

# MAINOSALAN VERKKOSIVUSTON ERITYIS- PIIRTEET JA NIIDEN TOTEUTTAMINEN

Heikki Ala-Nikkola

Opinnäytetyö  
Toukokuu 2013

Mediatekniikan koulutusohjelma  
Tekniikan ja liikenteen ala



Tekijä(t) ALA-NIKKOLA Heikki	Julkaisun laji Opinnäytetyö	Päivämäärä 06.05.2013
	Sivumäärä 39	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty ( X )
Työn nimi MAINOSALAN VERKKOSIVUSTON ERITYISPIIRTEET JA NIIDEN TOTEUTTAMINEN		
Koulutusohjelma Mediatekniikan koulutusohjelma		
Työn ohjaaja(t) MANNINEN, Pasi		
Toimeksiantaja(t) HAUTANEN, Arto - Into-Digital Oy		
<p>Tiivistelmä</p> <p>Opinnäytetyössä tutkittiin erityispiirteitä ja -toiminnallisuuksia, joita useimmat mainosalan verkkosivutoteutukset sisältävät. Toimeksiantajana opinnäytetyössä toimi Into-Digital Oy, jonka toteuttamaan verkkosivustohankkeen suunnitteluun ja toteutukseen opinnäytetyö pohjautui. Opinnäytetyössä käsiteltiin mainosalan sivustoja ominaisuustasolla ja tarkoituksena oli löytää näihin ominaisuuksiin kestäviä, luotettavia ja monipuolisia ratkaisuja.</p> <p>Opinnäytetyössä vertailtiin mainosalan verkkosivutoteutusten alustaksi soveltuvia sisällönhallintajärjestelmiä, joista tutkittiin tarkemmin mainosalan sivustojen vaatimuksiin eniten soveltuvaa WordPressiä sen erityisominaisuuksien osalta. Tarkasteltavia ominaisuuksia opinnäytetyössä olivat kieliversiointi, erikoisfonttien käyttö, responsiivinen suunnittelu ja teemojen käyttö. Näiden erityisominaisuuksien merkitystä tutkittiin mainosalan verkkototeutuksien osalta yleisesti.</p> <p>Opinnäytetyössä tutkittiin näitä tekniikoita verkkosivuhankkeen osalta käytännössä, toteuttamalla niitä julkiseen verkkosivustoon. Hanketta käsiteltiin suunnittelun, toteutusprosessin sekä tulosten osalta. Opinnäytetyön pohdintaosiossa vertailtiin teoriaosiossa tutkittujen erikoisominaisuuksien toteutusmahdollisuuksia hankkeessa käytettyihin toteutuksiin. Pohdintaosiossa käsiteltiin erityisesti hankkeessa käytettyjen ratkaisujen kannattavuutta niiden pitkäikäistä käyttöä ja jatkokehitysmahdollisuuksia ajatellen. Pohdintaosiossa analysoitiin myös tulevaisuudennäkymiä näiden ratkaisujen ja erikoisominaisuuksien osalta.</p> <p>Hankkeessa tutkitut ratkaisut osoittautuivat oikeiksi ratkaisuksi sivuston vaatimusten suhteen. Ratkaisujen todettiin olevan soveltuvia myös jatkossa toteutettavia hankkeita varten, joihin on suunniteltu samankaltaisia ratkaisuja. Hankkeen lopputuloksena syntynyt verkkosivusto julkaistiin marraskuussa 2012.</p>		
Avainsanat (asiasanat) Sisällönhallintajärjestelmä, Responsiivinen suunnittelu, Kieliversiointi		
Muut tiedot		

Author(s) ALA-NIKKOLA Heikki	Type of publication Bachelor's Thesis	Date 06.05.2013
	Pages 39	Language Finnish
		Permission for web publication ( X )
Title COMMON SPECIAL FEATURES OF MARKETING WEBSITES AND THEIR IMPLEMENTATIONS		
Degree Programme Media Engineering		
Tutor(s) MANNINEN, Pasi		
Assigned by HAUTANEN, Arto - Into-Digital Oy		
<p>Abstract</p> <p>The thesis examines the special features required by many marketing websites. The subject of the thesis was assigned by Into-Digital Oy. The thesis is based on the planning and designing of a website project set out by the company. The thesis examines websites at the level of features. The main goal of the thesis was to find sustainable solutions for the special features.</p> <p>The thesis also compared content management systems suitable as a platform for marketing websites' special features. Out of three candidates WordPress was chosen for further examination on how the special features could be implemented. The special features to be examined were multilingual content management for the content management system, using special fonts, responsive website design and packaging all this functionality as a theme. These features and their solutions were examined in general.</p> <p>The theories and solutions were applied to a website project and examining in a practice then implemented on a commercial website. The project was inspected in terms of planning, design, implementation and results. The results of the project and the solutions used were then compared to the examined theories and possible solutions. The summary specifically examined the viability of the chosen solutions by their longevity and further development aspects. The summary also dealt with possible future scenarios concerning the special features and the solutions used.</p> <p>The solutions examined turned out to be the correct solutions that matched the requirements of the website. The solutions have since been applied to many more projects with similar requirements. The project's result, the website was published in November 2012</p>		
Keywords Content Management System, Responsive design, Multilingual sites		
Miscellaneous		

## SISÄLTÖ

<b>1 TYÖN LÄHTÖKOHDAT</b> .....	<b>3</b>
1.1 Tausta .....	3
1.2 Toimeksiantaja.....	4
1.3 Tavoite.....	4
<b>2 SISÄLLÖNHALLINTAJÄRJESTELMÄT</b> .....	<b>5</b>
2.1 Yleistä .....	5
2.2 WordPress .....	5
2.3 Drupal .....	6
2.4 Joomla! .....	6
<b>3 MARKKINOINTISIVUSTOJEN ERIKOISOMINAISUUDET</b> .....	<b>8</b>
3.1 Yleistä .....	8
3.2 Kieliversiointi .....	8
3.2.1 Yleistä kieliversioinnista .....	8
3.2.2 Toteutustapa sisällönhallintajärjestelmässä .....	8
3.2.3 WordPressin kieliversiointimahdollisuudet .....	9
3.3 Erikoisfonttien käyttö.....	11
3.3.1 Yleistä erikoisfonteista .....	11
3.3.2 Toteutustavat.....	12
3.4 Teemat.....	13
3.4.1 Yleistä teemoista .....	13
3.4.2 Teeman rakenne .....	14
3.4.3 Sivupohjan rakenne.....	16
3.5 Responsiivinen suunnittelu .....	17
3.5.1 Yleistä responsiivisesta suunnittelusta .....	17
3.5.2 Tekniikat.....	18
<b>4 MARKKINOINTISIVUSTON TOTEUTUSHANKE</b> .....	<b>21</b>
4.1 Lähtötilanne.....	21
4.2 Tekninen suunnittelu .....	22
4.3. Sisällönhallintajärjestelmän valinta .....	23
4.3.1 Huomioonotettavat asiat sisällönhallinnan valinnassa .....	23
4.3.2 Valinta .....	25

4.4 Käyttöönotto .....	26
4.5 Erikoisominaisuuksien toteutus.....	27
4.6 Tulokset .....	34
5 POHDINTA.....	36
LÄHTEET.....	40

## KUVIOT

KUVIO 1. WordPress-sivun hallintanäkymän kielivalitsin.....	10
KUVIO 2. Esimerkki media query-tekniikalla porrastetusta responsiivisuudesta .....	19
KUVIO 3. Esimerkki fluid-taitosta .....	20
KUVIO 4. Multiple Content Blocks-lisäosan kaksi itsenäistä sisältöeditoria.....	29
KUVIO 5. Sisältösivun sivupohja .....	30
KUVIO 6. Väriteeman määrittäminen Custom Fieldin avulla.....	31
KUVIO 7. Etusivun sivupohjan rakenne .....	32
KUVIO 8. Responsiiviseksi optimoitu sisältösivu.....	33

# 1 TYÖN LÄHTÖKOHDAT

## 1.1 Tausta

Verkkosivustojen merkitys markkinointialalla on muuttunut viime vuosina merkittävästi. Vielä muutamia vuosia sitten verkkosivustot merkitsivät toteuttavalta taholta ostettavaa tuotetta, jonka päivittäminen tarkoitti HTML-koodiin kiinteästi sijoitettujen sisältöjen muokkaamista asiantuntevan henkilön toimesta. Myös nämä sivustot suunniteltiin etupäässä selainyhteensopivuuksien puitteissa. Ulkoasussa käytettiin monesti yksinkertaisia ratkaisuja, jotka toimivat hyvin kaikilla selaimilla ja grafiikan toteutus tapahtui pitkälti kuvina.

Nykyään verkkosivustot ovat luonteeltaan lähes palvelunomaisia. Ne ovat minimissään suuren palveluntarjoajan hyvin ylläpidetyssä webhotellissa sijaitsevia sisällönhallintajärjestelmän päällä toimivia intuitiivisia kehyksiä, joita lähes kuka tahansa tekstinkäsittelyohjelmaa käyttävä henkilö voi täydentää haluamallaan multim mediasisällöllä ilman merkittävää perehtymistä tekniikkaan. Sivustojen ulkoasut suunnitellaan sisällön mukaan venyviksi sekä katseltaviksi lähes millä tahansa verkkoyhteyteen kykenevällä laitteella. Yhä useampaan lisäosaan tai ohjelmistoon on saatavilla myös varten otettava ilmainen tai erittäin halpa kevyempi vaihtoehto, raskaan ja kattavan maksullisen sovelluksen sijaan.

Mikään internetistä ladattava, palvelimelle asennettava paketillinen tiedostoja ei kuitenkaan voi olla valmis sivustototeutus, joten ulkoasun suunnittelu sekä teknisen toteutuksen suunnittelu ovat yhä merkittäviä ja välttämättömiä vaiheita sivustohankkeissa. Myös valmiiden lisäosien ja sovelluksien tarkka arviointi edesauttaa toteutuksen jouhevuuksi. Huonosti toteutetun tai sattuman kautta de facto -asemaan päätyneen lisäosan valinta voi aiheuttaa runsaasti turhan ajan käyttöä Googlen kautta helposti löytyneen selvittämättömän ongelman ratkomiseen.

## 1.2 Toimeksiantaja

Opinnäytetyön toimeksiantajana oli Into-Digital Oy, joka on Jyväskylässä, Lappeenrannassa ja Helsingissä toimiva digitoimisto, jonka asiakaskunta koostuu pääosin mainostoimistoista. Suurimmaksi osaksi Into-Digitalin palvelut koostuvat brändisivustoista, kampanjasivuista, Facebook-sovelluksista, display-mainoksista, mobiilisovelluksista sekä muista internetissä tapahtuvan markkinoinnin toteutuksista. Myös mobiilisolustototeutukset ovat yleistyneet yrityksen toimenkuvassa. Into-Digital Oy on yksi viestintäkonserni Into ja Idan tytäryhtiöistä, mainostoimisto Ida Framin ohella.

## 1.3 Tavoite

Opinnäytetyön tavoitteena oli selvittää hyviä käytänteitä mainosalan verkkosivustojen toteutukseen ja niihin liittyviin erityisvaatimuksiin. Vaatimukset käsittävät monesti koko sivustonlaajuisen muokattavuuden asiakkaan toimesta, kieliversioinnin, blogitoiminnon kommentointineen, erikoisfonttien käytön sekä ulkoasun responsiivisuuden mobiililaitteita kohtaan. Opinnäytetyössä käsitellään myös hanketta, jossa ulkopuolinen taho on toteuttanut sivuston graafisen suunnittelun, käyttöliittymäsuunnittelun sekä sivuston rakenteen suunnittelun. Toteutettu hanke käsitteli sivuston sivuston teknistä toteutusta, jonka Into-Digital Oy teki. Sivusto on sijoitettu ulkopuoliselle palvelimelle, joten hanke ei käsittele kävijämääriä ja palvelimen mitoitusta.

Opinnäytetyössä pyrittiin löytämään edellä mainittuihin vaatimuksiin jatkossa käytettäviä ratkaisuja, sillä nämä ovat alati yleistyviä perusvaatimuksia mainosalan sivustoissa. Monesti nämä sivustot koostuvat perustaltaan eri sivutyypeistä, kuten etu-, sisältö- ja yhteystietosivusta. Osa näistä erikoistoiminnoista on yleisesti toteutettava joko yksittäisinä räätälöinteinä tai vaihtoehtoisesti etsimällä luotettavia valmiita lisäosia. Lisäosien käyttö on kuitenkin riskialtista, koska moniin tarkoituksiin ei löydy ns. de facto -ratkaisua, vaan ylistetty lisäosa saattaa olla hyvinkin huonosti toteutettu.

## 2 SISÄLLÖNHALLINTAJÄRJESTELMÄT

### 2.1 Yleistä

Sisällönhallintajärjestelmä on verkkosivun teknisenä pohjana käytettävä palvelimelle asennettava järjestelmä, jonka kautta sivuston sisältö on hallittavissa editorien kautta. Tämä mahdollistaa sivuston hallitsemisen ilman, että muokkaustoimissa tarvitsee osata käsitellä HTML-koodia tai tyylimäärityksiä. Sisällönhallinta myös hoitaa kaikki sivuston tietokantatoimenpiteet.

Tässä opinnäytetyössä sisällönhallintajärjestelmät on rajattu vain LAMP-ympäristöön asennettaviin ilmaisiin avoimen lähdekoodin sisällönhallintajärjestelmiin, jotka ovat laajalti tunnettuja ja käytettyjä.

### 2.2 WordPress

WordPress on yksi maailman suosituimmista Open Source- sisällönhallintajärjestelmistä. Se on hyvin kevyt sovellus, jonka suosio perustuu helppokäyttöisyyteen sekä yksinkertaisuuteen. WordPress kehitettiin b2/cafelog-blogisovelluksen pohjalta 2000-luvun alussa amerikkalaisten kehittäjien Matt Mullenwegin ja Mike Littlen toimesta. WordPressin ensimmäinen versio julkaistiin toukokuussa 2003 ja viimeisin versio 3.5.1 on julkaistu tammikuussa 2013. Wordpress.com tarjoaa valmiiksi WordPressin palvelimelle asennetun ilmaisen sisällönhallintapalvelun ja Wordpress.org ladattavan paketin, jonka voi asentaa omaan webhotelliin tai palvelimelle. (About 2013; History 2013.)

WordPress sisältää itsessään vain olennaisimmat työkalut blogisivuston tai tavanomaisemman verkkosivuston pystyttämiseen ja sisällönhallintaan. Se sisältää artikkeleihin, kategorioihin, kommentointiin ja avainsanoihin perustuvan blogitoiminnon, sisältösivutoiminnon, valikkotyökalun sekä käyttäjien hallinnan. Yksinkertaisista perustoiminnallisuuksista huolimatta WordPressin ollessa avoimen lähdekoodin sovellus



ja erittäin helppo muokattava, sen laaja kehittäjäkunta on kehittänyt merkittävän määrän lisäosia, jotka WordPressin GNU GPL -lisenssin vuoksi ovat myös avoimia lähdekoodiltaan ja siksi myös muokattavissa omiin tarpeisiin. Lisäosat voivat olla ilmaisia tai maksullisia. Opinnäytetyön kirjoitushetkellä niitä oli saatavilla yli 24300 erilaista. (About 2013.)

WordPressin tekniset vaatimukset ovat myös hyvin yksinkertaiset. Uusin versio 3.5.1 vaatii palvelintilalta PHP 5.2.4 -tukea ja MySQL 5.0 -tukea. Monet webhotellit riittävät hyvin WordPress-asennukseen. Jos jostain syystä esimerkiksi PHP 5.2 -tukea ei ole, voidaan käyttää vanhempaa WordPress-versiota 3.1, joka vaatii PHP 4.3 ja MySQL 4.1.2 -tuen. Kaikki vanhat WordPress-versiot ovat ladattavissa WordPressin sivustolta. Monista yleisimmistä lisäosista on saatavilla versiot myös vanhemmille WordPress-asennuksille. (Requirements 2013.)

### 2.3 Drupal

Drupal on WordPressin ja Joomla!':n tavoin avoimeen lähdekoodiin perustuva sisällönhallintajärjestelmä, jonka alkuperäinen kehittäjä on Dries Buytaert. Sen kehitys alkoi vuonna 1999 ja sen ensimmäinen julkaisu oli tammikuussa 2001. Viimeisin versio 7.22 on julkaistu huhtikuussa 2013. Drupalin takana on useiden yleisimpien sisällönhallintajärjestelmien tapaan laaja kehittäjäkunta ja kattava lisäosavalikoima. Kuten WordPress, Drupal vaatii tekniseksi pohjaksi vain PHP 5.2 -tuen sekä MySQL-, PostgreSQL- tai SQLite-tietokannan. Drupal on pitkälti hyvin modulaarinen järjestelmä, josta kehittävä taho voi räätälöidä juuri halutunlaisen kokonaisuuden. (History of Drupal 2013.)

### 2.4 Joomla!

Edellä mainittujen WordPressin ja Drupalin kaltaisesti Joomla! on myös yksi suosituimmista avoimen lähdekoodin sisällönhallintajärjestelmistä. Joomla! on vuonna 2005 julkaistu, Miro Internationalin kehittämä projekti, josta on kehittynyt yksi maa-

ilman tunnetuimmista sisällönhallintajärjestelmistä perinteisien verkkosivujen kehitykseen. Joomla'n uusin versio on 2.5.9, joka on julkaistu helmikuussa 2013. WordPressin ollessa enemmän blogikäyttöön orientoitunut sekä vapaamuotoisempi ohjelmointikäytännöiltään on Joomla! periaatteeltaan suuremmille verkkosivustoille ja intraneteille suunniteltu. Joomla! käyttää MVC-arkkitehtuuria (Model-View-Controller), jossa käytetään kokonaisvaltaisesti olio-ohjelmointia ja toiminnot on eritelty omiin kokonaisuuksiinsa. Myös Joomla! on teknisiltä vaatimuksiltaan hyvin yksinkertainen. Se vaatii PHP 5.3 ja MySQL 5.0.4 -tuen. (What is Joomla? 2013; Technical Requirements 2013; Platform 2012; Features overview 2013.)

## 3 MARKKINOINTISIVUSTOJEN ERIKOISOMINAISUUDET

### 3.1 Yleistä

Kappaleessa esitellyt erikoisominaisuudet: kieliversiointi, responsiivisuus, erikoisfonttien käyttö ja ulkoasun paketointi teemaksi ovat usein mainosalan sivustototeutuksissa toistuvia ominaisuuksia. Toimeksiantajan käyttäessä useimmissa toteutuksissa WordPressiä, kappaleessa rajataan ominaisuuksien toteutustapojen läpikäynti vain WordPressiin.

### 3.2 Kieliversiointi

#### 3.2.1 Yleistä kieliversioinnista

Verkkosivustojen yksi tavanomaisimmista vaatimuksista on kieliversiointi. Yksinkertaisissa yhden sivun sivustoissa voidaan monesti toteuttaa sama tekemällä sivusta duplikaatti ja kääntämällä tekstit halutulle kielelle. Toinen yksinkertainen tapa on tehdä lyhyt yhteenvetosivu, jossa on lyhyesti sivuston kaikki tärkeä tieto käännettynä. Katavampi toteutustapa käännösten toteuttamiseen verkkosivustoissa on kuitenkin tekstisisältöjen versiointi eri kielillä ja niiden keskinäisten suhteiden muodostaminen.

#### 3.2.2 Toteutustapa sisällönhallintajärjestelmässä

Sisällönhallintajärjestelmässä pyritään kieliversioinnin toteutuksessa siihen, että sivuston sivurakenne pysyisi muuttumattomana ja kaikelle sisällölle olisi mahdollisuus tehdä versiot eri kielille. Monissa sisällönhallintajärjestelmissä tämä ei ole perustoitettavuus, vaan se täytyy toteuttaa usein erillisen lisäosan kautta. Monet näistä lisäosista ovat kuitenkin puutteellisia, sillä hyvän kieliversiointitoiminnon tekeminen vaatii monesti sisällönhallintajärjestelmän tietokantarakenteen radikaalia muokkausta. Käytännössä sivuston sisältöjen vaatima rakenne tulee tässä rakennemuutoksessa duplikoitumaan eri kieliversioiden määrän verran.

Jotta kieliversiointitoiminnon käyttö sisällönhallintajärjestelmän hallinnassa olisi mahdollisimman johdonmukaista ja yksinkertaista, tulee muokattavasta tekstisisällöstä päästä luomaan uusi kieliversio tai muokkaamaan olemassa olevaa kieliversiota riittävän helposti. Lisäosassa tulee myös olla selkeät mahdollisuudet määrittää kieliversioinnin asetukset. Tämä on tärkeä asia sivustototeutuksissa, joissa sivuston käytölliittymässä on kielivalitsin josta pääsee auki olevan sivun eri kieliversioille. On tärkeää määrittää poikkeustilanteet, mihin ohjataan jos auki olevasta sivusta ei ole kieliversiota.

### 3.2.3 WordPressin kieliversiointimahdollisuudet

WordPressin vakioasennuksessa tulee mukana vielä toistaiseksi kokeellinen WordPress Multi Site, jota voi käyttää kieliversiointiin. Se on ominaisuus, joka antaa mahdollisuudet tehdä rinnakkaisia asennuksia eri aliosoitteiden alle. Ominaisuus mahdollistaa tehdä samaa teemaa käyttäen erillisiä WordPress-asennuksia, sekä niiden hallinnollisten asetusten muokkaamisen keskitetysti. Multi Site ei kuitenkaan anna mahdollisuutta linkittää sisältöjä keskenään tai tehdä kielivalitsinta, josta pääsee eri kieliversion vastaavalle sivulle. (Create A Network 2013.)

WordPressille on useampia kieliversiointiin tarkoitettuja lisäosia, jotka tarjoavat nämä puuttuvat toiminnallisuudet. Osa näistä lisäosista on ilmaisia osa maksullisia. Ilmaisisista lisäosista yksi suosituimmista on kiinalaisen Qian Qinin kehittämä qTranslate-lisäosa, joka tarjoaa yksinkertaisen käännöstoiminnon artikkeleille ja sivuille. qTranslaten tapauksessa kääntäminen tapahtuu joko tekemällä kieliversiot tai merkitsemällä samaan sisältöön short-tageilla (WordPressin sisältöeditorissa käytettäviä lyhyitä lisäosakohtaisia koodimerkintöjä) tekstin sekaan alla olevan esimerkin mukaisesti (qTranslate 2013.):

```
<!--:en-->Englanninkielinen tekstiversio<!--:-->
<!--:fi-->Suomenkielinen tekstiversio<!--:-->
```

qTranslate on kuitenkin hyvin kevyt lisäosa, joka tarjoaa vain yksinkertaisimmat tarpeet kieliversiointiin. Se tarjoaa artikkelien käännöstyöhön myös automaattisen kääntämisen. Kuten monet ilmaiset pienen kehittäjätahon tekemät lisäosat, qTranslate ei ole joissain tapauksissa täysin yhteensopiva muiden tietokantarakenteita muuttavien lisäosien kanssa. (qTranslate 2013.)

Maksullinen vaihtoehto qTranslatelle on WPML-lisäosaperhe. WPML-lisenssi on kehittäjäkohtainen ja kattaa rajattoman määrän sivuja, joita kehittäjä ylläpitää. Pelkkiin blogitoimintoihin WPML tarjoaa 29 dollarin arvoista lisenssiä, kun taas 79 dollarilla saa koko WordPress-toteutuksen kääntämiseen riittävän lisäosavalikoiman. Artikkeleihin ja sisältösivuihin WPML tarjoaa käyttöliittymässään ominaisuudet tehdä käännösversioita, sekä sisältöjen kopioinnin toisista kieliversioista. WPML tarjoaa myös artikkelien ja sivujen listauksiin indikaattorit jos kieliversiot ovat olemassa listatuista sisällöistä, sekä linkit niiden luomiseen ja muokkaamiseen. Tämä näkymä on esitetty kuviossa 1. (Purchase 2012; Features 2012.)



KUVIO 1. WordPress-sivun hallintanäkymän kielivalitsin

Näiden päätoiminnallisuuksien lisäksi WPML tarjoaa myös lisäosia muiden ominaisuuksien käännösten hallintaan. WPML Core tarjoaa sivuston reititysmahdollisuudet, kuten aliosoitteiden käytön kieliversioissa ja monipuolisen kielivalitsimen, jonka voi

sisällyttää sivuston käyttöliittymään. WPML String Translation on työkalu, jolla sivustolla ilmeneviä tekstejä voi kääntää "search and replace"-periaatteella, kääntäen jokaisen käännökseen viitatus merkkijonon, jotka sivustolta löytyvät. Tämä työkalu on erityisen hyödyllinen widgeteitä ja omia plugineja käyttäessä, jotka eivät WPML:n kautta normaalisti mahdollista kieliversiointia. WPML CMS navigation-lisäosa on työkalu valikoiden kieliversiointiin. (Features 2012; WPML Core and Add-on Plugins 2012.)

WPML on suuren kehittäjätahon toteuttama ja ylläpitämä vakaa ja monipuolinen lisäosa, joka tarjoaa ajoissa tuen uusimmille WordPress-julkaisuille, sekä arkiston vanhemmista lisäosien versioista vanhemmille WordPress-versioille. Huolellisesti toteutettuna WPML on myös hyvin yhteensopiva muiden lisäosien kanssa. WPML tarjoaa lisäpalveluna ammattilaiskääntäjien käännöspalveluja. (Features 2012.)

### 3.3 Erikoisfonttien käyttö

#### 3.3.1 Yleistä erikoisfonteista

Erikoisfonttien käyttö webissä on tullut aina yleisemmäksi ajan saatossa. Tässä kontekstissa erikoisfontit tarkoittavat fontteja, jotka eivät ole ns. websafe-fontteja. Websafe-fontit ovat fontteja, jotka löytyvät useilta päätelaitteilta tai niille on olemassa hyvin samankaltaisia korvaavia fontteja, jos vaadittua fonttia ei löydy. Mainosalan verkkosivustoille niiden käyttö on jo alan puolesta hyvin ominaista. Vaatimuksena näiden fonttien käytölle on esimerkiksi brändien graafisten ulkoasujen tiukat määreet. (CSS Web Safe Font Combinations 2013.)

Minimissään erikoisfonttien käyttö verkkosivustoilla käsittää esimerkiksi sivuston otsikot brändin graafisen ohjeistuksen vaatimalla erikoisfontilla. Ääritilanteissa koko sivuston tekstisisältö saattaa vaatia erikoisfontin käyttöä. Ideaalitalanteessa kuitenkin sivuston leipätekstin, linkkien ja listausten tulisi käyttää websafe-fontteja, koska

useimmissa toteutuksissa tämä ratkaisu parantaa suorituskykyä ja päätelaitteesta riippumatta on varmaa, että tekstisisältö näkyy oikein.

### 3.3.2 Toteutustavat

Erikoisfontteja on mahdollista käyttää usealla tavalla. Luotettavin tapa tähän on tehdä teksteistä, joissa näitä fontteja on käytetty, kuvia ja sisällyttää kuvatiedostot tekstin sekaan. Tämä toteutustapa on erittäin työläs ja tekstien muokkaus vaatii aina uusien kuvien tekemisen. Lisäksi tyylimuutokset eivät vaikuta kuvamuodossa olevaan tekstiin.

Ensimmäisiä erikoisfonttien käyttötapoja olivat mm. sIFR ja Cufón, jotka ovat JavaScript-kirjastoja, jotka muuttavat HTML-tiedostossa olevaa tekstiä eri tekniikoin erikoisfontteiksi. sIFR kääntää fontit Flash-elementeiksi ja Cufón Canvas-elementeiksi. Molemmat tekniikat vaativat fonttitiedoston kääntämisen omilla generaattoreillaan JavaScript-tiedostoiksi sekä oikeuden fonttien upottamiseen verkkosivustoille. Nämä tekniikat ovat kuitenkin epävakaita ja kerryttävät suuren osan sivuston latausajasta. Monesti Cufónilla tehdyt fontit käyttäytyvät eri tavoin selaimien välillä. Myös fonttien piirtyminen sivustolle on monesti häiritsevää, koska useita satoja kilotavuja suurien JavaScript-tiedostojen latautuminen ja suorittuminen sijoittuu melko myöhään sivun latauksessa, sillä niitä käytetään asynkronisesti. (Davidson 2004; About Cufón 2012.)

CSS3 on vähitellen tuonut mukanaan myös fonttien käytön @font-face-tekniikalla. Tämä tekniikka vaatii minimissään fonttitiedostojen sisällyttämisen TrueType-muodossa palvelimelle, sekä niiden kutsumisen tyylitiedoston @font-face-komennolla. Tämä tekniikka on erittäin vakaa ja nopea. Fontit piirtyvät parhaimmalla mahdollisella tavalla, sekä muotoilu toimii paremmin kuin esimerkiksi Cufónia käyttäessä. @font-facen ollessa CSS3-tekniikka, se ei toimi vanhemmilla selaimilla. Tällä hetkellä laajalti käytössä olevista selaimista @font-face ei tue Microsoftin Internet Explorer 8:aa tai vanhempaa. Jotta @font-face toimisi mahdollisimman monessa päätelaitteessa, tulee fontista olla useita erityyppisiä versioita. Näiden versioiden tekemi-

seen löytyy muutamia verkkopalveluita, kuten FontSquirrel. Nämä palvelut muodostavat alkuperäisestä TrueType- tai OpenType-fontista SVG-, EOT-, ja WOFF-versiot, sekä generoi automaattisesti tyylitiedostoon liitettävän @font-face-tagin. Myös @font-face vaatii käytettävältä fontilta upotusoikeuden ja nämä fonttigeneraattorit eivät anna generoida muita versioita jos fonttien kehittäjät ovat lisänneet fonttinsa ns. mustille listoille. (Introduction to web typography and @font-face 2011; CSS3 @font-face Rule 2013.)

Koska fonttien käyttöoikeuksien ostaminen verkkosivustolle upotukseen voi olla erittäin kallista, ei se ole välttämättä paras ratkaisu lyhytaikaisen kampanjasivuston toteutukseen. Muutamat tahot, kuten Google, Fonts.com ja Typekit, tarjoavat palveluja, jotka tarjoavat fontteja käytettäväksi joko ilmaiseksi tai vuosittaista pientä maksua vastaan. Nämä palvelut muodostavat hinnoittelun tai ilmaispalvelun joko sivun kävijämäärien mukaan. Esimerkiksi Typekitin tarjoama 49.99 dollaria vuodessa maksava palvelu kattaa mainostoimiston tarpeet erittäin hyvin, tarjoten 500,000 sivulatausta kuukaudessa, sekä kaikkien heidän tarjoamien fonttien rajoittamattoman käytön rajoittamattomassa määrässä verkkosivustoja. Vastaavasti yksi fonttiperhe, joka sisältää sen eri variaatiot, voi maksaa tuhansia euroja. Fontin ostaessaan yritys saa yrityksen kuitenkin sen käyttöoikeudet myös muissa medioissa. Tämä menetelmä on yleinen siinä tapauksessa, kun suurempi brändi sisällyttää erikoisfonttien käyttöä graafiseen ohjeistukseensa. (Pricing 2013.)

## 3.4 Teemat

### 3.4.1 Yleistä teemoista

WordPress-toteutuksessa ulkoasu ja perustoimintojen muokkaukset kootaan kokonaisuuksiin, joita kutsutaan teemoiksi. Teema sisältää sivustosta kaiken, mitä ei hallita sisällönhallintatyökalujen kautta. Teemojen tarkoitus WordPressissä on muodostaa yhteen kansioon sisällytettävä kokonaisuus tiedostoista, joita tarvitaan sivuston ulkoasuun. (Theme Review 2013.)



### 3.4.2 Teeman rakenne

WordPress-sivuston ulkoasussa näkyvä HTML-koodi rakentuu teemassa sijaitsevista template-tiedostoista, joista sisällytetään tarpeellisia osia riippuen ladattavan sivun tyyppistä. Template-tiedostot sisältävät HTML-koodia, jonne on sijoitettu PHP-kielillä WordPressin omilla komennoilla (Template Tags) haettavaa sisältöä, kuten tekstisisältöä, otsikoita, kuvia tai sisältösivun ja artikkelien ominaisuuksia, kuten julkaisupäivämääriä, kategoriatietoja ja kirjoittajatietoja. Keskeisimpiä teeman ulkoasuelementtejä verkkosivutoteutuksissa ovat tiedostot: header.php, footer.php ja page.php, johon header ja footer sisällytetään. Teema sisältää juurikansiossaan myös tavanomaisen tyyli-tiedoston style.css, johon sivuston tyylit ja teeman tiedot sisällytetään. (Theme Review 2013; Stepping Into Template Tags 2013.)

Teema sisältää usein tiedoston functions.php, jossa tehdään kaikki WordPressin vakiotoiminnallisuuden käsittely. Functions.php:n pääasiallinen tarkoitus on tarjota helppo alusta toiminnallisuuksien muokkaamiseen ja lisäämiseen ilman, että kehittäjän tarvitsee muokata WordPressin lähdekooditiedostoja. WordPress tarjoaa toimintojen muokkaamiseen Hook-komentoja, joilla voidaan lisätä koodia suoritettavaksi jonkin WordPressin perustoiminnallisuuden yhteyteen. Tällä toiminnallisuuksien keskittämällä helpotetaan myös ongelmanratkaisua, sekä selkeytetään koodin rakennetta. (Theme Review 2013; Creating Your Own Page Templates 2013; Stepping Into Template Tags 2013.)

Functions.php-tiedosto mahdollistaa myös suuren osan WordPress-asennuksen hallintaosion kustomoinnista, kuten perustoiminnallisuuksien käytön estämisen sekä käyttäjätasojen oikeuksien muokkaamisen. Nämä muokausmahdollisuudet ovat käytännöllisiä niissä tapauksissa, jolloin halutaan karsia hallintaosiosta tarpeettomia toiminnallisuuksia pois. (Stepping Into Template Tags 2013.)

Teeman pakollisia tiedostoja ovat:

- index.php – Sivupohja, jota käytetään oletuksena jos muita sivupohjia ei löydy.
- comments.php – Sivupohja, jossa listataan artikkelin kommentit
- screenshot.png – Esimerkkikuva teeman vakioulkooasusta, joka näytetään teeman valintaruudussa
- style.css – Tyylitiedosto, jossa on minimissään oltava teeman tiedot, kuten teeman nimi, kehittäjä tiedot, versionumerointi ja versioyhteensopivuus.

Teemaan suositellaan sisällyttää useimpien WordPressin perustoimintojen käytön mahdollistamiseksi:

- 404.php – Sivupohja, joka näytetään tilanteessa, jossa haluttua sivua ei löydy.
- archive.php – Arkistosivupohja, joka listaa artikkelit julkaisuajan mukaan
- page.php – Oletuksena käytettävä sisältösivupohja
- search.php – Sivupohja, jossa näytetään haun tulokset, Wordpressin hakutoimintoa käytettäessä
- single.php – Yhden artikkelin näyttämiseen tarkoitettu sivupohja
- header.php – Sivuston headerin sisällön liittämiseen tarkoitettu template, jonka tulisi sisältää kaikki joka sivulla käytettävät elementit, kuten <head>-elementti.
- footer.php – Sivuston footerin sisällön liittämiseen tarkoitettu template. Käytetään, kuten header.php:ta, mutta sivuston pohjaosaan.
- sidebar.php – Sivuston widgetien sisällyttämiseen käytettävä template.

Muita yleisimpiä tiedostoja, joita voidaan sisällyttää teemaan valinnaisesti:

- author.php – WordPress-asennuksen käyttäjän esittelyyn tarkoitettu sivupohja.
- category.php – Kategorian alla olevien artikkeleiden listaukseen tarkoitettu sivupohja.
- tag.php – Yhden asiasanan alla olevien artikkeleiden listaukseen tarkoitettu sivupohja.

WordPress-asennuksen mukana tulee esimerkkiteema Twentyeleven, joka sisältää kaikille perustoiminnoille yksinkertaiset toteutustavat ja kattavat esimerkit `functions.php`:n eri tekniikoiden käyttötavoista. Twentyeleven tarjoaa myös valmiin pohjan oman teeman jalostamiselle. (Creating Your Own Page Templates 2013.)

### 3.4.3 Sivupohjan rakenne

WordPressin sivupohjat pohjautuvat silmukkatekniikkaan, jota kutsutaan loopiksi. Eri sivupohjat, kuten kategorialistaus, yksittäisen artikkelin sivu, sisältösivu, tai haun tulosten listaussivu perustuvat loopiin. Loop-tekniikka käyttää aina samaa syntaksia ja sen hakema sisältö riippuu sivupohjan tyypistä. Loop käydään `have_posts()`-metodin ja `while`-silmukan kanssa läpi, joka mahdollistaa WordPressin omien `template tag`-funktioiden käyttämisen silmukan sisällä. `Template tag`-funktiot ovat lyhyitä ja selkeitä komentoja, joilla silmukassa olevasta elementistä saadaan oleellisia tietoja tulostettua sivupohjaan. Esimerkiksi funktio `the_title()` tulostaa suoraan artikkelille tai sivulle määritetyn otsikon. Vastaavasti `the_content()` tulostaa artikkelin tai sivun sisällön muotoiluineen. (The Loop 2013.)

Loopin ollessa sivupohjan ydin on muut toistuvat elementit, kuten header, footer ja sivupalkki, tarkoitus tulostaa omille paikoilleen niille tarkoitetuilla funktioilla. Esimerkiksi laittamalla `get_header()`-funktion tiedoston alkuun, tulostaa funktio `header.php`:n sisällön kohtaan, johon se on sijoitettu sivupohjassa. `Header.php` sisältää kaiken `html`-tiedoston alkuun liittyvät elementit ja ulkoasussa sivun yläosan toistuvat elementit, kuten logon ja navigaation. Vastaavasti `footer.php` sisältää sivuston ulkoasun alaosassa olevat elementit sekä `body`-elementin loppuun sisällytettävät asiat, kuten kävijäseurannat ja mahdolliset Javascript-tiedostojen sisällytykset. (The Loop 2013.)

Alla on esimerkki yksinkertaisesta sisältösivuna käytettävästä sivupohjasta. Tiedoston alussa kommenttien sisässä oleva merkintä "Template Name: Alasivu" määrittää si-

vupohjan nimen, jolla se voidaan kiinnittää sisältösivun sivupohjaksi hallinnasta. Funktiot `get_header()`, `get_footer()` ja `get_sidebar()` sisällyttävät lopulliseen HTML-koodiin omat template-tiedostonsa. Content-elementin sisällä oleva while-silmukka on loop-tekniikka, joka sisällyttää jokaiseen silmukan iteraatioon post-elementin. Post-elementin sisällä on käytetty edellämainittuja WordPressin template tag-komentoja, joista `the_title()` tuo sivun otsikon ja `the_content()` sisällön muotoiluineen. (Creating Your Own Page Templates 2013.)

```
<?php
    /* Template Name: Alasivu */
    get_header();
?>
<div id="content" role="main">
    <?php while ( have_posts() ) : the_post(); ?>
        <div class="post">
            <h2 class="page-title"><?php the_title(); ?></h2>
            <?php the_content(); ?>
        </div>
    <?php endwhile; ?>
</div>
<?php get_sidebar(); ?>
<?php get_footer(); ?>
```

## 3.5 Responsiivinen suunnittelu

### 3.5.1 Yleistä responsiivisesta suunnittelusta

Responsiivinen suunnittelu tarkoittaa sivuston ulkoasun muokkaamista paremmin yhteensopivaksi laitteille, joissa on pienempiä näyttöjä kuin tavanomaisissa tietokoneissa. Responsiivinen suunnittelu keskittyy monesti käyttöliittymän yksinkertaistamiseen sekä typografian selkeyttämiseen käytettävyyden parantamiseksi. Älypuhelimien ja tablettitietokoneiden yleistyessä ulkoasun optimoinnin oletetaan olevan yhä enemmän itsestään selvä ominaisuus mainosalan sivustossa. Tilastosivusto StatCounterin mukaan mobiiliselainten osuus on jo noussut lähes 15% kaikista päätelaitteista ja niiden osuus on jatkuvassa nousussa. (Mobile vs. Desktop 2013.)

### 3.5.2 Tekniikat

Responsiiviseen toteutukseen on useampia yksinkertaisia tekniikoita. Teknisen toteutuksen puolesta responsiivisen suunnittelun päämääräinen tarkoitus on näyttää mobiililaitteilla sisältö pienemmälle näytön resoluutiolle sopivalla tavalla häiritsemättä sivuston ulkoasua näytettäessä tietokoneen selaimella. Idean ollessa yksinkertainen, monet toteutustavat ovat yksinkertaisia ja yhteensopivia useiden vanhempien selainversioiden kanssa. (Pettit 2012.)

Yleisin tapa toteuttaa responsiivinen taitto teknisesti on käyttää CSS:n media query-tekniikkaa, jolla päätelaitteen tyyppi tunnistetaan päätelaitteen selaintietojen avulla antamalla esimerkiksi HTML-koodissa tyylitiedostoon viitatessa media-attribuutille haluttu päätelaitteen tyyppi. Alla olevalla esimerkillä voidaan kaikkien mobiililaitteiden kohdalla ottaa käyttöön vain mobiililaitteille tarkoitettu tyylitiedosto `mobile.css`. (Pettit 2012; Patel 2012.)

```
<link rel="stylesheet" media="screen" href="style.css" />  
<link rel="stylesheet" media="mobile" href="mobile.css" />
```

Vaihtoehtoisesti tyylimäärityksiä voidaan ottaa käyttöön selaimen leveyden ollessa tiettyjen leveyksien rajoissa. Esimerkkinä tästä, sivun tyylit voidaan jaotella käytettäväksi yleisimpien mobiililaitteiden leveyksien mukaan: tabletilaitteen vaaka-asento 1024px, tabletilaitteen pystyasento 768px, älypuhelimien vaaka-asento 480px ja älypuhelimien pystyasento 320px. Alla olevan esimerkin mukaisessa toteutuksessa media query-tunnisteet laitetaan sivuston normaalien tyylien alapuolelle ja niiden sisällä tehdään päätelaitekohtaisia muokkauksia. Koska media query-tunnisteilla on tässä tapauksessa vain maksimilevydet, periytyvät esimerkiksi leveämmille selaimille tarkoitetut tyylit myös kapeammille selaimille. Kuviossa 2 on esitetty esimerkki sisällön asemoinnista kynnystasoittain eri laitteille. Media queryt eivät vaikuta esimerkissä työpöytäversioon, mutta muuttavat asemointia tabletilaitteen ja älypuhelimien osalta. (CSS Media Types 2013.)

```

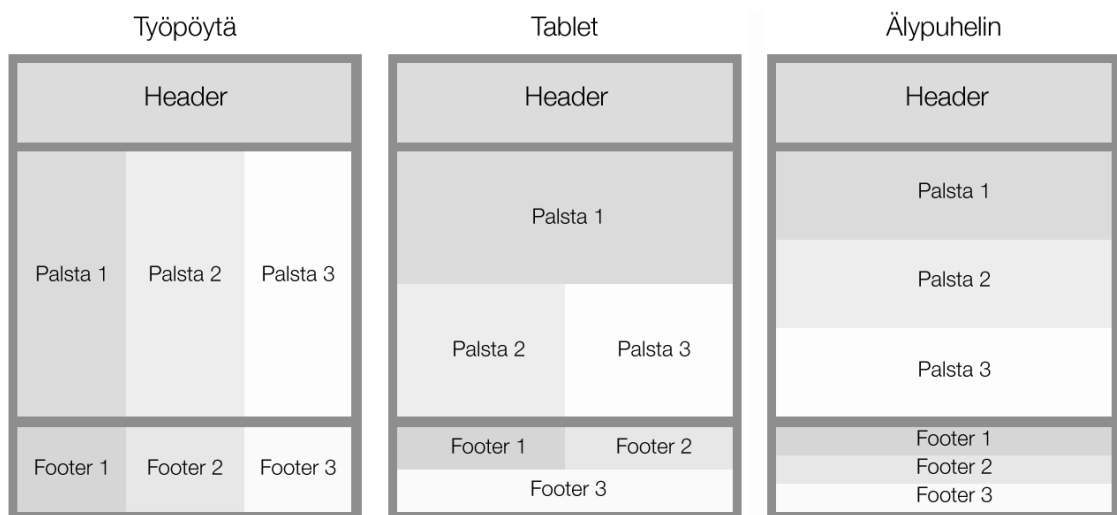
@media screen and (max-width: 1024px)
{
  /* Tyylimäärittelykset tabletilaitteen vaaka-asennolle */
}

@media screen and (max-width: 768px)
{
  /* Tyylimäärittelykset tabletilaitteen pystyasennolle */
}

@media screen and (max-width: 480px)
{
  /* Tyylimäärittelykset älypuhelimien vaaka-asennolle */
}

@media screen and (max-width: 320px)
{
  /* Tyylimäärittelykset älypuhelimien pystyasennolle */
}

```

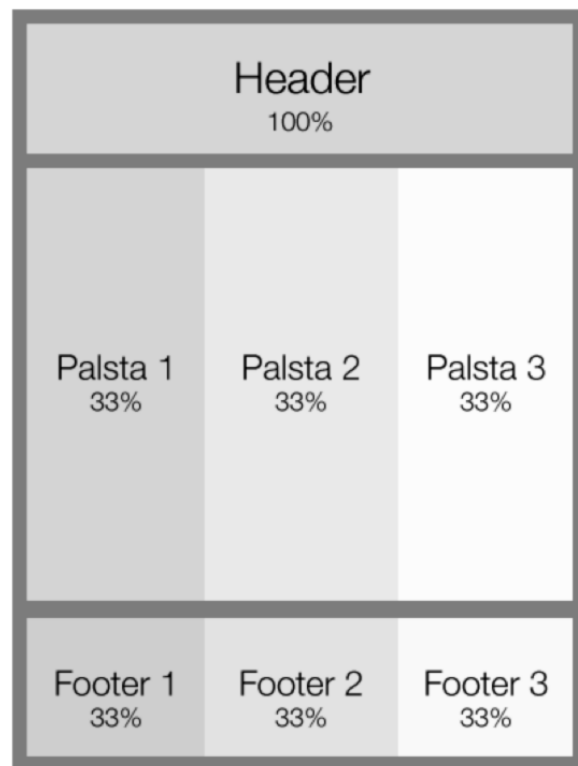


KUVIO 2. Esimerkki media query-tekniikalla porrastetusta responsiivisuudesta

Sivustosta voidaan myös tehdä kokonaan erillinen mobiilikäyttöön tarkoitettu versio, johon käyttäjä ohjataan saapuessaan mobiililaitteella sivulle. Tätä tekniikkaa käytetään tilanteissa, joissa halutaan näyttää sivu, joka on merkittävästi sisällöltään erilai-

nen kuin desktop-versio. Esimerkiksi, jos halutaan näyttää mobiililaitteilla pelkästään tärkeimmät tiedot, käytetään tätä tekniikkaa. (Patel 2012.)

Verkkosivuston ulkoasu voidaan toteuttaa siten, että se skaalautuu selaimen leveydestä riippuen. Tällöin elementtien leveydet määritetään prosentteina pikseliarvojen sijaan ja koko ulkoasulle määritetään maksimileveys, jolloin sisältö keskitetään. Tätä toteutustapaa kutsutaan Fluid-taitoksi. Monessa tapauksessa, joissa sisältö on aseteltu palstoihin, tämä tekniikka on yhdistettynä media query:llä tehtäviin laitekohtaisiin porrastuksiin. Kuviossa 3 on esitelty tavanomainen fluid-taitto, jossa sisältöelementtien leveydet ovat suhteellisia selaimen leveyteen. (Pettit 2012.)



KUVIO 3. Esimerkki fluid-taitosta

## 4 MARKKINOINTISIVUSTON TOTEUTUSHANKE

### 4.1 Lähtötilanne

Hankkeen lähtökohtana oli tilanne, jossa suomalainen yritys halusi uuden verkkosivuston brändiuudistuksen yhteydessä. Yritys oli uudistanut brändinsä bränditoimiston kanssa, joka oli luovuttanut graafisen ohjeistuksen, sivuston rautalankamallit ja karkean graafisen ulkoasun Into-Digitalille viimeisteltäväksi. Hankkeeseen osallistunut mainostoimisto keskittyi projektissa sisällöntuotantoon ja kuvitukseen.

Sivuston sisällön sovittiin olevan kokonaan muokattavissa, jotta mainostoimisto voisi muokata lähes kaikkia sivustolla olevia teksti- ja kuvasisältöjä. Ratkaisu tähän oli sivuston rakentaminen sisällönhallintajärjestelmän päälle. Sisällönhallinnan tulisi olla mahdollisimman helppo käyttää, jotta asiakkaan ei tarvitsisi opetella erikoistekniikoi- ta, vaan sisällönhallinta onnistuisi WYSIWYG-työkaluilla.

Suunnitteluvaiheessa sivusto koostui perinteisistä elementeistä. Sivustolla tul-taisiin käyttämään kolmea sisältösivutyyppeä: etusivua, sisältösivua, ja yhteystietosivua. Li-säksi sivustolle tulisi blogiosio sekä tapahtumaosio. Etusivu, joka toimisi portaalina muihin sivuston alaosioihin, sisältäisi selkeät linkit sivuston neljälle pääsisältösivulle, listaukset blogiartikkeleista ja tapahtumista, sekä Facebookin like boxin. Sisältösivut olisivat yksinkertaisia sivuja, jotka koostuisivat kahdesta palstasta: vasen palsta koko-naan tekstisisällölle ja oikea kuville. Yhteystietosivu myötäilisi muuten sisältösivun linjoja, mutta oikea palsta sisältäisi interaktiivisen Google maps-kartan.

Sivusto sisältäisi kautta linjan saman headerin ja footerin. Header sisältäisi vain yrityk-sen logon ja kaksitasoisen navigaation. Navigaation ensimmäinen taso olisi aina näky-villä ja sisältäisi sivuston päätason sivut. Jos auki olevalla sisältösivulla olisi alisivuja, navigaation ensimmäisen tason alapuolelle aukeaisi navigaation toinen taso, jossa listattaisi kaikki sivun kaikki alisivut. Footer sisältäisi vain ensimmäisen tason navigaa-tion ja tekijänoikeustiedot.



Sivustolle suunniteltiin myös blogitoiminto, jossa yritys voisi kertoa kuulumisistaan ja menneistä tapahtumista. Blogitoimintoon suunniteltiin listausnäkyvä, jossa näkyisi artikkelin otsikko, lyhyt katkelma artikkelista sekä erikseen määritettävä artikkeliin liittyvä kuva. Blogitoiminto tulisi myös sisältää yksittäisen artikkelin näkymän, jossa näytetään edellämäinittujen elementtien lisäksi koko artikkelin sisältö, sekä kommentointitoiminto, jossa vierailijat voisivat kommentoida artikkeleita yksinkertaisesta kommenttilomakkeesta jättämällä nimensä, sähköpostiosoitteensa ja kommenttinsa. Blogitoiminnon ohella sivulle haluttiin myös toinen artikkelitoiminto. Yritys halusi tapahtumalistauksen, joka toimisi samalla tavalla, kuten blogitoiminto mutta ei sisältäisi kommentointia eikä kuvia listauksessa.

Sivuston ulkoasusta suunniteltiin toteutettavaksi kaksi versiota. Toinen versio olisi ns. työpöytäversio eli tietokoneen selaimella käytettävä kiinteäleveysinen versio, jossa sisältö keskittyisi sivusuunnassa jos näyttö olisi leveämpi kuin 990 pikseliä leveä sisältöalue. Laitteelle jonka näytön leveys on pienempi kuin 990 pikseliä, toteutettaisiin ulkoasu responsiivisena taittona.

Näiden perustoimintojen lisäksi yritys halusi myös eriväriset teksteemat jokaista sivuston neljää sisältö-osiota kohden. Mahdollisesti myöhemmin toteutuvina toimintoina haluttiin lomaketyökalu, jolla asiakas voisi luoda sivustolle erinäisiä ilmoittautumislomakkeita tapahtumiin, sekä koko sivuston laajuisen kieliversioinnin. Pääpainona sivuston toiminnalle tuli olla selkeys, nopea ladattavuus, helppokäyttöisyys ja helppo muokattavuus hallinnan puolella.

## 4.2 Tekninen suunnittelu

Projektin suunnitteluvaiheessa asetettiin asiakkaan toimesta vaatimukset, minkä puitteissa sivusto tulisi teknisesti toteuttaa. Palveluntarjoajaksi ilmoitettiin Nebula ja tekniseksi alustaksi heidän tarjoamansa Webhotel Pro -palvelintila. Teknisenä pohjana

tulisi siis olemaan LAMP-ympäristö, joten sivuston tekniikan tulisi pohjautua palvelimen puolella PHP-kieleen. Levytilaa Webhotel Pro tarjoaa 2 Gt, joten tilaa on riittävästi lähes kaikkiin PHP-pohjaisiin CMS-järjestelmiin. Webhotel Pro ei sisällä sellaiseenaan tietokantaa, mutta CMS:ien vaatima MySQL-tietokanta on saatavilla lisäominaisuutena. (Webhotellit 2013.)

Liikennöintisuositukseksi Nebula ilmoittaa 50 Gt/kk, joka olisi asiakkaan mukaan riittävä määrä sivuston oletettuihin kävijämääriin. Jos liikennöinti ylittää suosituksen, Nebula ilmoittaa asiakkaalle ylityksestä. Toistuvista ylityksistä Nebula vaatii vaihtamaan palvelupaketin toiseen pakettiin, jossa on suurempi kapasiteetti. (Webhotellit 2013.)

### **4.3. Sisällönhallintajärjestelmän valinta**

#### **4.3.1 Huomioonotettavat asiat sisällönhallinnan valinnassa**

Yhtenä asiakkaan perusvaatimuksista oli sivuston helppo hallittavuus, joten käytettävän CMS:n valinta rajoittui ilmaisiin, laajalti käytössä oleviin open source-järjestelmiin. Suositun järjestelmän käyttö oli aiheellista siksi, että yleisimmät CMS:t ovat hyvin dokumentoituja ja ohjeistettuja molemmille tahoille, sekä kehittäjille että sisällönhallinnan käyttäjälle. Nopea tutustuminen poiki kolme Into-Digitalille jo ennestään tuttua ehdokasta: WordPress, Joomla! ja Drupal. Kaikkia kolmea on käytetty yrityksen tekemissä toteutuksissa ja ovat hyväksi todettuja alustoja.

Alustan valinnassa tuli kiinnittää huomio pääasiassa siihen, kuinka paljon resursseja kunkin järjestelmän käyttö kuluttaisi. Työmäärän mitoitus riippui pääosin kahdesta asiasta: Onko ehdolla oleva järjestelmä toteuttajalle ennestään tuttu eli tulisiko resursseja varata järjestelmään tutustumiseen ja kehitystyön opetteluun, vai onnistuisiko toteutus suoraan räätälöimällä ennestään tuttua järjestelmää? Jos valinta tehdään sillä perusteella, että järjestelmä on kehittäjälle tuttu, riittäisivätkö sen perusominais-

suudet ja laajasti käytettyjen luotettavien lisäosien ominaisuudet sivuston vaatimuksiin?

Into-Digitalin sisällönhallintaa vaativat verkkosivustoprojektit toteutetaan monesti WordPressin päälle hallintatyökalujen yksinkertaisuudesta johtuen. Koska useimmissa tilanteissa asiakas haluaa itse muokata ja lisätä sivuston sisältöjä, verkkosivustoprojekteissa halutaan käyttää mahdollisimman paljon WordPressin vakiotoimintoja niiden hyvän dokumentoinnin ja WordPressin levinneisyyden takia. Toisinaan WordPress on yrityksen projekteihin liiankin yksinkertaistettu ja pelkistetty. Näissä tapauksissa tulee harkita vaihtoehtoisia ratkaisuja, jotta WordPressin perustoiminnallisuutta ei vakaus- ja tietoturvallisuussyistä tarvitsisi lähteä liiaksi muokkaamaan.

Drupal on ollut Into-Digitalin projekteissa käytössä lähinnä niissä tilanteissa, jossa tiedetään tarvittavaksi tiettyä Drupalin olemassaolevaa lisäosaa, on verkkosivusto alunperin toteutettu jonkin toisen tahon toimesta Drupalilla tai niissä projekteissa missä tiedetään, ettei WordPress sovellu käytettäväksi liian yksinkertaistettuna järjestelmänä. Verrattuna WordPressiin, on yrityksen toteuttamissa projekteissa yleisesti Drupal ominaisuuksiensa puolesta hieman monimutkaisemman ja raskaamman oloinen, sekä kehitystyöltään hieman työlämpi kuin WordPress. Tämän lisäksi monissa yrityksen projekteissa Drupal ei vakioasennuksenaan tarjoa tarpeeksi yksinkertaista sisällönhallintatoiminnallisuutta, jolloin sivuston sisältö ei olisi tarpeeksi helposti suoraan asiakkaan hallittavana.

Joomla!:n käyttö on Into-Digitalin toteuttamissa verkkosivustoissa ollut melko vähäistä. Yleensä käytön syynä on ollut projektin aikaisempi toteutus Joomla!:lla ja jatkehityks on tehty Into-Digitalin toimesta. Joomla!:n perustoiminnallisuudet jäävät projektien vaatimusten puitteissa useissa tapauksissa liian vähälle ja esimerkiksi WordPressin yksinkertaisemmat hallintatyökalut mielletään paremmin sivuston sisältöä hallitsevan tahon tarpeisiin sopivammaksi.

### 4.3.2 Valinta

Projektin teknisen suunnittelun merkittävimpiä kysymyksiä oli sisällönhallintajärjestelmän valinta. Valinta tehtiin perusteella, olisiko CMS-ratkaisussa jo valmiiksi mahdollisimman paljon projektin vaatimia toiminnallisuuksia jo valmiina. Myöskin toteuttajan aikaisempi kokemus sisällönhallintajärjestelmästä oli merkittävä peruste valinnalle.

Wordpress nousi kolmesta vaihtoehdosta kärkeen jo alkuvaiheessa myös siksi, että asiakas oli toivonut sen käyttöä sivuston toteutuksessa. Myös valmiit blogitoiminnot, kuten artikkelit, kategoriat ja kommentoinnit, nostivat WordPressin kärkiehdokkaaksi. Muutamia pieniä vaatimuksia kuten sisältösivujen neljän kategorian erotteluja väri-teemoilla, eikä kieliversiointia ollut saatavilla suoraan WordPressin toiminnallisuuksista. (Drupal-Joomla-Wordpress-Comparison 2013.)

Drupalia ei valittu, koska se ei ollut tekijälle tuttu järjestelmä ja sen sisällönhallintatyökalut eivät olleet tarpeeksi yksinkertaiset siihen nähden, että asiakas toteuttaisi sisällön syötön pääosin itse. Kieliversiointimahdollisuudet olisivat paremmat, sekä tekniseltä kannalta blogitoiminnallisuuksien ja sisältösivujen erottelu oli paremmin toteutettu. Tämä olisi ollut mahdollisesti merkittävä seikka sivuston jatkokehitystä ennakkoon suunnitellessa. (Wilding 2011.)

Joomla! ei olisi tarjonnut tähän verkkosivustoon yhtä yksinkertaista blogitoiminnallisuutta, sekä sen kehitystyö ja lisäosat eivät olisi olleet tuttuja kehittäjälle. Joomla:ssa olisi tosin ollut natiivi tuki kieliversioinnille, mutta blogitoiminto olisi pitänyt toteuttaa lisäosilla.

Lopulliseksi valinnaksi päättyi WordPress sen tarjoamien valmiiden toiminnallisuuksien ja tekijän aikaisemman kokemuksen vuoksi. Ainoaksi epäselväksi asiaksi jäänyt kieliversiointi on käyty tarkemmin läpi luvussa "3.2 Kieliversiointi"

#### 4.4 Käyttöönotto

Nebulan Webhotel Pro:ssa vakiona käytössä oleva PHP-versio 4.3 on nostettu juurikansion .htaccess-tiedostossa versioon 5.2, joka mahdollistaa uusimman WordPress-version 3.5.1 käytön. Asennusta varten haettiin WordPress.org-sivustolta uusin paketti, joka siirrettiin kotisivutilan alikansioon. Tilan juurihakemistossa oli toteutettavan sivuston vanha versio, joka otettiin pois käytöstä vasta uuden sivuston julkaisun yhteydessä. (PHP-Ohjelmointi 2013)

WordPressin asennus palvelimelle on tehty järjestelmän oman asennusvelhon avulla. Asennusvelho pyytää tarkistettuaan yhteensopivuuden palvelimen ominaisuuksien kanssa, syöttämään sisällönhallinnan vaatimat perustiedot asennusympäristöstä, kuten tietokannan tiedot ja käyttäjätunnukset, sivuston nimen, ylläpitäjän sähköpostiosoitteen, halutun ylläpitäjän käyttäjätunnuksen ja salasanan, sekä lyhyen kuvauksen sivusta. Tietokantamäärittelyä lukuun ottamatta, asetukset ovat muokattavissa hallintapaneelista.

WordPressin viisitoista minuuttia vievän asennuksen jälkeen sivusto oli täysin käytettävissä. Hallinnan kautta sivustolle tehtiin ylläpitäjän lisäksi muutamia käyttäjätunnuksia katselmointikäyttöön ja asiakkaan käytettäväksi. Asennuksen jälkeen sivustolle tehtiin valmiiksi perusrakenteet, kuten sisältösivujen alustava rakenne, navigaatiot valikkotyökalulla, artikkelilistauksien vaatimat kategoriat, sekä otettiin käyttöön tarvittavat perusominaisuudet blogin käyttöön, kuten kommentointi ja avainsanojen eli tagien käyttö.

Sivustolle on asennettu muutamia yleishyödyllisiä tuttuja ja luotettavia lisäosia kuten hakukoneoptimointiin tarkoitettu All In One SEO-Pack, sekä Advanced TinyMCE-sisältöeditori, joka muuttaa WordPressin sisältöeditoria perinteisien tekstinkäsittelyohjelmien käyttöliittymää enemmän vastaavaksi.

## 4.5 Erikoisominaisuuksien toteutus

Sivuston vaatimusmäärittelyssä mainittu kieliversiointi on toteutettu WordPressin WPML-lisäosalla. Sivuston pääkielen ollessa suomi, on osalle sisällöstä olemassa myös ruotsinkieliset versiot. Kieliversion pääasiallinen käyttöliittymä koostuu sivun headerissa olevasta kielivalitsimesta, josta siirrytään etusivulle tai auki olevan sivun kieliversioon, jos sellainen on määritetty sivulle. Kielivalitsin on toteutettu WPML:n omalla kielivalitsintoiminnolla, joka on asetuksia voi muokata WPML:n hallintapaneelista. Kielivalitsimessa on vakioasetuksena kieliversioiden indikaattorina maiden liput, mutta sivustolla liput on korvattu tekstiversioilla. Muokkaus on tehty muuttamalla kielivalitsimen koodia headerin sivupohjasta header.php-tiedostosta.

Sivustolla oli julkaisun yhteydessä asiakkaan pyynnöstä toiminto, joka näyttää ruotsinkielisellä sivustolla blogiosiossa suomenkieliset blogimerkinnät. Tämä ei kuulunut WPML:n vakiotoiminnallisuuteen, joten se toteutettiin muuttamalla blogilistausten sivupohjan loopin toimintaa, suodattamalla näytettävää sisältöä. Tämä toiminto otettiin myöhemmin pois ja blogikirjoituksista tehtiin kieliversioinnit molemmille kielille.

Erikoisfonttien osalta tämän projektin tapauksessa oli graafisessa suunnittelussa käytetty ainoastaan Futura-fonttia. Fontti oli suunniteltu käytettäväksi kaikissa tekstisällöissä sekä navigaatioissa. Cufón suljettiin pois vaihtoehdoista jo aikaisessa vaiheessa, koska se on hyvin epävakaa ja vanhanaikainen. Jäljelle jäivät fontin käyttö @font-face:lla tai käyttää vuosittain maksettavaa fonttipalvelua. Asiakas päätti käyttää Typekit-yrityksen tarjoamaa palvelua jossa fontti ladataan käyttöön Typekitin palvelimella sijaitsevan JavaScript-tiedoston avulla. Fontti on itsessään melko raskas 350 kilotavun tiedostokoolla, mutta toimii moitteettomasti kaikilla päätelaitteilla ja selaimilla.

Typekitin käyttöönotto tapahtui kirjautumalla palvelun sivustolle tilauksen ja rekisteröinnin yhteydessä saatavilla käyttäjätunnuksella ja salasanalla, jonka jälkeen luotiin projekti johon käytettävät fontit liitettiin. Projekti tarkoittaa käytännössä yhtä verk-

kosivustoa johon ladataan vain käytettävien fonttien tarvittavat variaatiot, jotta ladattava JavaScript-tiedosto ei kuormittaisi sivuston latausaikaa liiaksi. Projektille asetettiin sallitut domainit, jotta JavaScript-tiedostoa ei voi käyttää ulkopuolisilla sivustoilla. Tämän jälkeen fontti oli käytettävissä tyylitiedoston kautta font-family-attribuuttia käyttämällä, kuten tiedoston alussa @font-face:lla tuotu lokaali fonttiedosto. Eri variaatioiden käyttö Futura-fontista (Bold, Heavy, Normal) vaati Typekitiä käytettäessä font-weight-attribuutin numeroarvoja.

Koko sivuston ulkoasu, sivupohjat, erikoisominaisuuksien käyttöönotto sekä pienemmät toiminnallisuusräätälöinnit on sisällytetty sivustolle tehtyyn omaan teemaan, joka on alusta asti Into-Digital Oy:n toteuttama. Kaikki perustoiminnot, kuten sisältösivut, blogitoiminto ja navigaatiot on toteutettu WordPressin perustyökaluilla joka mahdollistaa niiden käyttämisen hallintapaneelin kautta. Sivuston header ja footer, toistuvat elementit, sekä navigaatio sijoitettiin header.php- ja footer.php-tiedostoon, sekä toiminnallisuudet functions.php-tiedostoon. Sivuston kahdella eri artikkelilistausnäkyillä on eroavaisuuksia ulkoasussa. Tämä on ratkaistu tekemällä molemmille listauksille omat category.php-tiedostot, joissa kummassakin näytetään eri elementit artikkelilistauksissa. Kaikki artikkelisivut käyttävät samaa yksittäisen artikkelin sivupohjaa. Blogin toiminnallisuudet ovat muokkaamattomia WordPressin perustoiminnallisuuksia. Vain ulkoasuja on muokattu tyylitiedoston avulla.

Sivuston kiinteä sisältö koostuu kolmesta erilaisesta sivupohjasta: etusivun sivupohjasta, sisältösivun sivupohjasta ja yhteystietosivun sivupohjasta. Sisältösivun ja yhteystietosivun sivupohjat ovat hyvin tavanomaisia, niissä on headerin ja footerin välissä kaksi MultipleContentBlocks-lisäosan avulla toteutettua sisältöpalstaa vierekkäin aseteltuna. MultipleContentBlocks mahdollistaa useampien eri sisältöeditorien kautta muokattavien palstojen sisällyttämisen samaan sivupohjaan. Tämä tekniikka mahdollistaa sen, että sivuston sisällönsyöttäjä voi vapaasti sijoittaa haluamaansa sisältöä kumpaankin palstaan alkuperäisen suunnitelman sijasta, jossa toinen palsta olisi varattu pelkästään kuville ja toinen palsta tekstisisällölle. Kuviossa 4 on esitelty MultipleContentBlocks-lisäosan kaksi sisältöeditoria, joista muokataan vierekkäisten pal-

tojen tekstisisältöjä. Näiden palstojen asemointi ja sisältösivun sivupohjan rakenne on esitelty kuviossa 5.

**Edit Page** [Add New](#)

**Esimerkkiartikkeli**

Permalink: <http://domain.fi/esim/> [Edit](#)

Upload/Insert Visual HTML

**b** *i* [link](#) **b-quote** ~~del~~ **ins** **img** **ul** **ol** **li** **code** **more** **lookup**

[close tags](#) [fullscreen](#)

Vasemman palstan tekstisisältö

Word count: 2      Draft saved at 12:52:29 am. Last edited by Admin on January 23, 2013 at 10:26 am

---

**Multiple content blocks**

**Rightcolumn**

Upload/Insert Visual HTML

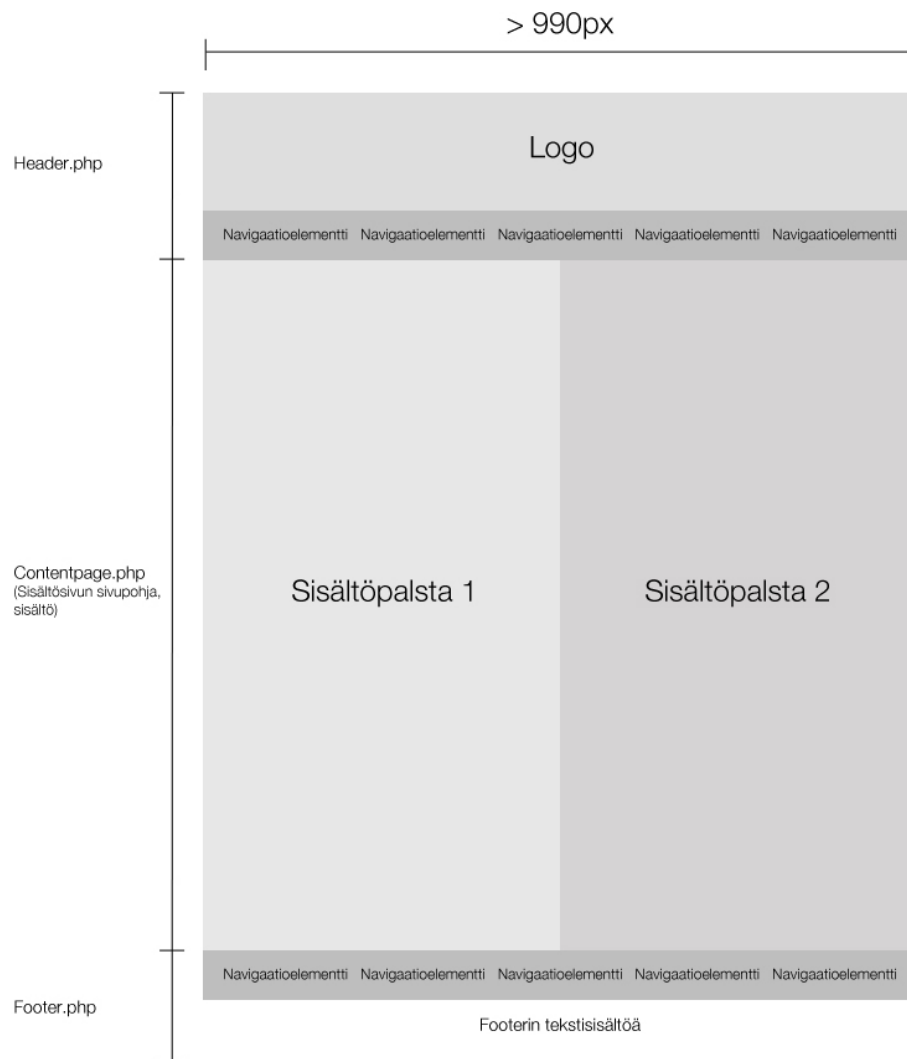
**b** *i* [link](#) **b-quote** ~~del~~ **ins** **img** **ul** **ol** **li** **code** **more** **lookup**

[close tags](#)

Oikean palstan tekstisisältö

KUVIO 4. Multiple Content Blocks-lisäosan kaksi itsenäistä sisältöeditoria





KUVIO 5. Sisältösivun sivupohja

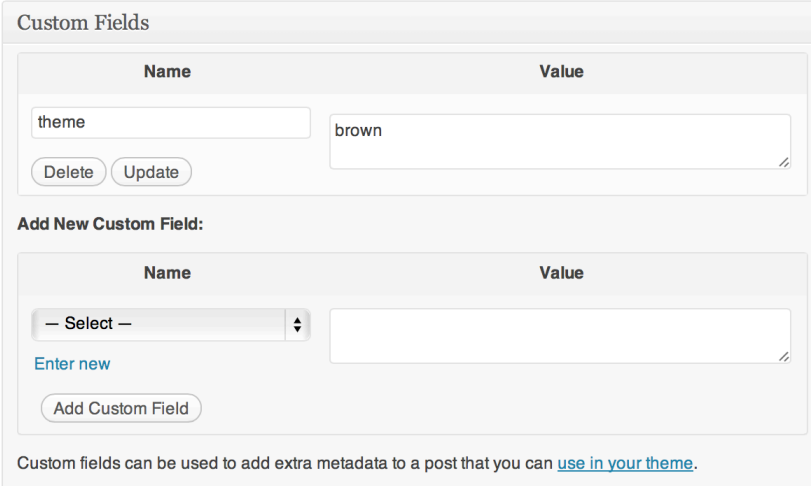
Yhteysivun sivupohja eroaa sisältösivusta Googlen karttalisäosan puolesta. Näiden sivupohjien tekstisisällön, sekä headerin sisällön väri vaihtuu sivulle asetettavan väriteeman myötä. Näitä väriteemoja käytetään indikaattorina sivuston neljälle pääosalle. Väriteemat on määritetty sisältösivuille käyttämällä WordPressin omaa Custom Field-työkalua, jolla sivulle voidaan määrittää sivupohjalle lähetettävä parametri, tässä tapauksessa väriteeman nimi. Tämä toiminnallisuus on esitelty kuviossa 6. Väriteema sisällytetään body-elementille alla olevalla koodilla.

```

<?php
$color = get_post_meta($postId, 'theme', 'true')
$parentColor = get_post_meta($post->post_parent, 'theme', 'true');

    if(!empty($color))
    {
        $colorTheme = $color.'-theme';
    }
    else if (!empty($parentColor))
    {
        $colorTheme = $parentColor.'-theme';
    }
    else
    {
        $colorTheme = '';
    }
?>
<body <?php body_class(); ?>>

```



Custom Fields

Name	Value
theme	brown

Delete Update

**Add New Custom Field:**

Name	Value
- Select -	

Enter new

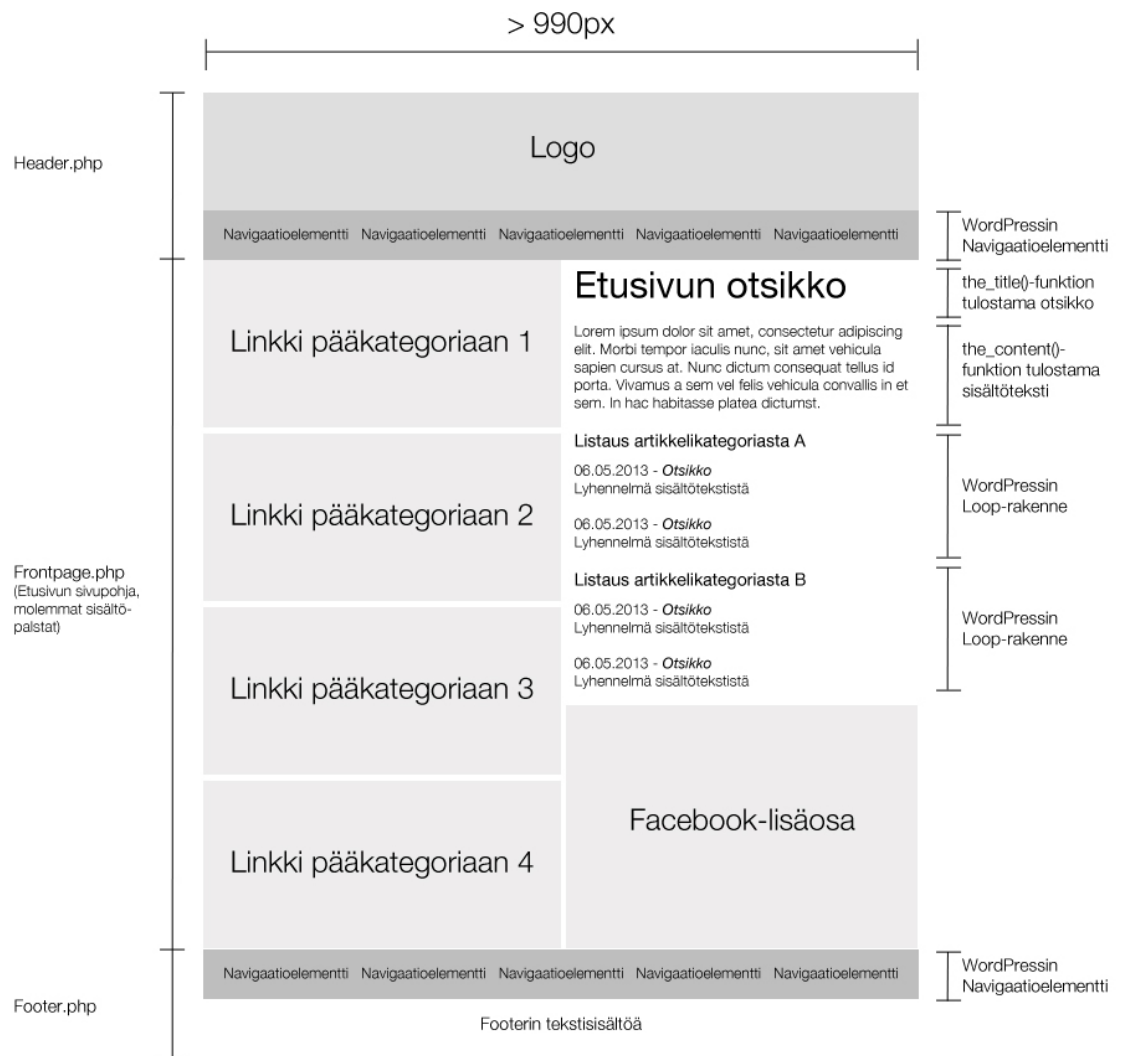
Add Custom Field

Custom fields can be used to add extra metadata to a post that you can [use in your theme](#).

## KUVIO 6. Väriteeman määrittäminen Custom Fieldin avulla

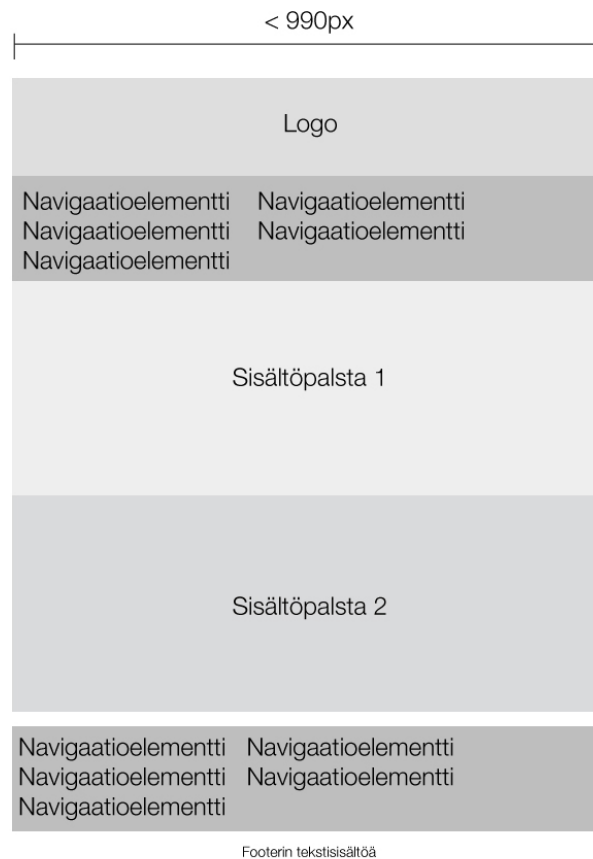
Etusivun sivupohja eroaa kahdesta muusta sivupohjasta merkittävästi. Etusivulla on suurikokoiset kuvina toteutetut linkit sivuston neljään pääosioon, jotka on sävytetty alisivujen väriteemojen mukaan, joihin linkit viittaavat. Linkeille on myös lisätty hover-tyyli, joka poistaa kuvien sävytyksen kun hiiren osoitin viedään linkin päälle. Sivupohjan oikeassa reunassa on sisältöpalstan tyylinen palsta jossa on kaikki muu etusivun sisältö, kuten otsikko, molempien sivuston artikkelikategorioiden listaukset, sekä sisältötekstipalsta. Otsikon alla on JavaScriptillä tehtyjen efektien avulla aukeava vi-

deosoitin, joka sisältää sivustolle suunnitellun videologon. Videosoitin on YouTube-videopalvelun tarjoama embed-elementti, joka sisällytetään sivupohjaan iframe-tekniikalla. Palstan pohjalla on Facebookin Like box-lisäosa, joka YouTube-soittimen tavoin sisällytetään iframella. Like box sisältää sivuston tilaajan tuoreimmat Facebook-päivitykset, sekä listauksen sivuston vierailijan kavereista jotka ovat tykänneet sivuston Facebook-sivusta. Kuten muutkin sivuston alisivut on etusivun sisältö sijoitettu headerin ja footerin väliin. Kuviossa 7 on näkyvillä etusivun sivupohjan rakenne.



KUVIO 7. Etusivun sivupohjan rakenne

Sivuston responsiivisuus on toteutettu tunnistamalla selaimen leveys media query-tekniikalla ja tekemällä tablettitietokoneen vaakanäkymää kapeammille näytöille omat käyttöliittymät. Selaimen leveyksien kynnykset, jotka aiheuttavat muutoksia käyttöliittymässä ovat sivuston normaalia leveyttä 990 pikseliä pienemmät näytöt, tablettitietokoneen pystyasento 768 pikseliä, älypuhelimien vaaka-asento 480 pikseliä sekä älypuhelimien pystyasento 320 pikseliä. Tablettitietokoneen pystyasennossa sivuston sisältöelementtejä on skaalattu hieman pienemmäksi sekä osa vierekkäisiin palstoihin sijoitetuista elementeistä on asemoitu päällekkäin. Alla olevassa kuviossa 8 on havainnollistettu sisältösivun sisältöelementtien uudelleensijoittelua tilanteissa, joissa selain on kapeampi kuin sivuston normaali leveys 990px.



KUVIO 8. Responsiiviseksi optimoitu sisältösivu

Älypuhelimien tapauksessa kaikki elementit on asemoitu päällekkäin, sekä kuvituksen ja muiden sisältöelementtien kokoa on pienennetty. Sivuston responsiivisuudessa on

huomioitu myös näytöt, joissa on korkea pikselitiheys, lataamalla kaikki kuvana käytettävät graafiset ulkoasuelementit kaksi kertaa näytettävää kuvaa suuremmalla resoluutiolla. Hallintaosion responsiivisuus oli valmiiksi WordPressin perustoiminnallisuuksissa mukana, joten sille ei ollut tarvetta toteuttaa.

#### 4.6 Tulokset

Sivuston toteuttaminen aloitettiin syyskuussa 2012 ja sivusto julkaistiin 1.11.2012 ohjaamalla sivuston liikenne vanhasta sivustosta uudelle sivustolle. Sivusto on julkaisusta lähtien ollut julki sekä jatkuvan pienimuotoisen jatkokehityksen alla. Sivuston suunnitelluista ominaisuuksista toteutettiin kaikki ja vain blogiosion kommentointitoiminnallisuus on otettu pois käytöstä roskapostin merkittävän määrän takia. Sivusto on toiminut vakaasti niin sisällönhallintajärjestelmän kuin palvelintilankin puolesta.

Sisältöhallintajärjestelmän valinta onnistui hyvin verkkosivustolle haluttuihin ominaisuuksiin ja niiden käytettävyyteen nähden. Sivuston kehitystyö tapahtui hyvin nopeasti ja ilman solmukohtia johtuen WordPressin hyvästä kehittäjille suunnatusta dokumentaatiosta. Myös lisäosat oli hyvin dokumentoituja ja niiden käyttöönotto onnistui hyvin. Kaikki ominaisuudet saatiin helposti hallittaviksi hallintapaneelin kautta. Mitään ominaisuutta ei tarvinnut tehdä kiinteäksi siten että muokkaus onnistuisi vain muokkaamalla käsin koodia. Myös teeman rakentaminen toteutui WordPressin tapauksessa siten, että sen rakenne on johdonmukainen ja toiminnot järkevästi toteutettu. Tämä mahdollistaa helpon jatkokehityksen, sekä tarvittaessa teeman kopioinnin toiseen WordPress-asennukseen muokkausta varten, jos projektissa on esimerkiksi käytetty runsaasti samoja elementtejä.

Responsiivinen suunnittelu ja toteutus onnistui myös hyvin. Sivusto on hyvin käytettävissä kaikilla yleisimmillä päätelaitteilla, suuria näyttökokoja unohtamatta. Älypuhelimien näytölle optimoituja versioita muutettiin hieman typografian ensimmäisestä julkaisusta versiosta käytettävyyden parantamiseksi. Myös erikoisfonttien käyttö Typetitillä oli hyvä ratkaisu, käytössä olevan lisenssin riittäessä sivun näyttökertoihin.

Typekitillä toteutettuna fontit ovat myös hyvin yhteensopivia eri päätelaitteiden ja selaimien kesken ja latautuvat kevyesti.

## 5 POHDINTA

Opinnäytetyön yksi tärkeimmistä tutkinnan kohteista oli pysyvien ratkaisujen löytäminen erikoisominaisuuksien toteuttamiseen vastaavanlaisissa sivustohankkeissa. Ratkaisujen onnistuminen laajassa julkaistavassa toteutuksessa eivät takaa kuitenkaan kannattavuutta niiden käyttöön jatkossa. Monessa toteutuksessa lisäosan kannattava käyttö saattaa ilmetä esimerkiksi siten, että lisäosa on sopinut sivuston kehitystyöhön perehtymisen ja käyttöönoton osalta pelkän aikataulupaineen vuoksi. Tässä vaiheessa esimerkiksi lisäosan uusien versioiden hidas kehitystyö sisällönhallinnan uuden version julkaisun jälkeen tai huono muokattavuus saattaa jäädä kokonaan huomioimatta.

Erikoisfonttien osalta Typekitin tai Fonts.comin webfonttipalvelun käyttäminen pysyvänä ratkaisuna on äärimmäisen hyvä. Se on erittäin riippumaton käytettävästä alustasta ja selaimesta, toistuen samoin tavoin kaikilla laitteilla. Näiden palvelujen käyttäminen on toisaalta turhaa sivustoprojekteissa, jonka asiakas omistaa jo valmiiksi käyttöoikeudet erikoisfonttiin. Myös Typekitin ja Fonts.comin halvimmat, noin 50€ vuodessa maksavat lisenssit saattavat myös hämätä hinnallaan, näyttäen hyvin kalliilta vaihtoehdolta. Todellisuudessa ne ovat lähes halvimpia ratkaisuja, yksittäisten fonttien käyttöoikeuksien ollessa hyvinkin kalliita. Tilanteessa, jossa haluttua fonttia ei löydy niitä tarjoavista palveluista, tulee harkita käytettävän fontin vaihtamista tai tekstien kääntämistä kuviksi Cufónin tai muiden vanhanaikaisten tekniikoiden sijasta.

Responsiivinen suunnittelu elää jatkuvasti ja menetelmät sen toteuttamiseen kehittyvät. Kuitenkin yleisimmät käytettävät menetelmät responsiivisuuden toteutukseen ovat hyvin vanhoja CSS-tekniikoita, kuten mittojen määrittäminen prosentteina. Nämä takaavat hyvän toimivuuden useimmissa vanhemmissakin selaimissa mutta tarjoavat monesti liian helpon keinon karsia ulkoasu vain skaalautuvaksi tekstisisällöksi mobiililaitteille. Hyvä responsiivinen toteutus koostuu kuitenkin pääosin hyvästä ideasta, sekä sisällön priorisoinnista. Suunnittelijan on päätettävä, halutaanko esille nostaa sivuston sisällön tärkeimpiä osia, kuten logoa, yhteystietoja ja linkityksiä sosiaali-

seen mediaan korostaa vai halutaanko panostaa enemmän visuaaliseen toteutukseen.

Kieliversiointi ideana on erittäin käytännöllinen sisällön monipuolisuutta ja sivuston kävijäkuntaa ajatellen. Myös sille tarjotut lisäosat, kuten WPML ovat pitkälle kehitetyjä ja hyviä ratkaisuja tämän ominaisuuden toteuttamiseen. Kieliversiointi pohjautuu kuitenkin suurimmaksi osaksi sisällön syöttämiseen sekä sivuston sisällön monipuolisuuden toissijaisten kieliversioiden osalta. Jos kieliversioita varten ei synny sisältöä, voi olla parempi vaihtoehto pitää puolityhjät kieliversiot kokonaan pois näkyvistä ja tehdä sivustolle halutulla kielellä tiivistelmä kieliversiointin sijasta. Tietysti sisällöltään lähes identtinen kieliversiointi toimii erittäin hyvin sivuston kävijäkunnan kasvattamisessa.

WordPress osoittautui myös hyväksi valinnaksi erikoisominaisuuksia toteuttaessa. WordPress ei pelkästään ole järkevästi rakennettu yksinkertainen toteutusalue näille ominaisuuksille, vaan loistaa erityisesti alustana jolla on laaja kehittäjäkunta ja runsaasti mahdollisuuksia ratkaista ominaisuudet monella eri tavalla. Järjestelmän joustavuus vaikuttaa etenkin tapaan, millä ominaisuuksia hallitaan hallintapaneelin kautta. Hallintatoiminnallisuudet tulisi olla sisällönsyöttäjälle mahdollisimman vaivattomia ja selkeitä, sillä sivustoissa joita asiakas itse hallinnoi, tulee pyrkiä mahdollisimman palvelumaiseen toteutukseen. Tästä esimerkkinä kieliversiointi, joka olemassaolollaan muistuttaa esimerkiksi sisältösivua muokatessa kieliversioiden tilasta ja olemassaolosta. Jos kieliversio on luotu, näytetään muokkausnäkyvässä linkit joista kieliversioita pääsee vaivattomasti muokkaamaan. Jos versioita ei ole luotu, muistutetaan käyttäjää että versiot ovat vielä tekemättä.

Hankkeessa käytetty All In One SEO-pack taas muistuttaa käyttäjää miettimään, miten muokattavan artikkelin haluaisi nähtävän hakukoneen silmin. Mainosalalle kriittinen hakukonenäkyvyys kärsii, jos sivustolla ainoat hakukoneoptimointiin tarkoitettut metatiedot ovat kiinteästi asetettu ja ne kuvailevat ympäröivästä ja lyhyesti koko sivun yleisluonnetta. Tästä syystä modernin sivuston tulee olla laajalti ja helposti sisäl-



lönsyöttäjän muokattavissa, sillä sivuston kehittäjällä on harvoin yhtä laajaa näkemystä sivuston sisällöstä rakennusvaiheessa. Näiden muokkaustoimintojen laajuudesta, hyvästä dokumentoinnista ja erityisesti helppokäyttöisyydestä johtuen WordPress on erittäin hyvä valinta myös jatkossakin mainosalan verkkosivutoteutuksiin.

Hyvät ratkaisut kierrätetään monesti toisiin projekteihin sellaisena ja muokataan riittämään uuden projektin vaatimuksiin. Tässä tilanteessa käytetään hyväksi jo aikaisemmin tehtyä työtä. Kierrättämisen vaarana on oikoen toteutettujen pienempien räätälöintien leviäminen valmiina ratkaisuina esimerkiksi tekijältä toiselle. WordPressin osalta jatkokehityksen ja kestävien ratkaisujen osalta yksi merkittävimmistä asioista on teemarakenne. Teemarakenne kokoaa yhteen kaikki toiminnallisuuksien muokkaukset ylikirjoittamalla ja kaikki sivuston rakenteen osat. Teema auttaa siis löytämään nämä muokkaukset helposti ja nopeasti tilanteessa, jossa vaaditaan nopeita muokkauksia sivustolle. Vaihtoehtona olisi järjestelmän toimintojen suora muokkaaminen lähdekooditiedostoihin. Tämä ratkaisu ei ole kestävä eikä vakaa. Jo pelkästään sisällönhallintajärjestelmän päivittäminen ylikirjoittaisi kaikki tehdyt muutokset. Järkevästi tehtyä teemaa voi käyttää pienin muokkauksin tai käyttää pelkästään esimerkkinä uusien toiminnallisuuksien luomisessa toiseen verkkosivustoprojektiin. WordPressin teeman rakenne mahdollistaa myös lisäosien muokkaamisen erillään niiden lähdekoodista. Täten myös lisäosat ovat päivitettävissä ilman muokkauksien ylikirjoittamista.

Tietysti melko yksinkertaisena työkaluna WordPress ei ole soveltuva kaikkiin toteutuksiin, kuten esimerkiksi laajamittaiseen datan käsittelyyn tai toisella tekniikalla tehdyn sovelluksen suoraan porttaamiseen selainversioksi. Näitä toteutuksia varten voidaan käyttää paremmin soveltuvia modulaarisia sisällönhallintoja kuten Drupalia tai jopa laajempia frameworkoja alustana. Monipuolisena alustana WordPress antaa kuitenkin hyvät puitteet kehittäjille luoda vapaasti haluamiaan ominaisuuksia kestävinä ratkaisuina. WordPress pitää kuitenkin rakenteillaan huolta siitä, että ratkaisut tehdään oikeaoppisesti. Mainosalan toteutusten painottuessa kuitenkin visualisuu-

teen, lyhyisiin sanomiin, sekä kohtuullisen yksinkertaiseen sisältöön, on WordPress myös jatkossa suositeltava alusta näihin tarkoituksiin.

## LÄHTEET

About. 2013. WordPressin yleiskatsaus WordPressin omilla sivuilla. Viitattu 4.4.2013.  
<http://wordpress.org/about/>

About Cufón. 2012. Cufónin kehittäjän ohjeistussivu tekniikan käytöstä. Viitattu 29.3.2013. <https://github.com/sorccu/cufon/wiki/about>

Create A Network. 2013. WordPress Codexin dokumentaatio WordPress Multi Site:n käytöstä. Viitattu 17.4.2013. [http://codex.wordpress.org/Create\\_A\\_Network](http://codex.wordpress.org/Create_A_Network)

Creating Your Own Page Templates. 2013. Wordpress Codexin dokumentointi loopin käytöstä. Viitattu 10.4.2013.  
[http://codex.wordpress.org/Pages#Creating\\_Your\\_Own\\_Page\\_Templates](http://codex.wordpress.org/Pages#Creating_Your_Own_Page_Templates)

CSS3 @font-face Rule. 2013. World Wide Web Consortiumin ohjeistus @font-face-tekniikan käytöstä. Viitattu 29.3.2013.  
[http://www.w3schools.com/cssref/css3\\_pr\\_font-face\\_rule.asp](http://www.w3schools.com/cssref/css3_pr_font-face_rule.asp)

CSS Media Types. 2013. World Wide Web Consortiumin ohjeistus @font-face-tekniikan käytöstä. Viitattu 14.4.2013.  
[http://www.w3schools.com/css/css\\_mediatypes.asp](http://www.w3schools.com/css/css_mediatypes.asp)

CSS Web Safe Font Combinations. 2013. World Wide Web Consortiumin ohjeistus websafe-fonttien käytöstä. Viitattu 29.3.2013  
[http://www.w3schools.com/cssref/css\\_websafe\\_fonts.asp](http://www.w3schools.com/cssref/css_websafe_fonts.asp)

Davidson. Mike. Introducing sIFR. 2004. Esittelyteksti sIFR-tekniikasta. Viitattu 29.3.2013. <http://www.mikeindustries.com/blog/archive/2004/08/sifr>

Drupal-Joomla-Wordpress-Comparison. 2013. Amerikkalaisen DCKAP-konsultointiyrityksen vertailu sisällönhallintajärjestelmistä. Viitattu 4.4.2013.  
<http://www.dckap.com/drupal-joomla-wordpress-comparison.htm>

Features. 2012. WPML-lisäosan esittelysivu kehittäjän sivustolla. Viitattu 17.4.2013:  
<http://wpml.org/features/>

Features overview. 2013. Joomla!-n esittelyteksti sisällönhallinnan ominaisuuksista. Viitattu 4.4.2013. <http://www.joomla.org/core-features.html>

History. 2013. WordPressin historiaa WordPressin teknisessä dokumentaatiossa. Viitattu 4.4.2013. <http://codex.wordpress.org/History>

History of Drupal. 2013. Historiakuvaus Drupalin sivustolla. Viitattu 4.4.2013. <http://drupal.org/about/history>

Introduction to web typography and @font-face. 2011. Adoben artikkeli @font-face-tekniikan käytöstä ja web-typografiasta. Viitattu 29.3.2013. <http://www.adobe.com/devnet/html5/articles/web-typography-and-css-font-face.html>

Johnson, J. n.d. The Essential Guide to @font-face. Ohjeistus @font-facen käyttämiseen ja mahdollisuuksiin. Viitattu 29.3.2013. <http://sixrevisions.com/css/font-face-guide/>

Mobile vs. Desktop. 2013. Amerikkalaisen tilastoyrityksen tilasto mobiiliselaisten ja desktop-selaisten jakaumasta. Viitattu 10.4.2013. [http://gs.statcounter.com/#mobile\\_vs\\_desktop-ww-monthly-201203-201303](http://gs.statcounter.com/#mobile_vs_desktop-ww-monthly-201203-201303)

Patel, S. 2012. Simple Steps to Proper Mobile Optimization. Artikkeliverkkosivujen mobiilioptimoinnista. Viitattu 14.4.2013. <http://www.searchenginejournal.com/simple-steps-to-proper-mobile-optimization/54153/>

Pettit, N. 2012. Beginners Guide to Responsive Web Design. Artikkeliverkkosivustojen responsiivisesta ulkoasuunnittelusta. Viitattu 14.4.2013. <http://blog.teamtreehouse.com/beginners-guide-to-responsive-web-design>

PHP-Ohjelmointi. n.d. Nebulan ohjeistus PHP-versioasetuksien käyttöön webhotelleissaan. Viitattu 4.4.2013. <https://www.nebula.fi/fi/node/218>

Platform. 2012. Joomla! Platformin esittelysivu dokumentaationsivuilla. Viitattu 4.4.2013. <http://docs.joomla.org/Platform>

Pricing. 2013. Typekitin hinnoittelu. Viitattu 4.4.2013 <https://typekit.com/plans>

Purchase. 2012. WPML-lisäosien hinnoittelu. Viitattu 17.4.2013. <http://wpml.org/purchase/>

qTranslate. 2013. qTranslate-lisäosan kehittäjän esittelysivu lisäosasta. Viitattu 17.4.2013. <http://www.qianqin.de/qtranslate/>

Requirements. 2013. Teknisten vaatimusten esittely WordPressin sivuilla. Viitattu 4.4.2013. <http://wordpress.org/about/requirements/>

Stepping Into Template Tags. 2013. Wordpress Codexin dokumentointi template tagien käytöstä. Viitattu 10.4.2013

[https://codex.wordpress.org/Stepping\\_Into\\_Template\\_Tags](https://codex.wordpress.org/Stepping_Into_Template_Tags)

Technical Requirements. 2013. Joomla!:n teknisten vaatimuksien esittelysivu. Viitattu 4.4.2013. <http://www.joomla.org/technical-requirements.html>

The Loop. 2013. Wordpress Codexin dokumentointi loopin käytöstä. Viitattu 10.4.2013 [http://codex.wordpress.org/The\\_Loop](http://codex.wordpress.org/The_Loop)

Theme Review. 2013. Wordpress Codexin dokumentointi teeman käytön perusteista. Viitattu 9.4.2013. [http://codex.wordpress.org/Theme\\_Review](http://codex.wordpress.org/Theme_Review)

Webhotellit. 2013. Webhotellipakettien esittely Nebulan sivuilla. Viitattu 29.3.2013 <http://www.nebula.fi/fi/palvelut/pilvipalvelut/webhotellit>

What is Joomla?. 2013. Joomla!:n yleinen esittelysivu. Viitattu 4.4.2013:<http://www.joomla.org/about-joomla.html>

Wilding, R. 2011. Wordpress vs. Joomla vs. Drupal. A Comparison of CMS Solutions. Artikkelisiällönhallintajärjestelmien vertailusta. Viitattu 4.4.2013. <http://www.socialtechnologyreview.com/articles/wordpress-vs-joomla-vs-drupal-comparison-cms-solutions>

WPML Core and Add-on Plugins. 2012. WPML-perheen lisäosien esittelyt kehittäjän sivustolla. Viitattu 17.4.2013. <http://wpml.org/documentation/wpml-core-and-add-on-plugins/>