];
settype($my, "integer");
}
// "yhteystieto" taulun id
$sql = "SELECT * FROM yhteystieto WHERE id = $my";

$result = mysql_query($sql);
while ($row = mysql_fetch_array( $result ))
{
?>

<tr><td><label>ID: </td>
<td><? echo $row[0]; ?>
</label><br/> </td>
</tr>

<input type = "hidden" name = "1" value = "<? echo $row[0]; ?>" >

<tr><td><label>Yritys kategoria: </td>
<td><? echo $row[1]; ?>
</label><br/></td>
</tr>
```

```
<tr><td><label>Yrityksen nimi: </td>
<td><input type = "text" name = "3" value = "<? echo $row[2]; ?>"
></label></td>
</tr>
```

```
<tr><td><label>Osoite: </td>
<td><input type = "text" name = "4" value = "<? echo $row[3]; ?>"
></label></td>
</tr>
```

```
<tr><td><label>Postinumero: </td>
<td><input type = "text" name = "5" value = "<? echo $row[4]; ?>"
></label></td>
</tr>
```

```
<tr><td><label>Postitoimipaikka: </label></td>
<td><input type = "text" name = "6" value = "<? echo $row[5]; ?>"
></td>
</tr>
```

```
<tr><td><label>Paikkakunta: </td>
<td><? echo $row[6]; ?>
</label></td>
</tr>
```

```
<tr><td>Alue: </td>
<td><select name="alue" />
<option value="Etela-Suomi" <?php if ($row['alue'] == "Etela-Suomi"){
echo 'selected="selected"';}?> /> Etel&auml;-Suomi
</option>
<option value="Lansi-Suomi ja Ahvenanmaa" <?php if ($row['alue'] ==
"Lansi-Suomi ja Ahvenanmaa"){ echo ' selected="selected"';}?> />
L&auml;nsi-Suomi ja Ahvenanmaa
</option>
<option value="Ita-Suomi" <?php if ($row['alue'] == "Ita-Suomi"){ echo
'selected="selected"';}?> /> It&auml;-Suomi</option>
<option value="Pohjois-Suomi" <?php if ($row['alue'] == "Pohjois-
Suomi"){ echo 'selected="selected"';}?> /> Pohjois-Suomi</option>
</select></td>
</tr>
```

```
<tr><td><label>Puhelinnumero (esim.09-5555050): </td>
<td><input type = "text" name = "9" value = "<? echo $row[8]; ?>"
></label></td>
</tr>
```

```
<tr><td><label>Fax-numero (esim.09-5555055): </td>
<td><input type = "text" name = "10" value = "<? echo $row[9]; ?>"
></label></td></tr>
```

```
<tr><td><label>S&#228;hk&#246;posti: </td>
<td><input type = "text" name = "11" value = "<? echo $row[10]; ?>"
></label></td>
</tr>

<tr><td><label>WWW-osoite: </td>
<td><input type = "text" name = "12" value = "<? echo $row[11]; ?>"
></label></td>
</tr>

<tr><td>Tuoteryhm&#228;; Asetet </td>
<td><input type="checkbox" name="13" value="X" <? if($row[12] ==
'X'){ echo 'checked="checked"';} ?> /> </td>
</tr>

<tr><td>Tuoteryhm&#228;; Varusteet </td>
<td><input type="checkbox" name="14" value="X" <? if($row[13] ==
'X'){ echo 'checked="checked"';} ?> /> </td></tr>

<tr><td>Tuoteryhm&#228;; Asusteet </td>
<td><input type="checkbox" name="15" value="X" <? if($row[14] ==
'X'){ echo 'checked="checked"';} ?> /> </td>
</tr>

<tr><td>Tuoteryhm&#228;; Muut </td>
<td><input type="checkbox" name="16" value="X" <? if($row[15] ==
'X'){ echo 'checked="checked"';} ?> /></td>
</tr>

<tr><td><label>Julkinen: </td>
<td><input type = "radio" name = "17" value = "X" <?= ($row[16] == 'X')
? 'checked' : " ?> /> Kyll&#228;; </label>
<label><input name="17" type="radio" value="" <?= ($row[16] == ") ?
'checked' : " ?> /> Ei </label> </td>
</tr>

<tr><td><label><input type="submit" value="Tallenna muutokset"
/></label> </td>
</tr>
</table>

</form>

<?
}

if (isset($_POST['1']))
{ $m1 = $_POST['1']; settype($m1, "integer"); }
if (isset($_POST['2']))
```

```
$m2 = $_POST['2'];
if (isset($_POST['3']))
$m3 = $_POST['3'];
if (isset($_POST['4']))
$m4 = $_POST['4'];
if (isset($_POST['5']))
$m5 = $_POST['5'];
if (isset($_POST['6']))
$m6 = $_POST['6'];
if (isset($_POST['7']))
$m7 = $_POST['7'];
if (isset($_POST['alue']))
$m8 = $_POST['alue'];
if (isset($_POST['9']))
$m9 = $_POST['9'];
if (isset($_POST['10']))
$m10 = $_POST['10'];
if (isset($_POST['11']))
$m11 = $_POST['11'];
if (isset($_POST['12']))
$m12 = $_POST['12'];
if (isset($_POST['13']))
$m13 = $_POST['13'];
if (isset($_POST['14']))
$m14 = $_POST['14'];
if (isset($_POST['15']))
$m15 = $_POST['15'];
if (isset($_POST['16']))
$m16 = $_POST['16'];
if (isset($_POST['17']))
{ $m17 = $_POST['17'];}

if (isset($_POST['1']))
{
$ssql = "UPDATE yhteystieto SET
yritys_kategoria = '$m2', yritys_nimi = '$m3', osoite = '$m4', postinumero
= '$m5', postitoimipaikka = '$m6', paikkakunta = '$m7', alue = '$m8', pu-
helinumero = '$m9', fax = '$m10', email = '$m11', www = '$m12', tuote-
ryhma = '$m13', tuoteryhma2 = '$m14', tuoteryhma3 = '$m15', tuoteryh-
ma4 = '$m16', julkinen = '$m17' WHERE id = $m1
";

if (mysql_query($ssql)) {
echo 'Tietojen tallennus onnistui!';
}
else
{
echo 'Yhteystietojen muokkaus ei onnistunut: ';
mysql_error();
```

}

}

?>

</body>

</html>

poista.php

```
<title>Yhteystietomoduuli</title>

<!-- JavaScript tarkistus ennen yhteystietojen poistamista -->
<script language=javascript>
function verify(){
    msg = "Haluatko poistaa valitun / valitut yhteystiedot?";
    return confirm(msg);
}
</script>

</head>
<body>

<h2>Yhteystietomoduuli</h2>

<!-- linkit muihin sivuihin -->
<a href="/assets/modules/yhteystietomoduuli/nayta.php" style="text-decoration: underline"><font color="#0000CC">N&auml;yt&auml;yhteystiedot</font></a> |
<a href="/assets/modules/yhteystietomoduuli/lisaa/lisaa.php" style="text-decoration: underline"><font color="#0000CC">Lis&auml;&auml;</font></a> |
<a href="/assets/modules/yhteystietomoduuli/muokkaa/muokkaa.php" style="text-decoration: underline"><font color="#0000CC">Muokkaa</font></a> |
<a href="/assets/modules/yhteystietomoduuli/poista/poista.php" style="text-decoration: underline"><font color="#0000CC">Poista</font></a> |
<a href="/assets/modules/yhteystietomoduuli/hallinta/hallinta.php" style="text-decoration: underline"><font color="#0000CC">Hallinta</font></a> |
<hr/>
<h3>Poista yhteystietoja. Voit poistaa yhden tai useamman yhteystiedon samanaikaisesti.</h3>

<?php

// tietokanta asetukset
$host=""; // Host nimi
$username=""; // Mysql käyttäjätunnus
$password=""; // Salasana
$db_name=""; // Tietokannan nimi
$table_name="yhteystieto"; // Taulun nimi

// yhteys tietokantaan
mysql_connect("$host", "$username", "$password")or die("Yhteys ei onnistunut!");
```

```
mysql_select_db("$db_name")or die("Virhe tietokanta yhteyden luomisessa!");
```

```
// valitse "yhteystieto" taulusta alueen ja paikkakunnan mukaan nousevas-  
sa järjestyksessä  
$result = mysql_query("SELECT * FROM yhteystieto ORDER BY alue,  
paikkakunta ASC");  
?>
```

```
<!-- lomakkeen ja taulukon tiedot alkavat -->  
<form name="form1" method="post" action="/assets/modules/yhteystietomoduuli/poista/poista2.php" onSub-  
mit="return verify()">  
<table width="65%" border="1" cellspacing="1" cellpadding="1">  
<tr>  
<td align="center" bgcolor="#FFFFFF"><strong></strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Alue</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Paikkakunta</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Yritys katego-  
ria</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Yrityksen ni-  
mi</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Osoite</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Postinumero</strong></td>  
<td align="center" bgcolor="#FFFFFF"><strong>Postitoimipaikka</strong></td>  
</tr>
```

```
<?php  
while($row = mysql_fetch_array($result)) {  
    echo '<tr>  
        <td><input type="checkbox" name="poista[]" value="" .  
$row['id'] . "'></td>;  
        echo '<td>' . $row['alue'] . '</td>;  
        echo '<td>' . $row['paikkakunta'] . '</td>;  
        echo '<td>' . $row['yritys_kategoria'] . '</td>;  
        echo '<td>' . $row['yritys_nimi'] . '</td>;  
        echo '<td>' . $row['osoite'] . '</td>;  
        echo '<td>' . $row['postinumero'] . '</td>;  
        echo '<td>' . $row['postitoimipaikka'] . '</td></tr>; }'
```

```
// sulje yhteys  
mysql_close($con);  
?>
```

```
</table>
<br/>
<input name="delete" type="submit" id="delete" value="Poista">
</form>

</body>
</html>
```

poista2.php

```
<?php

$host=""; // Host nimi
$username=""; // Mysql käyttäjätunnus
$password=""; // Salasana
$db_name=""; // Tietokannan nimi
$table_name="yhteystieto"; // Taulun nimi

// yhteys tietokantaan
mysql_connect("$host", "$username", "$password")or die("Yhteys ei onnistunut!");
mysql_select_db("$db_name")or die("Virhe tietokanta yhteyden luomisessa!");

$poista_yhteystieto = $_POST['poista'];

// poista valitut yhteystiedot "yhteystieto" taulusta
foreach($poista_yhteystieto as $id) {
    mysql_query("DELETE FROM yhteystieto WHERE id='$id'");
}

if (!mysql_query)
{
    die('Virhe yhteydessä: ' . mysql_error());
}
echo "Yhteystieto / yhteystiedot on poistettu onnistuneesti!";

//sulje yhteys
mysql_close()

?>

</body>
</html>
```

Sivupohja

```
{{head_osa}}

<body>

<div id="kokosivu">

<div id="logo">
<a href="http://demo6.moukaridemo.com/">
</a>
</div>

<div id="linkit">
<hr />
[[Wayfinder? startId=`0` &firstClass=`first` &hereClass=`active` ]]
<hr />
</div>

<div id="sisalto">

[*content*]

</div>

</div>

{{footer}}

</body>
</html>
```

css_tiedosto.css

```
* {
margin: 0;
padding: 0;
}

#kokosivu {
margin: 0 auto;
width: 900px;
height: 800px;
margin-top: 20px;
background-color: #CCC;
border: 2px solid #000;
}

#logo {
width: 900px;
height: 91px;
}

#linkit {
width: 900px;
height: 31px;
line-height: 1.6em;
}

#linkit li {
margin-left: 5px;
display: inline;
}

#linkit ul {
display: inline;
}

#sisalto {
width: 860px;
height: 650px;
margin: 1em 1em 1em 1em;
background-color: #CCC;
}

div.map {
position: absolute;
top: 0;
left: 0;
display: none;
}
```

```
#maps img{
    border: none;
}

#map_top {
    display: block;
    z-index: 1100;
}

#maps {
    margin: 4em 2em 4em 2em;
    padding:0;
    position:relative;
    float: left;
    background-image:
url(/assets/templates/yhteystietomoduuli/kartta-laanit-suomi.png);
    background-repeat: no-repeat;
    width:300px;
    height:530px;
}

#oikea {
    margin:0;
    padding:0;
    position:relative;
    background-repeat: no-repeat;
    width:300px;
    height:500px;
}

#footer {
    margin: 0 auto;
    margin-top: 5px;
    width: 532px;
}

body {
    background-color: #CCC;
    color: #ffffff;
    font-family: Arial, Helvetica, sans-serif;
}

p {
    font-size: 1.1em;
    color: #000;
}
```

```
h1 {  
font-size: 1.2em;  
color: #000;  
}
```

```
h2 {  
font-size: 1em;  
color: #000;  
}
```

```
h3 {  
font-size: 0.8em;  
color: #000;  
}
```

```
a, a:visited, a:active {  
color: #333;  
text-decoration: none;  
}
```

```
a:hover {  
text-decoration: none;  
}
```

[[haku]]

```
<?php
$yhteys = mysql_connect("", "");
if (!$yhteys)
{
    echo '<p>Yhteys ei onnistunut!</p>';
    exit();
}

if (!mysql_select_db("")) {
    exit('<p>Yhteys ei onnistunut!</p>');
}

$softa = $_SERVER['PHP_SELF'];

echo '<form action="[~15~]" method="post">';

// haetaan aloitteet listaan
$sql = "select id, yritys_nimi from yhteystieto order by yritys_nimi ASC";
$result = mysql_query($sql);
print "<p>Valitse listalta yhteystieto:</p> <br/>";
print "<select name = '1'>";
while ( $row = mysql_fetch_array( $result ) )
{
    print "<option value = '$row[0]' > $row[1]";
}
print "</select>\n";

echo '<input type = submit value = "Hae tiedot">';
echo '<br>';
echo '</form>';
?>
```

[[haku_toiminto]]

```
<?php
$con = mysql_connect("", "", "");
if (!$con)
{
    die('Virhe yhteydessä ' . mysql_error());
}

mysql_select_db("", $con);

$term = $_POST['haku'];

$result = mysql_query("select * from yhteystieto where yritys_nimi like
'%" . $term . "%' AND julkinen = 'X'");

while ($row = mysql_fetch_array($result)){
    echo 'ID: ' . $row['id'];
    echo '<br/> Yritys nimi: ' . $row['yritys_nimi'];
    echo '<br/><br/>';
}

mysql_close($con);
?>
```

Sivupohja_kartta_sivulle

```
{{head_osa2}}

<body>

<div id="kokosivu">

<div id="logo">
<a href="http://demo6.moukaridemo.com/">
</a>
</div>

<div id="linkit">
<hr />
[[Wayfinder? startId=`0` &firstClass=`first` &hereClass=`active` ]]
<hr />
</div>

<div id="sisalto">

<div id="maps" align="left">
<!--style="top:-100px; left:370px; z-index:300;!-->

<div class="map" id="map_lappi"></div>

<div class="map" id="map_oulu"></div>

<div class="map" id="map_itasuomi"></div>

<div class="map" id="map_lansisuomi"></div>

<div class="map" id="map_etelasuomi"></div>

<div class="map" id="map_ahvenanmaa"></div>
```

```
<div class="map" id="map_top"></div>
```

```
<map name="laanimap">  
<area href="[~11~]" onmouseover="selectArea('ahvenanmaa');" on-  
mouseout="selectArea('ahvenanmaa');" shape="poly"  
coords="34,421,19,422,10,417,0,407,14,397,24,409" alt="Ahvenanmaa"  
>
```

```
<area href="[~13~]" onmouseover="selectArea('lappi');" on-  
mouseout="selectArea('lappi');" shape="poly"  
coords="36,36,58,61,70,65,86,87,84,102,89,111,84,119,91,142,88,156,95,  
175,114,193,121,182,136,183,152,186,160,178,176,183,177,178,179,174,  
178,161,182,154,198,157,188,133,201,106,184,80,171,73,172,49,177,36,1  
81,24,174,12,153,0,123,17,122,47,106,59,93,50,82,57,62,38,46,24,46,37"  
alt="Lappi" />
```

```
<area shape="poly"  
coords="120,209,118,218,106,218,86,245,110,279,118,270,133,282,140,2  
77,134,270,147,256,162,260,176,271,185,263,203,262,222,261,225,245,2  
13,235,212,222,204,220,203,188,210,180,200,159,183,153,178,164,181,1  
77,178,185,160,177,152,189,125,183,114,192" href="[~13~]" alt="Oulu"  
onmouseover="selectArea('oulu');" onmouseout="selectArea('oulu');" />
```

```
<area href="[~12~]" onmouseover="selectArea('itasuomi');" on-  
mouseout="selectArea('itasuomi');" shape="poly"  
coords="154,364,159,370,154,375,152,370,142,370,141,354,138,333,147,  
324,133,302,142,294,134,285,141,279,135,271,149,255,169,264,176,270,  
185,262,202,261,208,263,219,262,222,274,236,286,246,301,244,318,238,  
329,232,338,227,345,216,345,202,356,197,360,186,358,178,365"  
alt="Itä-Suomi" />
```

```
<area href="[~11~]" onmouseover="selectArea('lansisuomi');" on-  
mouseout="selectArea('lansisuomi');" shape="poly"  
coords="24,391,23,375,27,364,28,352,25,335,29,322,25,302,33,291,43,28  
2,53,282,66,270,84,246,99,265,113,280,119,270,132,281,139,296,132,304  
,141,311,146,325,135,334,140,354,129,350,115,357,99,366,83,376,68,381  
,68,388,80,392,81,402,75,408,73,413,59,416,49,406,34,400" alt="Länsi-  
Suomi" />
```

```
<area href="[~10~]" onmouseover="selectArea('etelasuomi');" on-  
mouseout="selectArea('etelasuomi');" shape="poly"  
coords="83,403,83,392,70,388,69,380,80,380,94,372,110,360,123,355,13  
4,350,140,355,141,369,151,370,155,373,159,372,157,365,180,366,187,35  
6,197,361,217,345,228,347,207,369,179,396,167,403,147,403,135,405,12  
5,406,106,415,102,409,90,419,68,423,58,418,77,410" alt="Etelä-Suomi"  
>
```

```
</map>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
{{footer}}
```

```
</body>
```

```
</html>
```

[[Haku_Etela]]

```
<?php
$host="";
$username="";
$password="";
$db_name="";
$tbl_name="yhteystieto"; // taulun nimi

mysql_connect("$host", "$username", "$password")or die("Ei onnistu tietokantayhteyden luominen");
mysql_select_db("$db_name")or die("Ei onnistu yhteyden luominen");

$result = mysql_query("SELECT * FROM $tbl_name WHERE julkinen='X' AND alue = 'Etelä-Suomi' ORDER BY paikkakunta ASC")
or die("Virhe yhteydess&#228;;: " . mysql_query());

while ($row = mysql_fetch_array($result)) {
echo $row['alue'] . " " . $row['paikkakunta'] . " " . $row['yrittys_nimi'];
echo "<br /> <br />";
}
?>
```

Ohjeet Yhteystietomoduuliin

Lataa ja purkaa yhteystietomoduuli.zip koneellesi. Tämän jälkeen ota yhteys muokattavan MODX-järjestelmän tiedostoihin FTP-yhteyden kautta. Mene /modx/assets/modules kansioon ja kopioi sinne yhteystietomoduulikansio.

Tämän jälkeen kirjaudu sisään MODX-järjestelmään. Valitse ”Moduulit” välilehti ja sieltä ”Moduulit”. Tee ”Uusi moduuli”. Anna moduulille nimi Yhteystietomoduuli (tai joku muu kuvaava nimi). Lisää lähdekoodiin koordinaattoritiedot:

```
include_once $modx-
>con-
fig['base_path'].'assets/modules/yhteystietomoduuli/index.p
hp';
```

Seuraavaksi ”Tallenna”. Tämän jälkeen suorita moduuli, menemällä moduulit välilehdelle ja paina oikean hiiren painalluksella yhteystietomoduulin vieressä olevaa kuvaketta. Kirjaudu ulos järjestelmästä ja sitten kirjaudu takaisin järjestelmään. Nyt Moduulit välilehdellä lukee Yhteystietomoduuli.

Tee tarvittavat tietokantayhteydet. Tässä versiossa kannassa on kolme taulua: yhteystieto, paikka ja kategoria.

Yhteystieto-taulu:

Tietokenttä	Tietotyyppi
ID-numero	SMALLINT
Yrityskategoria	VARCHAR
Yrityksen nimi	VARCHAR
Katuosoite	VARCHAR
Postinumero	VARCHAR
Postitoimipaikka	VARCHAR
Paikkakunta	VARCHAR
Alue	VARCHAR
Puhelinnumero	VARCHAR
Fax-numero	VARCHAR
Sähköposti	VARCHAR
WWW-osoite	VARCHAR
Tuoteryhmä	VARCHAR
Tuoteryhmä2	VARCHAR
Tuoteryhmä3	VARCHAR
Tuoteryhmä4	VARCHAR
Julkinen	VARCHAR

Kategoria-taulu:

Tietokenttä	Tietotyyppi
ID-numero	SMALLINT
Yrityskategoria	VARCHAR

Paikka-taulu:

Tietokenttä	Tietotyyppi
ID-numero	SMALLINT
Paikkakunta	VARCHAR

Tee itsellesi kaksi uutta sivupohjaa. Lisää sivupohjiin koodit, jotka on nimetty ”Sivupohja” ja ”Sivupohja_kartta_sivulle”. Talleta uudet sivupohjat. Lisää FTP-yhteyden kautta /assets/templates/ kansioon Sivut-kansio. Tarkista sivupohjien linkit oikeisiin tiedostoihin (JavaScript-linkitykset, css-tiedosto ja kuvat).

Tee MODX-järjestelmässä hakemistopuun mukaiset sivut. Määritä kaikille sivuille sivupohjaksi ”Sivupohja”, paitsi Kartta-Haku sivulle määritä sivupohjaksi ”Sivupohja_kartta_sivulle”.



- Tee Haku-sivulle linkki [!haku!].
- Tee Haku2-sivulle linkki {{haku}}.
- Tee Hakutulos-sivulle linkki [!haku_toiminto!].
- Tee Kartta_Etela-sivulle linkki [!Haku_Etela!].
- Tee sama myös Kartta_Lansi, Kartta_Ita ja Kartta_Pohjois sivuille ja linkitä jokaiselle sivulle linkki PHP-palaseen oman alueen mukaan.

Tee HTML palaset: footer, haku ja head-osa. Liitä niihin oikeat koodit nimen perusteella.

Tee PHP-palaset: haku, Haku_Etela, Haku_Ita, Haku_Lansi, Haku_Pohjois ja haku_toiminto. Liitä niihin oikeat koodit nimen perusteella.