

Hakukoneoptimointi uuden sisällön- hallintaohjelmiston käyttöönotossa

Aatu Pitkänen

Opinnäytetyö

Lokakuu 2016

Tekniikan ja liikenteen ala

Insinööri (AMK), Ohjelmistotekniikan koulutusohjelma

Tekijä(t) Pitkänen Aatu	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä 11.10.2016
	Sivumäärä 43	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x
Työn nimi Hakukoneoptimointi uuden sisällönhallintaohjelmiston käyttöönotossa		
Tutkinto-ohjelma Ohjelmistotekniikan koulutusohjelma		
Työn ohjaaja(t) Rantala Ari, Hämäläinen Raija		
Toimeksiantaja(t) Suomen Digimarkkinointi Oy		
Tiivistelmä <p>Opinnäytetyön toimeksiantajana toimi Suomen Digimarkkinointi Oy. Tavoitteena oli rakentaa hakukoneoptimoidut, uuden brändi-ilmeen mukaiset verkkosivut, jonne tuotiin sisällöt vanhan sivuston tietokannasta.</p> <p>Toteutus aloitettiin vertailemalla sisällönhallintaohjelmistoja ja valitsemalla niistä sopivin toteutukseen. Kehitysympäristönä verkkosivuston rakentamista varten toimi Vagrant ja versionhallintana käytettiin git-versionhallintaohjelmistoa.</p> <p>Työssä käsitellään hakukoneoptimointia ja erilaisia avoimen lähdekoodin sisällönhallintaohjelmistoja. Lisäksi käydään läpi työn toteutukseen käytettyjä erilaisia työkaluja ja teknologioita.</p> <p>Lisäksi opinnäytetyössä eritellään verkkosivuston suunnittelun ja toteutuksen eri vaiheita, hakukoneoptimointia sekä sisältöjen siirtämistä tietokannasta toiseen eri sisällönhallintaohjelmistojen välillä. Myös Wordpress-teeman kehitystä käsitellään.</p> <p>Lopputuloksena saatiin uudet hakukoneoptimoidut, yrityksen tarpeisiin räätälöidyt verkkosivut. Aikataulusta myöhästettiin pari viikkoa, mutta sivuston valmistuttua siitä löytyi kaikki vaaditut ominaisuudet. Palvelusivut ja blogi löytyvät korkealta Googlen ilmaisissa hakutuloksissa, ja tehtyjen nopeustestien perusteella sivut myös latautuvat keskimäärin nopeammin kuin kilpailijoilla. Sivustoa on jatkossa helppo ylläpitää ja sinne on helppo lisätä sekä uusia ominaisuuksia että laskeutumissivuja.</p>		
Avainsanat (asiasanat) PHP, CSS, JQuery, MySQL, Hakukoneoptimointi, sisällönhallintaohjelmisto		
Muut tiedot		

Author(s) Pitkänen Aatu	Type of publication Bachelor's thesis	Date 11.10.2016 Language of publication:
	Number of pages 43	Permission for web publication: x
Title of publication Search engine optimization in deployment of content management system		
Degree programme Software Engineering		
Supervisor(s) Rantala Ari, Hämäläinen Raija		
Assigned by Suomen Digimarkkinointi Oy		
Abstract <p>The purpose of the thesis was to make a new search engine optimized website that follows the company's new brand guidelines. The initial requirement was to import data from the database of the old website into the new website. This project was assigned by Suomen Digimarkkinointi Ltd which is a digital marketing company.</p> <p>The implementation started with studying the website content management systems, comparing them with each other and choosing the right one for this thesis. Vagrant development environment was used for this project.</p> <p>The theoretical section focuses on fundamentals of search engine optimization, different open-source content management systems and the development tools used in the thesis.</p> <p>The implementation section of the thesis covers different stages of website design and development, search engine optimization and importing content from one content management system to another. Wordpress theme development is also covered in the thesis.</p> <p>The result of the thesis is a search engine optimized website with all of the data imported from another content management system. Even though the project finished two weeks later than originally scheduled, it met all the initial requirements and had all planned functionality. The site is very scalable, therefore maintaining it and adding new functionality in the future will be easy.</p>		
Keywords/tags (subjects) PHP, CSS, JQuery, MySQL, Search engine optimization, Content management system		
Miscellaneous		

Sisällys

Käsitteet	3
1 Lähtökohdat.....	5
1.1 Toimeksiantaja: Suomen Digimarkkinointi Oy	5
1.2 Tavoitteet	5
2 Sisällönhallintaohjelmistot	6
2.1 Yleistä	6
2.2 Wordpress	7
2.3 Drupal	7
2.4 Joomla	7
2.5 Valinta.....	8
3 Työkalut ja teknologiat.....	9
3.1 Asana	9
3.2 Basecamp	9
3.3 Bitbucket ja Git	9
3.4 Kehitysympäristö Vagrant	9
3.5 MySQL Workbench.....	10
3.6 JQuery.....	10
4 Hakukoneoptimointi ja konversio	11
4.1 Yleistä	11
4.2 Sivuston sisäiset tekijät	13
4.2.1 Yleistä.....	13
4.2.2 Yksilölliset otsikot	13
4.2.3 Laadukas sisältö	13
4.2.4 Metatiedot.....	14
4.2.5 Selkokielliset URL-osoitteet.....	14
4.2.6 Mobiiliystävällisyys	14

	2
4.2.7	Verkkosivuston latautumisenopeus 15
4.2.8	Sisäiset linkit 15
4.2.9	Sivustokartta 15
4.3	Sivuston ulkoiset tekijät 16
5	Verkkosivuston toteutus 16
5.1	Suunnittelu ja käytettävyys 16
5.2	Kehitysympäristö 17
5.3	Layout 17
5.4	Sisällön tuonti 18
5.4.1	tausta 18
5.4.2	toteutus 18
5.5	Kävijäseuranta 19
5.6	Sisältötyypit 19
5.7	Teeman kehitys 20
5.8	Sivuston nopeuden optimointi 21
5.9	Muu räätälöinti 22
6	Tulokset 24
7	Johtopäätökset 27
Lähteet 29
Liitteet 31
Liite 1.	Sivuston Gzip-pakkaukseen käytetty koodi 31
Liite 2.	Sivuston Expire headerien lisäykseen käytetty koodi 32
Liite 3.	Alapalveluiden listaaminen ja lisääminen shortcodeen 33
Liite 4.	Palvelu-sisältötyypille oman kategorian rekisteröinti 34
Liite 5.	Uuden sisältötyypin rekisteröinti 35
Liite 6.	Etusivun rautalankamalli 36
Liite 7.	Blogi-sivun rautalankamalli 37
Liite 8.	Yksittäisen blogin rautalankamalli 38
Liite 9.	Drupalin blogikirjoitusten siirto Wordpressintietokantaan 39
Liite 10.	Käyttäjien haku Drupalintietokannasta 40

Käsitteet

Apache

Apache HTTP Server on avoimeen lähdekoodiin perustuva HTTP-palvelinohjelma.

API

Application program interface

CTA

Toimintakehotus eli call to action -napit ohjaavat vierailijaa sivuston omistajan haluamaan suuntaan.

CSS

Cascadian Style Sheets on erityisesti WWW-dokumenteille kehitetty tyyliohjeiden laji.

Git

Git on avoimeen lähdekoodiin versionhallintaohjelmisto.

FTP

FTP eli File Transfer Protocol. Se on tiedonsiirtomenetelmä asiakas ja palvelinkoneiden välillä.

HTML

Hypertext Markup Language on verkkosivuilla käytetty markup kieli.

Javascript

JavaScript on pääasiassa [Web](#)-ympäristössä käytettävä dynaaminen [komentosarjakieli](#)

Sisällönhallintaohjelmisto (CMS)

Sisällönhallintaohjelmisto on verkkosivujen tekemiseen ja ylläpitoon tarkoitettu selainpohjainen ohjelmisto.

MYSQL

MYSQL on suosittu relaatiotietokantaohjelmisto.

PHP

PHP (Hypertext Preprocessor) on palvelinpuolella käytetty ohjelmointikieli.

Plugin

Lisäosa, jolla voidaan laajentaa Wordpressin toiminnallisuutta ja ominaisuuksia.

Shortcode

Hakasulkeiden välissä oleva teksti, jonka avulla Wordpressin eri sisältötyyppien, kuten sivujen, artikkeleiden ja vimpaimien sisällä voidaan näyttää monimutkaista sisältöä kirjoittamatta koodia suoraan kyseiseen osioon.

1 Lähtökohdat

1.1 Toimeksiantaja: Suomen Digimarkkinointi Oy

Suomen Digimarkkinointi Oy on vuonna 2012 perustettu jyvaskyläläinen yritys. Sen toimiala on digitaalinen markkinointi ja yrityksen palveluihin kuuluvat Google Ad-words-mainonta, hakukoneoptimointi sekä Facebook-markkinointi. Myös sähköpostimarkkinointi ja kotisivujen tekeminen kuuluvat yrityksen tarjoamiin palveluihin. Suomen Digimarkkinointi Oy:llä oli 12 työntekijää työn aloitushetkellä keväällä 2016.

1.2 Tavoitteet

Selkeiden tavoitteiden asettaminen jo alkuvaiheessa oli tärkeää hyvän lopputuloksen aikaan saamiseksi. Lähtökohtana oli se, että yrityksen brändiä ja graafista ilmettä aloitettiin uudistamaan ja samalla syntyi myös tarve uusille verkkosivuille. Vanhojen verkkosivujen toteutus ei vastannut käyttötarkoitusta: niiden ulkoasu oli vanhanlainen ja Drupal-julkaisujärjestelmä turhan monimutkainen peruskäyttäjälle. Yrityksen blogiin oli neljän vuoden aikana kirjoitettu kaksi kertaa viikossa, joten blogissa oli satoja kirjoituksia. Toimeksiantaja halusi säilyttää nämä blogikirjoitukset ja siirtää ne uusille verkkosivuille. Uusien verkkosivujen haluttiin löytyvän mahdollisimman korkealta Googlen ilmaisissa eli orgaanisissa hakutuloksissa verkkosivustolle tulevan liikenteen kasvattamiseksi. Korkea sijoitus hakutuloksissa on myös erityisen tärkeää jo senkin takia, että hakukoneoptimointi kuuluu yrityksen tarjoamiin palveluihin.

Tavoitteena oli parantaa sivuston suorituskykyä, responsiivisuutta ja hakukoneystävällisyyttä vanhaan sivustoon verrattuna. Varsinkin mobiiliversion toimivuus kaikin puolin nostettiin korkealle tavoite listalla, koska tulevaisuudessa mobiilikäyttäjien määrä tulee vain kasvamaan ja on jo ohittanut pöytäkonekäyttäjien määrän (Mobile Marketing statistics compilation 2016). Lisäksi toimeksiantaja antoi listan toteutettavista ominaisuuksista ja toiminnollisuuksista, joita ei löytynyt yrityksen vanhoilta verkkosivuilta. Alustavaksi aikatauluksi muodostui 16.05.-15.08.2016.

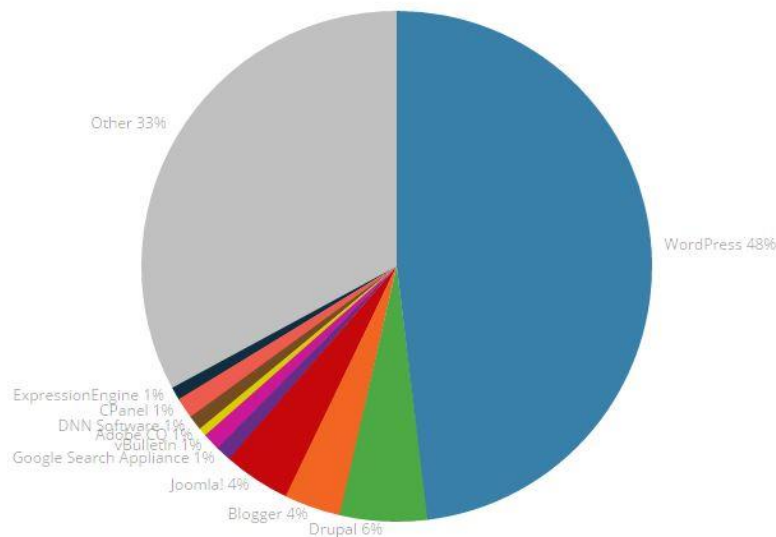
2 Sisällönhallintaohjelmistot

2.1 Yleistä

Verkkosivujen julkaisujärjestelmä, josta käytetään myös yleisesti nimitystä selainpohjainen sisällönhallintajärjestelmä (CMS), on verkkosivujen tekemiseen tarkoitettu selainpohjainen ohjelmisto. Julkaisujärjestelmää käyttäen pystytään tekemään kaikenkokoisia verkkosivustoja, yksinkertaisesta blogista aina suuriin yritysten ja organisaatioiden verkkosivustoihin asti. Kolme suosituinta sisällönhallintaohjelmistoa ovat Wordpress, Drupal ja Joomla (CMS Usage Statistics 2016) (ks. kuvio 1).

CMS Usage Statistics

Statistics for websites using CMS technologies



Kuvio 1. Julkaisujärjestelmien käyttö miljoonalla suosituimmalla verkkosivulla. (alkup. kuvio ks. CMS Usage Statistics 2016)

Kaikki kolme ovat ilmaisia ja avoimeen lähdekoodiin perustuvia GPL-lisenssin alaisuudessa. Julkaisujärjestelmiä yhdistää myös se, että ne on pääasiassa kirjoitettu PHP:llä ja käyttävät MySQL:ää tietokannan hallintaohjelmistona. Niistä löytyy kuitenkin ratkaisevia eroavaisuuksia, ja siksi olikin tärkeää tehdä oikea valinta parhaan lopputuloksen aikaansaamiseksi.

2.2 Wordpress

Wordpress on maailman suosituin selainpohjainen verkkosivujen sisällönhallintaohjelmisto. Se on parhaimmillaan pienten- ja keskisuurten sivustojen sekä blogien toteuttamiseen, koska se oli alun perin suunniteltu juuri tätä tarkoitusta varten.

Wordpress on kuitenkin kehittynyt valtavasti sen alkuajoista ja myös monet suuret sivustot on rakennettu Wordpress-sisällönhallintaohjelmistolla. Wordpress on haku-kone-ystävällinen, ja siihen on saatavilla loistavia lisäosia hakukonenäkyvyyden parantamiseksi. Wordpress-sivustoa on helppo ylläpitää ja muokata intuitiivisen käyttöliittymän ansiosta. Wordpressiin on saatavilla paljon lisäosia ja teemoja, joiden avulla voi laajentaa sivuston toiminnallisuutta ja muuttaa sen ulkonäköä vaivattomasti.

(Wordpress 2016) Omia lisäosia ja teemoja tekemällä sivustosta saa räätälöityä juuri omaan tarpeeseensa sopivan. Suuren suosion ja laajan käyttäjämäärän ansiosta Wordpressin takana on laaja tukiyhteisö, joten internetistä löytyy paljon materiaalia aiheeseen liittyen.

2.3 Drupal

Drupal on maailman toiseksi suosituin julkaisujärjestelmä. Suomen Digimarkkinointi Oy:n vanhat verkkosivut oli toteutettu Drupalilla. Sillä on muihin vaihtoehtoihin verrattuna jyrkempi oppimiskäyrä. Drupal sopii kuitenkin erityisen hyvin suurempien sivustojen tekemiseen, koska se skaalautuu muita vaihtoehtoja paremmin suuremmille sivustoille. Tietoturva on etenkin Drupalin vahvin osa-alue. Tämän takia esimerkiksi Valkoisen talon verkkosivut ovat tehty Drupalin avulla (White House now uses Drupal 2009). Drupal on erityisesti sovelluskehittäjien suosiossa sen suuren muokattavuuden ansiosta.

2.4 Joomla

Kolmanneksi suosituin sisällönhallintajärjestelmä Bloggerin ohella on Joomla (ks. Kuvio 1). Joomla on yleisesti käytössä erityisesti verkkokauppa-alustana, koska siinä on natiivi tuki verkkokauppasovelluksille. Verkkokaupan pystyttäminen onnistuu myös Drupalin ja Wordpressin avulla, mutta se vaatii enemmän aikaa ja on työläämpää. Joomlaista löytyy myös melko paljon lisäosia, mutta ne eivät ole yhtä kattavia kuin

Wordpressin lisäosat. Joomlaan tutustuttaessa siitä ei löytynyt mitään merkittäviä ominaisuuksia, joiden avulla se olisi selvästi erottunut edukseen muista vertailussa olevista järjestelmistä. Joomla-sivuston ylläpitäminen vaatii perusteknisten tietojen hallitsemisen. Joomla voisi sanoa olevan ominaisuuksiltaan ja teknisyydeltään jossain Drupalin ja Wordpressin välimaastossa.

2.5 Valinta

Kaikki kolme julkaisujärjestelmää ovat hyviä vaihtoehtoja useimmille verkkosivuprojekteillemme. Teknisessä osaamisessa tai toteutuksessa ei julkaisujärjestelmien välillä ole suuria eroja, sillä kaikkien taustakehityksessä käytetään PHP:tä ja MySQL-tietokantaa. Tämän opinnäytetyön tekijällä oli Wordpress-julkaisujärjestelmästä jo aiempaa kokemusta, joten sen valinta oli hyvin luonnollinen muihin vaihtoehtoihin verrattuna. Wordpressistä löytyy myös helpompi ylläpidettävyys ja käyttäjäystävällisempi hallintapaneeli muihin vertailussa oleviin julkaisujärjestelmiin verrattuna. Luokusten eri käyttäjäroolien ansiosta Wordpress-sivustolla pystytään helposti tekemään useita eri tasoisia käyttäjiä.

Yrityksen vanhoilla verkkosivuilla oli käytössä Drupal, mikä on hyvä suurempia sivustoja tehtäessä, mutta sen käyttö on kohtalaisen hankalaa uusille käyttäjille, eikä ole tarkoituksenmukainen pienelle yritykselle. Joomla taas tuntui lähinnä kompromissiratkaisulta Drupalin ja Wordpressin välillä.

Täydellistä sisällönhallintaohjelmistoa ei ole olemassa, joten päätöstä tehtäessä joutui tekemään pieniä kompromisseja. Haasteena oli valita helposti ylläpidettävä ohjelmisto, joka olisi ominaisuuksiltaan tarpeeksi riittävä ja joustava. Sisällönhallintaohjelmiston koodinmuokkaamis- ja ulkonäön muutosmahdollisuudet sekä kustannukset olivat myös merkittäviä tekijöitä valintaa tehtäessä.

Näistä kolmesta julkaisujärjestelmästä Wordpress täytti parhaiten yllä mainitut vaatimukset. Wordpressin eri käyttäjäroolit mahdollistavat sisällöntuotannon hajauttamisen koko yrityksen kesken ilman, että kaikille tarvitsisi antaa admin-oikeuksia. Wordpressin ylläpidettävyys ja hakukoneystävällisyys nousivat ratkaiseviksi tekijöiksi, joten toteutuksen tekemiseen valittiin Wordpress-sisällönhallintaohjelmisto.

3 Työkalut ja teknologiat

3.1 Asana

Asana on tehtävien organisoimiseen käytetty projektihallintajärjestelmä. Projektin aikana Asanaan merkattiin kaikki verkkosivujen tekemiseen liittyvät tehtävät ja niiden eri vaiheet.

3.2 Basecamp

Basecamp on selainpohjainen projektihallintaohjelmisto. Basecamp-projektissa mukana olevat henkilöt voivat aloittaa keskustelua, jakaa tiedostoja ja organisoida tehtäviä vaivattomasti, eikä pitkiä viestiketjuja tarvitse ruveta etsimään esimerkiksi sähköpostin syövereistä (Basecamp 2016). Basecampiin tehdyssä projektissa säilytettiin asiakastapaamisista kertyneet muistiinpanot, sivuston rautalankamallit, sivuston tunnukset sekä suunnitelma verkkosivuston toteutuksen eri vaiheista.

3.3 Bitbucket ja Git

Bitbucket on Git-versionhallintaohjelmistoa käyttävä pilvipohjainen palvelu. Bitbucket valittiin tähän toteutukseen sen takia, että sinne saa ilmaiseksi rajoittamattoman määrän yksityisiä koodivarastoja eli repositoreja tiimin koon ollessa alle viisi henkilöä (Bitbucket 2016). Bitbuckettiin tallennettiin verkkosivujen eri versiot kehitysvaiheissa.

3.4 Kehitysympäristö Vagrant

Vagrant on työn toteutukseen valittu avoimeen lähdekoodiin perustuva kehitysympäristö (Vagrant 2016). Vagranttiin asennettiin Varying Vagrant Vagrants, joka on Wordpress-kehitykseen keskittyvä Vagrant-konfiguraatio. Se helpottaa ja nopeuttaa Wordpress-ympäristön pystyttämistä.

3.5 MySQL Workbench

MySQL Workbench on tietokantojen hallintaan käytetty visuaalinen työkalu. Workbenchin avulla tutkittiin Drupalin ja Wordpressin tietokantoja blogikirjoitusten siirron toteuttamiseksi.

3.6 JQuery

JQuery on suosituin käytössä oleva javascript-kirjasto. Se on suunniteltu erityisesti html-dokumenttien manipuloimiseen ja animaatioiden tekemiseen. JQuerystä löytyy paljon ominaisuuksia ja sitä tukevat kaikki suosituimmat internet selaimet (JQuery 2016).

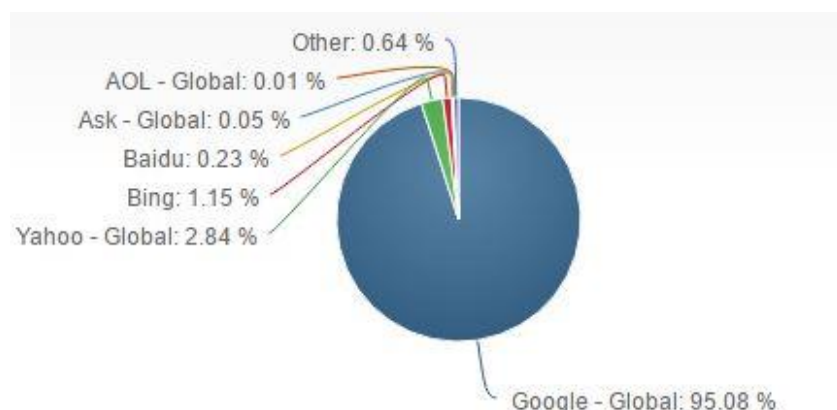
4 Hakukoneoptimointi ja konversio

4.1 Yleistä

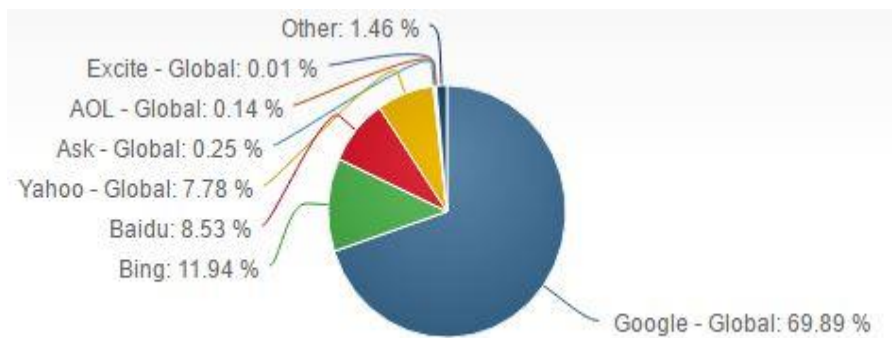
Hakukoneoptimoinnilla tarkoitetaan toimenpiteitä, joilla vaikutetaan verkkosivujen hakukonenäkyvyyteen ja sijoitukseen hakukoneiden ilmaisissa eli orgaanisissa hakutuloksissa. Kilpailukyvyn ylläpitämiseksi tulee yritysten seurata hakukonesijoituksiaan. Yleisesti voidaan sanoa, että mitä korkeammalla ja useammin sivusto esiintyy hakutuloksissa, sitä enemmän sivusto saa vierailijoita hakukoneiden käyttäjistä. (Search engine optimization 2016) Hyvin suunnitellulla verkkosivustolla nämä kävijät voidaan konvertoida esimerkiksi asiakkaiksi tai yhteydenotoiksi. Tämän takia korkealla hakutuloksissa näkyminen on elintärkeää monelle yritykselle. Pelkkä korkealla hakutuloksissa näkyminen ei kuitenkaan yleensä riitä, jos käyttäjiä ei ohjata sivuston omistajan haluamaan suuntaan.

Konversioprosentilla tarkoitetaan sitä osuutta verkkosivulla vierailijoista, jotka pelkän sivuston katselun ja selailun sijaan tekevät jotain sivuston omistajan tavoitteeksi asettamista asioista. Konversioksi voidaan laskea sivuston tyypistä riippuen esimerkiksi sähköpostilistalle liittyminen, viestin lähettäminen yhteydenottolomakkeelta, videon katsominen tai suoraan tuotteen ostaminen. (Conversion marketing 2016.)

Kuvioista 2 ja 3 nähdään, että Google on ylivoimaisesti käytetyin hakukone sekä mobiililaitteilla että pöytäkoneilla (Netmarketshare 2016).



Kuvio 2. Hakukoneiden käyttötilastot mobiililaitteilla kesäkuu 2016. (alkup. kuvio ks. Netmarketshare. 2016)



Kuvio 3. Hakukoneiden käyttötilastot pöytäkoneilla kesäkuussa 2016. (alkup. kuvio ks. Netmarketshare 2016)

Hyvän hakukonesijoituksen avulla sivustolle tulee enemmän ilmaista eli orgaanista liikennettä. Googlen hakutulosten kolme ensimmäistä linkkiä haalivat yli 90 % kaikesta liikenteestä. 33 % käyttäjistä klikkaa ensimmäistä, 18 % toista ja 11 % kolmatta orgaanista hakutulosta. (Lee 2013.) Jos sivuston rakenne ja sisältö on toteutettu oikein, se ohjaa vierailijoita sivuston ylläpitäjän haluamaan suuntaan. Motiivina saattaa olla esimerkiksi vierailijoiden liittyminen sähköpostilistalle tai yhteydenotot yrityksen asiakaspalveluun. Konversioksi kutsutaan sivustolle tulevan liikenteen muuttumista yhteydenotoiksi tai suoraan asiakkiksi. Ei kannata houkuttaa vierailijoita, jotka etsivät jotain muuta kuin mitä tarjoat. Jos ihmiset eivät löydä verkkosivustolta etsimäänsä, he lähtevät sivulta ja jatkavat etsimistä. Tällöin vierailijat eivät myöskään muutu asiakkiksi tai kassavirraksi. Jokaisella laskeutumissivulla tulisi olla jokin teema tai aihepiiri, mihin tämä kyseinen sivu liittyy, jolloin aihetta etsittäessä henkilö löytää ideaalisesti vain itseään kiinnostavaa sisältöä. (Hunt 2011.) Hyvän tekstin sisällöstä tulisi käydä ilmi ainakin, miten vierailija hyötyy juuri tästä kyseisestä tuotteesta ja miten se erottuu kilpailijoista ja miten aiheesta kiinnostunut näkee hyödyt jatkossa.

Sivuston hakukonesijoitukseen vaikuttavat sekä sivuston ulkoiset, että sisäiset tekijät. Verkkosivujen suunnittelu- ja toteutusvaiheissa sivuston tekijä voi pääasiassa vaikuttaa vain sivuston sisäisiin tekijöihin. Google kannustaa kirjoittamaan sisältöä ja rakentamaan verkkosivuja, jotka on suunniteltu käyttäjiä eikä hakukoneita varten. (Webmaster Guidelines 2016.)

Google ja muut hakukoneet päivittävät algoritmejaan satoja kertoja vuodessa. Vaikka useimmat näistä muutoksista ovatkin erittäin pieniä, välillä tulee suurempia algo-

ritmimuutoksia, jotka vaikuttavat hakutuloksiin merkittävästi. Google ei paljasta algoritmien tarkkaa sisältöä, mutta kertoo yleensä suuntaukset.

4.2 Sivuston sisäiset tekijät

4.2.1 Yleistä

Hakukonesijoitukseen vaikuttavat sivuston sisäiset tekijät täytyy ottaa huomioon jo verkkosivua suunniteltaessa, kun haluaa sivuston löytyvän korkealta Googlen hakutuloksissa. Verkkosivustoa suunniteltaessa kannattaa aloittaa miettimällä, kenelle sivusto on suunnattu ja jaotella sisältö aihepiirien mukaan. Tämän jälkeen kannattaa suunnitella laskeutumissivut niin, että niitä tulee yksi laskeutumissivu jokaiselle aiheelle, jolloin kyseistä aihetta hakukoneista etsittäessä vierailija löytäisi heti sille aiheelle omistetun laskeutumissivun, jossa on vain sisältöä, josta hän on kiinnostunut. (Hunt 2011.)

Googlen botit käyvät läpi html-tiedostojen sisältöä algoritmien avulla eivätkä ymmärrä sivuston visuaalisuudesta, kuten väreistä mitään. Jos verkkosivuilla olevaa kuvaa ei voida jostain syystä näyttää, näkyy kuvan tilalla tekijän määrittelemä vaihtoehtoinen teksti eli alt text. Myös hakukoneet tarkistavat vaihtoehtoisen tekstin, minkä takia kuville kannattaakin laittaa vaihtoehtoinen teksti niiden sisällön kuvaamiseksi ja auttaa hakukoneita ymmärtämään sen. (Search Engine Optimization Starter Guide 2010.)

4.2.2 Yksilölliset otsikot

Ttitle-tagin on yksittäisen laskeutumissivun otsikko, joka näkyy selaimen välilehdellä. Title näkyy myös hakukoneen hakutuloksissa klikattavan linkin otsikkona. Titleen kannattaa sisällyttää aina laskeutumissivun tärkein avainsana. Tärkein avainsana tulisi myös sisällyttää pääotsikkoon ja vähintään yhteen alaotsikkoon eli mino heading -tagiin.

4.2.3 Laadukas sisältö

Yksi tärkeimmistä sijoitukseen vaikuttavista tekijöistä on laadukas tekstisisältö. Kannattaa valita yksi avainsana laskeutumissivun aiheen mukaan, jonka perusteella ky-

seinen sivu optimoidaan. Avainsana tulisi sisällyttää vähintään yhden kuvan nimeen ja sen vaihtoehtoiseen tekstiin. Sisällön tulisi olla vähintään 300-sanaa pitkä, jotta siitä saa hyötyä hakukoneiden silmissä. Hakukoneet kiinnittävät huomiota myös hakusanojen tiheyteen ja avainsanojen asetteluun. (GoogleInside Search N.d)

4.2.4 Metatiedot

Sivustoa optimoitaessa on hyvä pitää mielessä myös metatiedot, joita verkkosivuilla ilmaistaan metaelementeillä. Meta-elementti Meta description on lyhyt, hakutuloksissa klikattavan linkin alla näkyvä tekstin pätkä. Se on noin 160 merkkiä pitkä kuvaus kyseisen laskeutumissivun sisällöstä. Hakukoneet näyttävät metakuvauksen silloin, kun haettu termi sisältyy kyseisen sivun metakuvaukseen. Metakuvaus on epäsuora vaikutus hakukonesijoitukseen. Se vaikuttaa linkin klikkausprosenttiin ja sitä kautta myös sivun sijoittumiseen pidemmällä aikavälillä, jos onnistut kirjoittamaan sellaisen kuvauksen, joka saa ihmiset klikkaamaan sivun linkkiä. Tässä mielessä laskeutumissivujen metakuvaus onkin tärkeä osa sivuston sisäistä hakukoneoptimointia.

4.2.5 Selkokiehiset URL-osoitteet

Selkokiehiset URL-osoitteet voivat auttaa hakukoneita lukemaan sivuja paremmin. Sivun URL-osoite näytetään Googlen hakutuloksessa Metakuvaksen alapuolella ja siinä olevat sanat näkyvät lihavoituna, jos ne sisältyvät käyttäjän hakulauseeseen. (Search Engine Optimization Starter Guide 2011, 8.)

4.2.6 Mobiiliystävällisyys

Google alkoi rankaista puhelimille epäystävällisiä sivuja vuonna 2015 (Google Webmaster Central Blog 2015). Mobiiliystävällisyys tarkoittaa verkkosivujen käytettävyyttä erilaisilla mobiililaitteilla, kuten puhelimilla tai tableteilla.

Aiemmin suosiossa olleet erilliset mobiilisivut ovat vaihtumassa responsiivisiin verkkosivuihin. Responsiivisuus tarkoittaa verkkosivuston sisällön mukautumista sivustolla vierailevan laitteen näytönkoon mukaan.

4.2.7 Verkkosivuston latautumisenopeus

Verkkosivun latautumisenopeus on yksi hakukonesijoitukseen vaikuttavista tekijöistä. Hyvän palvelimen tai webhotellin valintaan kannattaakin kiinnittää huomiota. Verkkosivun nopeuteen pystytään vaikuttamaan melko paljon myös palvelimen konfiguraatiota muuttamalla, joko suoraan palvelimen asetuksista tai Apachen tapauksessa tekemällä muutoksia Htaccess-tiedostoon. Verkkosivujen nopeuden mittaamiseen käytettiin kolmea eri työkalua: Google PageSpeed Insight, GTMetrix ja Pingdom speed test.

Yksi merkittävä nopeuteen vaikuttava tekijä on kuvien ja verkkosivun koko. Mitä suurempi verkkosivun tiedostokoko on, sitä kauemmin sen latautumisessa selaimeen kestää. Kuvien resoluutio tulisi skaalata siihen kokoon, jossa niitä on tarkoitus näyttää sivustolla. Jos kuva on liian suuri kyseiseen osioon, selain lataa kuvan ensin sen alkuperäisessä koossa ja skaalaa sen sitten kyseiseen osioon sopivaksi. Näin selain lataa ikään kuin turhia pikseleitä ja sitten skaalaa kuvan pienemmäksi, mikä on hitaampaa, kuin suoraan oikean kokoisien kuvien lataaminen. Kuvat kannattaa myös pakata, jolloin niiden tiedostokoko pienenee ilman, että kuvanlaatu kärsii.

4.2.8 Sisäiset linkit

Sisäinen linkki on hyperlinkki, joka osoittaa saman domainin alla olevalle toiselle laskeutumissivulle. Tällaiset linkit ovat tärkeitä useasta eri syystä. Ne auttavat käyttäjiä navigoimaan sivustolla, kertovat hakukoneille sivuston tietohierarkiasta ja auttavat levittämään linkkimehua eli sijoituspisteitä ympäri koko sivuston. (Moz 2016.) Näitä linkkejä käytetään yleisesti sivuston navigaatiossa. Linkit muihin sivustoihin auttavat Googlea määrittelemään, mihin aiheeseen kyseinen laskeutumissivu liittyy.

4.2.9 Sivustokartta

Sivustokartta on tiedosto, minne voidaan listata kaikki sivustolla olevat laskeutumissivut. Sivustokartan avulla Google ja muut hakukoneet saavat selville sivuston rakenteen. Kun hakukoneiden botit ovat lukeneet kyseisen tiedoston, voivat ne käydä sivuston läpi järkevämmin. Sivustokartta voi myös tarjota hakukoneille arvokasta metatietoa mm. siitä, milloin laskeutumissivu on viimeksi päivitetty, kuinka usein sitä

muokataan ja miten tärkeä se on muihin sivuston URL-osoitteisiin verrattuna. (Google Search Console Help 2016.)

4.3 Sivuston ulkoiset tekijät

Sivuston ulkoisten tekijöiden optimointi tarkoittaa pelkistettynä sivuston suosion ja kävijämäärän lisäämistä. Sivustolle osoittavien linkkien määrä ja laatu ovatkin yksiä tärkeimmistä hakukonesijoitukseen vaikuttavista kriteereistä. ”Linkkimehulla tarkoitetaan muilta sivuilta omalle sivustollesi osoittavien ja sivuston sisäisten linkkien yhteenlaskettua voimaa.” (Saarinen, J. 2010) Jokainen sivustolle osoittava linkki auttaa sivustoa nousemaan paremmille paikoille hakutuloksissa.

Sijoitukseen vaikuttaa myös sivuston maine sosiaalisessa mediassa. Hakukoneet käyttävät myönteisiä sosiaalisen median signaaleja sivustojen sijoitusten määrittämiseen hakutuloksissa. Signaaleina toimivat sisällön twiitit, tykkäykset, jaot ja niiden määrä, jotka auttavat hakukoneita määrittämään onko sisällöstä hyötyä hakukoneenkäyttäjille. (Suomen digimarkkinointi Oy 2016.)

5 Verkkosivuston toteutus

Työn toteutus aloitettiin tekemällä vaatimusmäärittely ja suunnittelemalla sen pohjalta rautalankamallit. Tehtävien hallintaan käytettiin Asana-nimistä tehtävienhallintaohjelmia ja projektin hallintaan Basecampia.

5.1 Suunnittelu ja käytettävyys

Hakukoneoptimointi, sivuston toimivuus mobiilissa, ylläpidettävyys ja mahdollisimman suuri konversioprosentti olivat tärkeimmät vaatimukset uudelle verkkosivustolle. Vaatimusmäärittelyn mukaan tehtiin rautalankamallit ja vuokaaviot. Toteutuksen tekeminen aloitettiin kehitysympäristöön sen jälkeen, kun toimeksiantaja oli hyväksynyt sivuston rakenteesta tehdyt rautalankamallit.

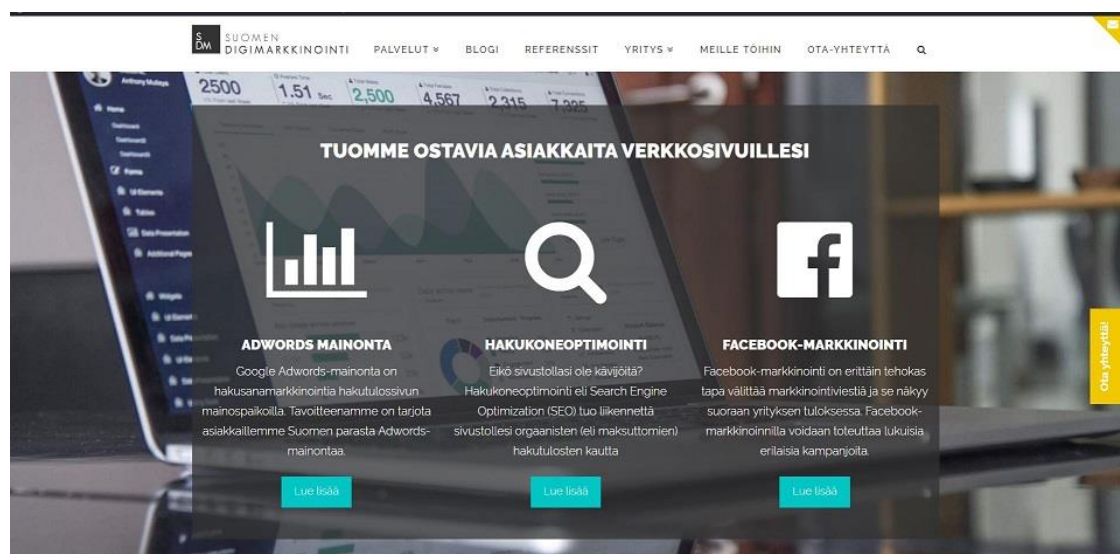
5.2 Kehitysympäristö

Työkoneessa käytettiin käyttöjärjestelmänä Untuntu 16.04 -versiota, johon asennettiin Virtual box -virtuaalikone. Virtuaalikoneeseen asennettiin kehitysympäristö Vagrant ja Varyng Vagrant Vagrants, jota myös Wordpress-ytimen kehittäjät käyttävät.

Projektin versionhallintaan käytettiin git-versionhallintaa ja eri versiot tallennettiin Bitbucket-pilveen. Teeman ja kooditiedostojen muokkaamiseen käytettiin Sublime text -tekstinkäsittelyohjelmaa, joka tunnistaa eri koodikieliä, osaa värittää syntaksin ja sientää koodia.

5.3 Layout

Rautalankamallit sivujen asettelusta piirrettiin Ninjamock-suunnittelutyökalulla. Wordpressin-teemaksi valittiin X-teema ja sille rakennettiin oma lapsiteema vastamaan vaatimusmäärittelyä. X-teemalle tehtiin lapsiteema, jonka avulla X-teemaa pystyttiin muokkaamaan toimeksiantajan vaatimusmäärittelyn mukaisesti ilman, että teemaa päivittäessä tarvitsisi pelätä muutosten häviämistä. Ulkoasu suunniteltiin yrityksen uuden brändi-ilmeen ja vaatimusmäärittelyn ominaisuuslistan pohjalta, jolloin siitä tuli toimeksiantajan toiveiden mukainen. Toimeksiantajalla oli tarkat vaatimusmäärittelyt, joten valittua teemaa piti muokata paljon. Lopullisesta ulkoasusta muodostui kuvion 4 mukainen.



Kuvio 4. Kuvankaappaus <http://www.digimarkkinointi.fi> etusivulta.

5.4 Sisällön tuonti

5.4.1 Tausta

Vaatimusmäärittelyyn kuului vanhalla sivustolla olevien artikkeleiden siirtäminen uudelle sivustolle. Molemmat julkaisujärjestelmät tallentavat artikkelit tietokantaan, joten niiden siirtämisen julkaisujärjestelmästä toiseen tiedettiin mahdolliseksi. Sekä Drupal että Wordpress käyttävät MySQL-tietokantaa ja artikkeleiden siirtämiseksi tuli ymmärtää molempien julkaisujärjestelmien tietokantakaaviot. Kaikkien artikkeleissa olevien kuvien, siinä käytettyjen avainsanojen (tags) ja tiedon siitä, kuka artikkelin on kirjoittanut, piti siirtyä uuden tietokannan oikeisiin kenttiin. Drupal tallentaa ja käsittelee blogikirjoituksia Nodeina (About nodes 2016). Nodet vastaavat Wordpressin sisältötyyppejä.

Tietokantakaavioiden tutkiminen on työlästä, ja siksi sisällön siirtämiseen varattiin yksittäisistä tehtävistä eniten aikaa. Kaikkien toisiaan vastaavien kenttien löytäminen molemmista tietokannoista ja kyselyiden kirjoittamien osoittautui ajoittain haastavaksi ja aikaa vieväksi. Aluksi tutustuttiin eri vaihtoehtoihin blogin siirron tekemiseen. Parhaaksi vaihtoehdoksi osoittautui oman lisäosan tekeminen.

Mallia otettiin myös pari vuotta vanhasta lisäosasta, joka oli suunniteltu Drupal-sivuston sisällön siirtämiseen Wordpress-sivustolle. Kyseinen lisäosa oli kuitenkin tehty Wordpress 3-versiolle ja ei toiminut sellaisenaan nykyisessä 4.5.3 versiossa. Lisäosan tietokantakyselyjä muokattiin vastaamaan nykyistä tietokantarakennetta, joiden avulla tiedot lopulta saatiin siirrettyä.

5.4.2 Toteutus

Vanhalta sivustolta ladattiin Drupalin käyttämä tietokanta. Tämän jälkeen molempien julkaisujärjestelmien tietokannat laitettiin samalle MySQL-tietokantapalvelimelle, jolloin tietokantoihin pääsi helposti käsiksi tietokantakyselyillä. Palvelimella oleva Wordpressin-tietokanta oli käytössä tuoreessa Wordpress-asennuksessa, jossa tietokantakyselyjä tekevää lisäosaa muokattiin.

Aluksi tietoja siirrettäessä PHP-muistiraja tuli vastaan ja ohjelma kaatui tietokannassa olevan suuren blogikirjoitus määrän takia. Muistirajaa nostettiin ja lopullista siir-

toa tehtäessä export-tiedosto pilkottiin pienempiin osiin ja sisällön tuonti suoritettiin kahdessa eri osassa, jolloin ongelmia ei tullut.

Drupal-sivuston palvelimella olevat kuvat ladattiin ftp-yhteyden avulla ja siirrettiin Wordpress-asennuksen uploads -kansioon. Tiedostopolkujen muuttuessa kuvien lähteiden osoitteet eivät enää täsmänneet ja kuvat ”hajosivat”. Tämä ongelma korjattiin vaihtamalla tietokannassa olevien artikkeleiden kuvien lähdeosoitteet yksinkertaisella tietokantakyselyllä (ks. kuvio 5). Hakukoneet eivät välitä siitä, kuinka sisältö on tehty, mutta niiden sijainti sivustolla on erittäin tärkeä. Sivuston osoiterakente pyrittiin rakentamaan mahdollisimman samanlaiseksi kuin vanhalla sivustolla. Muuttuneet linkit 301- uudelleen ohjattiin Apachen mod_rewrite -moduulin avulla. Sisällöntuonti-lisäosaan ei voitu täysin luottaa, joten siirretty sisältö piti vielä tarkastella manuaalisesti läpi, mikä oli aikaa vievää ja tuskallista. Tämä oli kuitenkin ainut tapa varmistua, että kaikki sisältö todellakin oli siirtynyt Wordpress-ympäristöön. Lisäosa saatiin toimimaan ja vanhasta tietokannasta saatiin onnistuneesti kaikki blogikirjoitukset ja niihin liittyvä tieto uuteen ympäristöön.

5.5 Kävijäseuranta

Sivustolle tulevan liikenteen seuraamiseen käytettiin Google Analytics -kävijäseurantaa, Google Tag Manager -työkalua ja Crazy Eggiä -heatmap mittauksia.

5.6 Sisältötyypit

Wordpress pystyy säilyttämään ja näyttämään monenlaista sisältöä. Kaikki sisältötyypit eli post types ovat tallennettuna samaan wp_posts-nimiseen tietokantatauluun ja niitä erottaa toisistaan post_type-niminen kenttä. (Wordpress Codex 2016.) Mukautettu sisältötyyppi on sivuston tarpeiden mukaiseksi tehty pala sisältöä.

Wordpressissä on oletuksena viisi erilaisia sisältötyyppiä ja niiden luokitteluja. Nämä sisältötyypit ovat Post, Page, Attachment, Revision ja Nav Menus (Williams, Damstra & Stern 2015, 127-128). Pientä blogia tai yksinkertaisia kotisivuja varten nämä tyypit usein riittävät sivuston toteuttamiseen. Toimeksiannon tekemiseksi oli kuitenkin välttämätöntä tehdä omia sisältötyyppejä, jotta lopputuloksesta tulisi täsmälleen haluttuunlainen. Mukautettuja sisältötyyppejä käyttämällä voidaan määrittää minkälaisia

sisältöä tahansa, eikä olla enää pakotettuja käyttämään aiemmin listattuja sisältötyyppejä.

Uusia sisältötyyppejä varten tehtiin Wordpress-lisäosa. Tekemällä oman lisäosan sen sijaan, että koodi sisällytettäisiin teemaan, sivuston toiminnallisuus ei muutu, vaikka käytössä olevaa teema haluttaisiinkin joskus tulevaisuudessa vaihtaa.

Uudet sisältötyypit tehtiin palveluille, hakemuksille, työntekijöille ja referensseille, jolloin Wordpressin hallintapaneelissa on niille jokaiselle oma välilehti (ks. liite 5). Näin niitä on helppo hallita ja uusia lisättäessä ne menevät sivustolla automaattisesti oikeaan paikkaan.

5.7 Teeman kehitys

Wordpress-sivuston ulkoasu ja sivuston asettelu määräytyvät käytössä olevan teeman mukaan. Valinta millaista teemaa käytetään rippuu tarvittavan räätälöinnin määrästä sekä yksittäisen projektin tarpeista. (Williams, ym. 2015, 136). Tässä työssä sivustoa lähdettiin rakentamaan X-teeman pohjalta. Teeman muokkaamiseksi tehtiin lapsiteema, jotta tehdyt muutokset eivät häviäisi mahdollisten teemaan tulevien päivitysten mukana. Lapsiteema tehtiin lisäämällä Isäntäteemaa vastaava kansiorakenne Wordpressin themes –kansioon. Tämän jälkeen teeman juuri-kansioon lisättiin Styles.css –tiedosto, jota Wordpress käyttää teemaan viittaamiseen ja se on pakko olla hakemistossa teeman toimimiseksi (Williams ym. 2015, 137). Kuvio 5 sisältää teeman luontiin tarvittavat kentät.

```

1  /*
2
3  Theme Name: Digimarkkinointi
4  Theme URI: http://digimarkkinointi.fi
5  Author: Aatu Pitkänen
6  Author URI: http://digimarkkinointi.fi
7  Description: Suomen digimarkkinointi Oy:n teema
8  Version: 1.0.0
9  Template: x
10
11 */

```

Kuvio 5. Teeman luontiin käytetty styles.css -tiedoston sisältö.

Tämän jälkeen kyseistä teemaa lähdettiin rakentamaan ja se aktivoitiin Wordpressin-hallintapaneelin kautta, jolloin teema tuli sivustolla käyttöön.

Wordpress etsii teeman tiedostoja ensin lapsiteeman hakemistosta ja jos niitä ei löydy, käyttää se oletuksena Isäntäteeman-tiedostoja. Teeman muokkaaminen pitää sisällään pääasiassa PHP- ja CSS -tiedostojen muokkauksia. Varsinkin eri sisältötyyppien arkistosivujen template-tiedostot vaativat paljon muutoksia, jotta niistä saatiin rautalankamallien mukaiset.

Arkistosivun tarkoitus Wordpressissä on listata kaikki sisältötyypin jäsenet. Esimerkiksi palvelut sisältötyypin arkistosivulla listataan kaikki yksittäiset palvelut. Arkistosivu aktivoitiin lisäämällä `has_archive`-parametrin arvoksi `true` sisältötyypin rekisteröinnin yhteydessä ks. Rivi 114 (Liite 5.) Wordpress osaa automaattisesti yhdistää tiedoston oikeaan sisältötyyppiin, kun template-tiedosto nimetään muotoon *archive-palvelut.php*. Yksittäisen sisällön templatien voi niin ikään tallentaa nimellä *single-palvelut.php*, jolloin Wordpress osaa yhdistää sen automaattisesti yksittäisen palvelun templateksi. Jokaiselle uudelle sisältötyypille tehtiin oma lajittelu, mikä mahdollistaa sisältöjen ryhmittelyn kategorioihin, mikäli sille on tulevaisuudessa tarvetta.

Lisäksi sisältötyypeille tehtiin omat kentät, esimerkiksi uusia työntekijöitä -sivustolle lisättäessä voi tittelin suoraan syöttää titteli-nimiseen tekstikenttään. Tämän ansiosta ylläpitäjän ei tarvitse huolehtia uuden sisällön asettelusta, vaan jokainen uusi työntekijä noudattaa samaa formaattia. X-teemasta tuli joitain tekstipätkiä sivustolle englannin-kielillä. Nämä tekstit muutettiin suomenkielisiksi lisäämällä kuvion 6 mukainen koodi lapsiteeman `Functions.php`-tiedostoon.

```

341 add_filter( 'gettext', 'replace_search_string', 20, 3 );
342 function replace_search_string( $search_text, $text, $domain ) {
343     if ( $text == 'Type and Press &ldquo;enter&rdquo; to Search' ) return __( 'Kirjoita hakutermi ja paina "Enter"', $domain );
344     else return $search_text;
345 }
```

Kuvio 6. X-teemasta tulevan hakutekstin korvaaminen suomenkielisellä tekstillä

5.8 Sivuston nopeuden optimointi

Verkkosivujen nopea latautuminen on erittäin tärkeää. Jaetusta verkkoympäristöstä johtuen ei pääsyä ollut palvelimen konfiguraatioon suoraan, joten konfiguraatiomuutokset tehtiin Apachen `Htaccess` -tiedostoon.

Gzip-pakkaaminen on pakkauksen muoto, joka pakkaa CSS, Javascript, ja html -tiedostot palvelimen puolella ennen kuin ne lähetetään käyttäjän selaimeen. Näin verkkosivuston tiedostokoko pienenee ja myös lataus nopeutuu. Gzip-pakkaus aktivoitiin lisäämällä uusi moduuli Apachen .htaccess -tiedostoon. (Ks. liite 1.)

Selaimen välimuistia hyödynnettiin lisäämällä vanhentumisotsakkeet (expire headers) Htaccess-tiedostoon (Ks. liite 2.). Expire headerit kertovat selaimelle, tulee-ko sen pyytää tietyt tiedostot palvelimelta vai ottaa ne suoraan selaimen välimuistista. Näin vähennettiin tarvittavien http-pyyntöjen määrää, mikä osaltaan myös nopeuttaa sivuston latautumista. Sivustolla olevien kuvien pakkaamiseen käytettiin Lossless compression -nimistä metodia, jonka avulla kuvien tiedostokokoa voi pienentää ilman, että kuvien laatu kärsii.

Proxy-palvelimet eivät lisää välimuistiinsa resursseja, joissa on "?"-merkki URL-osoitteessa. Tästä saattaa olla haittaa nopeuden kannalta, joten Query-stringit poistettiin kuvion 7 mukaisella -koodilla.

```

286 // =====
287 // remove script version
288 // =====
289 |
290 function _remove_script_version( $src ){
291 $parts = explode( '?ver', $src );
292 return $parts[0];
293 }
294 add_filter( 'script_loader_src', '_remove_script_version', 15, 1 );
295 add_filter( 'style_loader_src', '_remove_script_version', 15, 1 );
296

```

Kuvio 7. Kyselyparametrien poistaminen CSS-tiedostoista

5.9 Muu räätälöinti

Mailchimp on sähköpostilistan keräys ohjelma. Olemassa olevalle Mailchimp-sähköpostilistalle liittyminen haluttiin uusille verkkosivuille. Sivustolle tehtiin kolme erilaista lomaketta Mailchimp integraatiota varten. Kaksi eri tuotteiden ostajan oppaita varten ja yksi lomake, josta pystyi liittymään pääsähköpostilistalle. Mailchimpin integrointiin Wordpressin kanssa käytettiin lisäosaa. Sivustolle lisättiin toiminnallisuutta myös Javascript-kirjasto JQueryyn avulla. Yksittäisten blogikirjoitusten asettelusta johtuen kaikkia vanhoissa blogikirjoituksissa olevia kuvia piti tummentaa, jotta

kuvan päällä oleva otsikko ja teksti näkyisivät paremmin. Kuvat tummennettiin JQuery-funktion avulla (ks. kuvio 8). Myös Headerin väri laitettiin vaihtumaan scrollauskeksen jälkeen läpinäkyvästä valkoiseksi JQuery avulla.

Sivupalkissa oleva yhteydenottolomakkeen skrollauksen seuraaminen toteutettiin niin ikään JQueryn avulla. (ks. kuvio 8). Funktio tarkistaa ensin, että näytön leveys on alle 980-pikseliä. Sivupalkki ei näy mobiililaitteilla, joten yhteydenottolomaketta ei myöskään haluta silloin ruudulle. Funktio lisää lomakkeseen CSS-luokan, joka muuttaa sen position-ominaisuuden fixed tilaan, kun käyttäjä on Skrollannut sivua yli 1200 pikseliä ja lomakkeen etäisyys sivun footeriin on yli 750-pikseliä. Näin yhteydenottolomake saadaan seuraamaan skrollausta ja sivuston alareunaan tultaessa se ei mene footerin päälle.

```

6 /*Skrollausta seuraava yhteydenotto Lomake*/
7 jQuery(function($) {
8   if ($(window).width() > 980) {
9     $(window).scroll(function(){
10      if($(window).scrollTop() >1200 && ($(window).scrollTop() < $('mailchimp-footer').offset().top - 750)) {
11        $('.sidebar-form').addClass("fixed-form");
12      }
13    }else {
14      $('.sidebar-form').removeClass("fixed-form");
15    }
16  });
17  $(window).scroll(function(){
18    if($(window).scrollTop() > 1200 && ($(window).scrollTop() < $('mailchimp-footer').offset().top - 450)) {
19      $('.sidebar-chimp').addClass("fixed-form");
20    }
21    } else {
22      $('.sidebar-chimp').removeClass("fixed-form");
23    }
24  });
25  });
26  });
27 /*Blogi headerin tummennus*/
28 jQuery(function($) {
29   $(".blogi-header-single").wrapInner("<div class='section-overlay'></div>");
30 });
31 /*Leijuva yhteydenotto nappi*/
32 jQuery(function($) {
33   if( ! $('body').hasClass("page-id-210") ) {
34     $('<a href="http://digimarkkinointi.fi/ota-yhteytta/" class="floating-button x-btn x-btn-flat">Ota yhteyttä!</a>').appendTo('.site');
35   }
36 });

```

Kuvio 8. Sivustolla käytetyt JQuery-funktiot

Wordpress järjestää oletuksena sisältötyypit arkistosivuilla uusimmasta pienimpään, eli uusin sisältö tulee ylimmäksi. Joillekin sisältötyypeille haluttiin kuitenkin vanhojen sisältöjen pysyvän aina ylimpänä ja uusien menevän listan pohjalle. Palveluiden ja asiantuntijoiden määrä sekä niiden järjestys muutettiin lisäämällä Query_posts-funktio näiden sisältötyyppien template-tiedostoihin. (ks. kuvio 9.)

```

10 <?php global $query_string; query_posts($query_string . "&order=ASC&posts_per_page=30" ); ?>
11

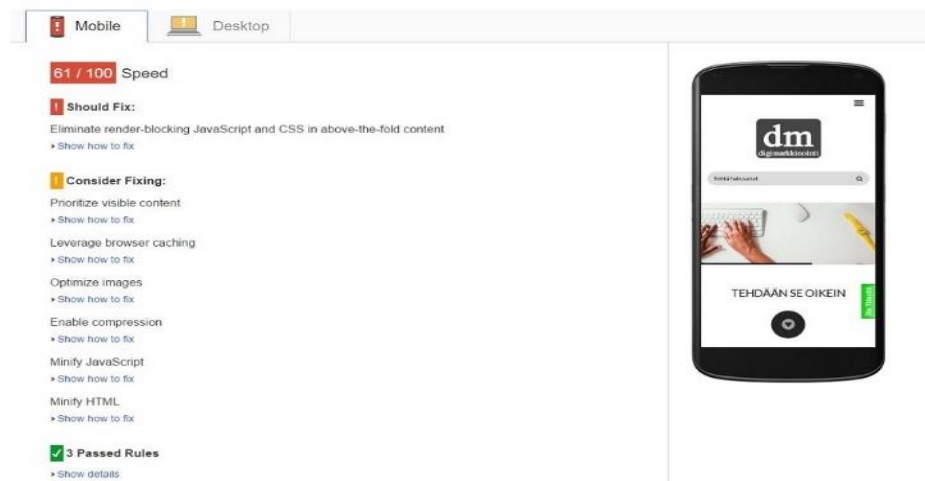
```

kuvio 9. Sisältötyypin määrän ja järjestyksen muuttaminen

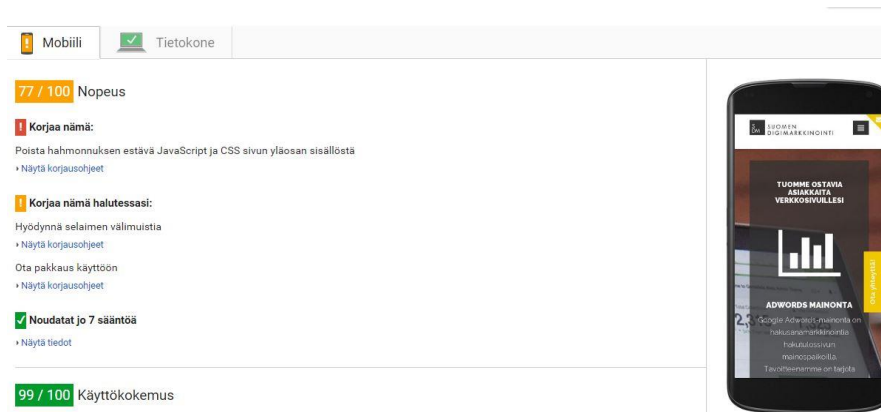
Lisäksi Palvelut-sisältötyypille tehtiin paljon alisivuja eli alapalveluita. Nämä alapalvelut haluttiin näyttää sivupalkissa aina, kun sivustolla kävijä tarkastelee jotain yksittäistä palvelua. Alapalveluiden listaus sivupalkkiin toteutettiin wp_list_pages-funktion avulla ja lopuksi lisättiin ne shortcodeen, joka lisätään sivupalkkiin. (Ks. liite 3).

6 Tulokset

Lopputuloksena saatiin uudet hakukoneoptimoidut verkkosivut. Aikataulusta myöhästettiin pari viikkoa mm. muuttuneiden vaatimusten sekä muiden aikaa vieneiden projektien takia. Sivuston valmistuttua siitä löytyi kuitenkin kaikki vaaditut ominaisuudet, palvelusivut ja blogi löytyvät edelleen korkealta Googlen ilmaisissa hakutuloksissa, sekä tehtyjen nopeustestien perusteella sivun myös latautuvat keskimäärin nopeammin kuin kilpailijoilla. Google PageSpeed Insights on Googlen kehittäjätyökalu, jonka avulla voi testata sivustonsa nopeutta Googlen algoritmien avulla. Kuviossa 10 ja 11 näkyvät sekä vanhalle että uudelle sivustolle kyseisellä työkalulla tehtyjen testien tulokset.

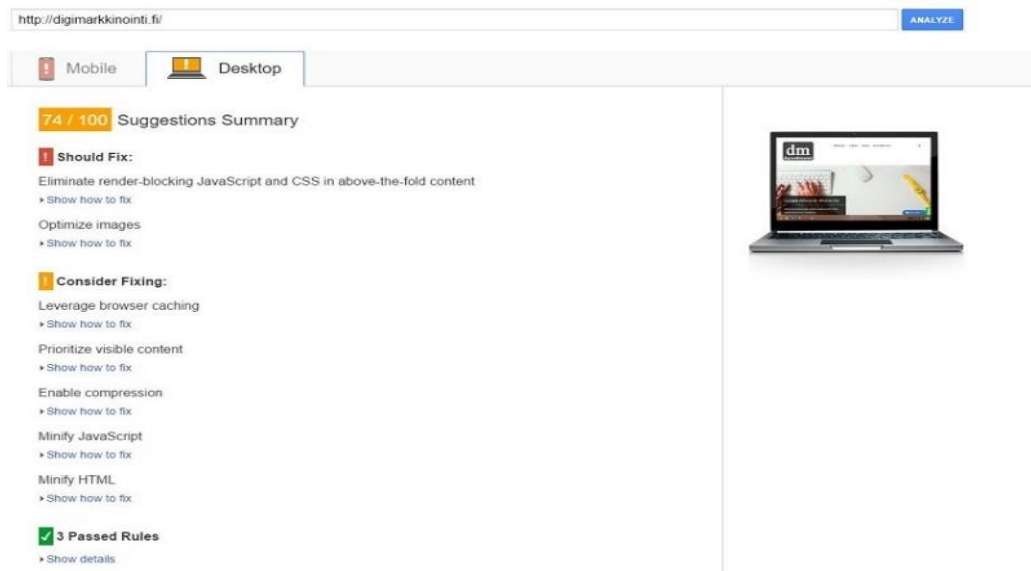


Kuvio 10. Vanhan verkkosivun nopeusanalyysi mobiiliversiosta

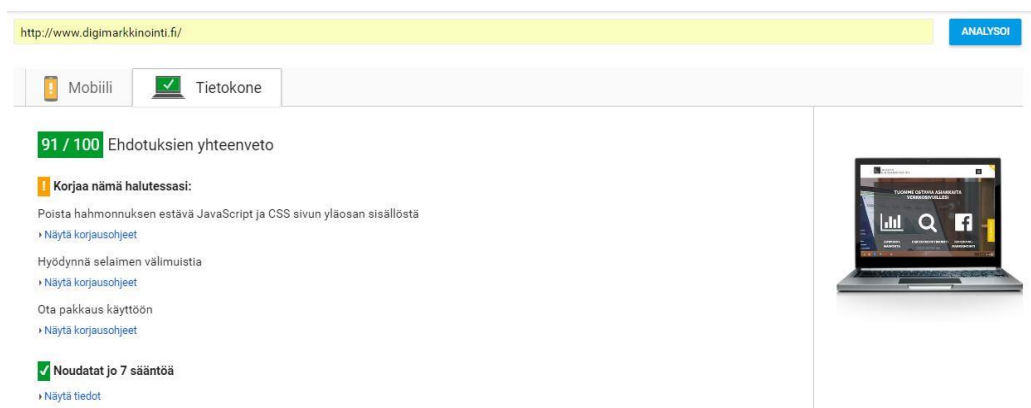


Kuvio 11. Uuden verkkosivun nopeusanalyysi mobiiliversiosta

Kuviossa 12 ja 13 näkyvät samojen testien tulokset pöytäkoneversioille.



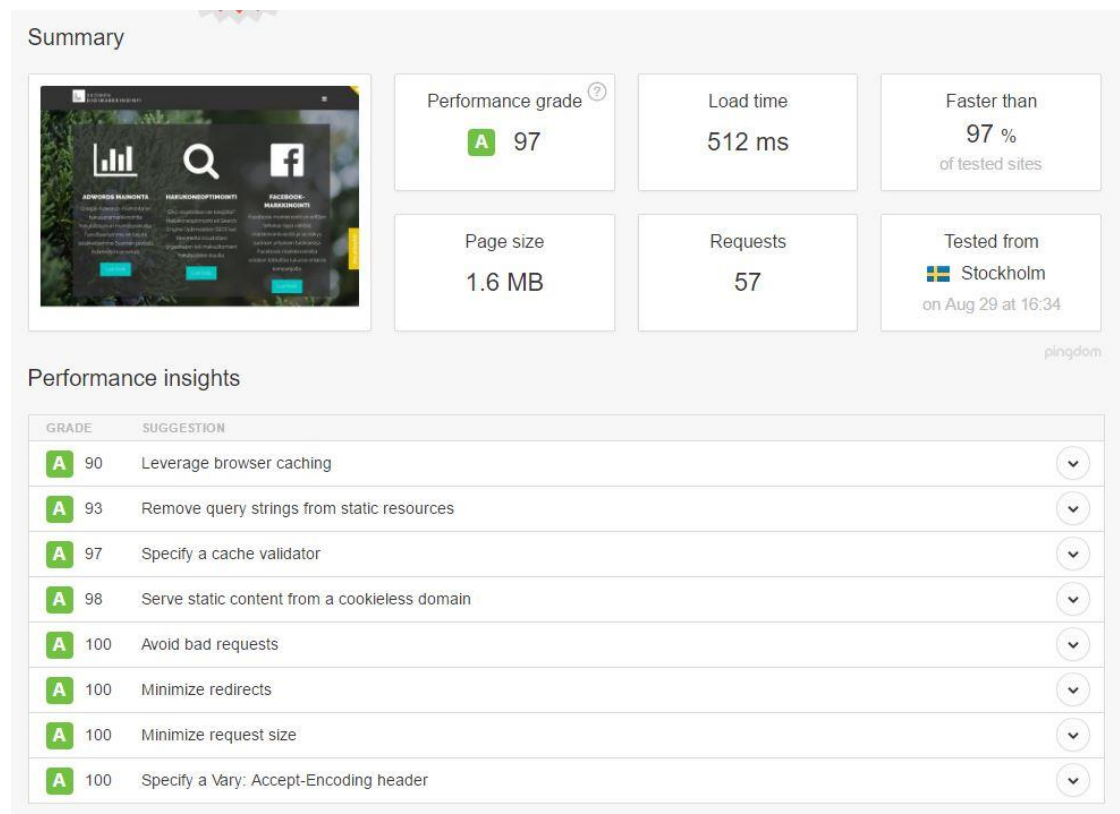
Kuvio 12. Vanhalle verkkosivustolle tehty nopeusanalyysi pöytäkoneversiosta



Kuvio 13. Uudelle verkkosivustolle tehty nopeusanalyysi pöytäkoneversiosta

Projektin lopputulokseen olen tyytyväinen, sillä annetun ajan puitteissa tuskin olisi kokonaisempaa pakettia pystynyt yhden hengen voimin tekemään. Testien perusteella uudet verkkosivut ovat nopeuden kannalta paremmat vanhoihin sivuihin verrattuna. Ulkoisten pyyntöjen määrä on alle puolet siitä, mitä se oli vanhalla sivustolla. Tämä johtuu suurelta osin siitä, että Drupal-sivustolla oli käytössä paljon moduuleita, jotka tekivät pyyntöjä ulkoisille palvelimille. Uutta sivustoa tehtäessä käytettiin vain välttämättöimpiä ja parhaita kolmannen osapuolen lisäosia. Kaikki järkevästi aikataulun puitteisiin mahtuva toiminnallisuus tehtiin itse, mikä näkyy suorituskyvyssä (ks. Kuvio 14).

Niin kuin kaikkien yritysten verkkosivujen, myös tämän sivuston pääasiallinen tarkoitus on saada lisää maksavia asiakkaita. Konversio-optimointiin on kiinnitetty huomiota ja käytössä olevien statistiikka työkalujen avulla verkkosivuilla vierailijoiden käytöstä tullaan seuraamaan. Saadun datan perusteella sivuston optimointia jatketaan konversioprosentin kasvattamiseksi. Myös uusia ominaisuuksia ja toiminnallisuutta on helppo lisätä hyvin tehdyn pohjatyon ansiosta.



Kuvio 14. Pingdom speed –testin tulokset sivuston ollessa testipalvelimella

7 Johtopäätökset

Vaikka valmiita toteutuksia ja lisäosia löytyy Wordpress-ympäristöön paljon, ei kaikkea kannata tehdä muiden tekemien lisäosien avulla. Jokainen käytössä oleva lisäosa hidastaa sivun latautumista. Ei voida olla myöskään täysin varmoja siitä, tuleeko lisäosan tekijä ylläpitämään ja päivittämään lisäosaa. Jos lisäosaa ei pidetä ajantasalla, on vaarana lisäosan lakkaaminen toimimasta uusien Wordpress-versioiden kanssa. Ei voida olla myöskään täysin varmoja siitä, miten lisäosa toimii ja hakeeko se esimerkiksi tyylitiedostoja ulkoisilta palvelimilta, mikä myös osaltaan hidastaa sivun latautumista. Varsinkin Google Tag manager -lisäosan tekemien uudelleenohjausten vaikutus sivustonlatausnopeuteen yllätti. Käytännössä lisäosa esti sivuston latautumista ennen kuin sen tekemät pyynnöt olivat tehty ja sillä oli suuri prosentuaalinen osuus sivuston latausajasta.

Toteutuksessa päätettiin käyttää valmista teemaa ja muokata siitä käyttötarkoitukseen sopiva tiukan aikataulun vuoksi. Sivustoa rakennettaessa huomattiin kuitenkin, että valmis teema oli yllättävän kankea, eikä tämä sen pohjalta rakentaminen säästänyt niin paljon aikaa kuin oletettiin. Lapsiteemaan tehtiin muutoksia raskaalla kädellä, minkä ansiosta myös allekirjoittanut oppi paljon Wordpressin toiminnasta.

Hyvä suunnittelu ja tarkka vaatimusmäärittely helpottivat toteutuksen tekemistä huomattavasti. Sivuston julkaisu myöhästyi noin kaksiviikkoa alkuperäisestä suunnitelmasta, minkä takia sivuston testaamiseen jäi hieman suunniteltua vähemmän aikaa. Julkaisun lykkääminen kuitenkin kannatti, sillä lopputulos oli huomattavasti parempi ja yhtenäisempi paketti ylimääräisten tuntien jälkeen. Aikataulua sotkivat muut työn alla olleet projektit, vaatimusten muuttuminen matkan varrella sekä ajoittain huono aikataulut.

Suurimmat haasteet projektin aikana olivat tietokantojen tutkiminen ja sisältöjen siirto sisällönhallintaohjelmistosta toiseen ilman virheitä, mikä vei paljon aikaa projektin alkupuolella. Myös Url-osoitteiden saaminen vastaamaan täysin vanhaa sivustoa ja Skaalautuvuuden toimiminen kaikilla selaimilla ja näytön ko'oilla. Lopputulokseen olen tyytyväinen. Sivustosta tuli parempi, kuin suurimmalla osalla yrityksen kilpailijoista, sekä sivuston latausnopeuden että käyttöliittymän osalta. Konversio-optimointia on mietitty paljon ja sitä tullaan vielä parantamaan sen jälkeen, kun si-

vustolta saadaan enemmän kävijästatistiikkaa. Nähtäväksi jää onko sivuston uudistuksella huomattavaa vaikutusta esimerkiksi eri laskeutumissivujen hakukonenäkyvyyden kannalta.

Salatun yhteyden eli https:n käyttöönotto voisi olla yksi tulevaisuudessa suoritetuista toimenpiteistä. Tällä olisi positiivinen vaikutus hakukonenäkyvyyden kannalta pitkällä aikavälillä, vaikka sijoitukset hieman laskisivatkin lyhyellä tähtämellä URL-osoitteiden muuttuessa. Salattuun yhteyteen siirryttäessä taattaisiin myös se, ettei hallintapaneeliin kirjautuessa kirjautumistiedot lähde selkokielisenä käyttäjän koneelta ”internetin yli” palvelimelle. Salattu yhteys sallisi myös http/2 protokollan käytön, sillä nykyiset selaimet tukevat sitä vain HTTPS-protokollan yli. http/2 protokollan etuna on sen nopeus yleisesti käytössä olevaan http 1.1-versioon verrattuna.

Yksi tekijä, joka saattaa aiheuttaa ongelmia tulevaisuudessa, on Wordpressin niin sanottu sadan sivun rajoitus. Hallintapaneelissa minkä tahansa sisältötyypin välilehdelle mentäessä, Wordpress hakee kaikki kyseiseen sisältötyyppiin kuuluvat jäsenet tietokannasta. Tästä seuraa se, että sivujen määrän kasvaessa Wordpressin hallintapaneelin puoli hidastuu jatkuvasti. Ongelma liittyy Wordpressin ytimen toimintaan, eikä siihen ole tämän raportin kirjoitushetkellä näkyvissä selkeätä ratkaisua.

Vaativuusmäärittely saatiin täytettyä ja sivustolla on pyritty tekemään kaikki oikein, nopeimman ja helpoimman ratkaisun sijaan. Tämän ansiosta sivustoa on helppo ylläpitää ja sinne on helppo lisätä uusia ominaisuuksia. Sivuston hiomista ja konversio-optimointia jatketaan tulevaisuudessa muun muassa käytössä olevan kävijäseurannasta saadun tiedon perusteella.

Lähteet

About nodes. 2016. Drupal. Viitattu 1.9.2016.

<https://www.drupal.org/docs/7/nodes-content-types-and-fields/about-nodes>

Basecamp. 2016. Basecamp Ltd. Viitattu 19.9.2016. <https://basecamp.com/>

Bitbucket. 2016. Atlassian Ltd. Viitattu 30.8.2016. <https://bitbucket.org/>

CMS Usage Statistics. 2016. BuiltWith® Pty Ltd. Viitattu 25.5.2016.

<http://trends.builtwith.com/cms>

Conversion marketing. 2016. Wikipedia. Viitattu 1.9.2016.

https://en.wikipedia.org/wiki/Conversion_marketing#Conversion_rate

Drupal to WordPress migration explained. Another cup of coffee. 2014. Viitattu 1.6.2016. <http://anothercoffee.net/drupal-to-wordpress-migration-explained/>

White House now uses Drupal. Drupal. 2009. Viitattu 5.9.2016.

<https://www.drupal.org/node/613892>

Drupal Suomi. 2014. Suomen Drupal –yhdistyksen verkkosivut. Viitattu 23.5.2016.

<http://www.drupal.fi/>

GoogleInside Search. N,d. Google Inc. Viitattu 10.9.2016.

<https://www.google.com/insidesearch/howsearchworks/crawling-indexing.html>

Google Search Console Help. 2016. Google Inc. Viitattu 20.8.2016.

<https://support.google.com/webmasters/answer/156184?hl=en>

Google Webmaster Central Blog. 2015. Google Inc. Viitattu 22.08.2016.

<https://webmasters.googleblog.com/2015/02/finding-more-mobile-friendly-search.html>

Hakukoneoptimointi (SEO). 2016. Suomen Digimarkkinointi Oy. Viitattu 15.9.2016.

<http://www.digimarkkinointi.fi/hakukoneoptimointi-seo>

Hunt, B. 2011. Convert! : Designing Web Sites to Increase Traffic and Conversion. California, USA. New Riders publishing.

Joomla. 2016. Open Source Matters. inc. Viitattu 22.5.2016.

<https://www.joomla.org/core-features.html>.

JQuery. 2016. Wikipedia. Viitattu 1.9.2016. <https://en.wikipedia.org/wiki/JQuery>

Krug, S. 2006. Don't make me think! A common sense approach to web usability. Berkley, California, USA. New Riders publishing.

Mobile Marketing statistics compilation. 2016. Smart Insights Ltd. Viitattu 14.9.2016.

<http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>

Netmarketshare. 2016. Net Applications Inc. Viitattu 28.6.2016.

<https://www.netmarketshare.com/search-engine-market-share.aspx?qprid=4&qpcustomd=0>

Post Types. 2014. Wordpress Codex. Viitattu 10.8.2016.

https://codex.wordpress.org/Post_Types

Lee, J. 2013. Search Engine Watch. Viitattu 22.8.2016.

<https://searchenginewatch.com/sew/study/2276184/no-1-position-in-google-gets-33-of-search-traffic-study>

Saarinen, S. 2010. Tulos Helsinki Oy. Viitattu 10.9.2016.

<http://www.tulos.fi/artikkelit/tuhoaako-informaatioarkkitehtuuri-sivustosi-hakukonenakyyvyyden/>

Search engine optimization. N,d. Wikipedia. Viitattu 1.9.2016.

https://en.wikipedia.org/wiki/Search_engine_optimization

Search Engine Optimization Starter Guide. 2010. Google Inc. Viitattu 26.5.2016.

<http://static.googleusercontent.com/media/www.google.com/en/webmasters/docs/search-engine-optimization-starter-guide.pdf>

Vagrant. HashiCorp. 2016. Viitattu 2.9.2016. <https://www.vagrantup.com/docs/why-vagrant/>

Webmaster Guidelines. 2016. Google Inc. Viitattu 07.09.2016.

<https://support.google.com/webmasters/answer/35769?hl=en>

Williams, B., Damstra, D. & Stern, S. 2015. Professional WordPress Design and Development. Indianapolis, Indiana. John Wiley & Sons Inc.

Wordpress. 2016. Viitattu 20.8.2016. <https://wordpress.org/>

Liitteet

Liite 1. Sivuston Gzip-pakkaukseen käytetty koodi

```
<IfModule mod_deflate.c>
# Compress HTML, CSS, JavaScript, Text, XML and fonts
AddOutputFilterByType DEFLATE application/javascript
AddOutputFilterByType DEFLATE application/rss+xml
AddOutputFilterByType DEFLATE application/vnd.ms-fontobject
AddOutputFilterByType DEFLATE application/x-font
AddOutputFilterByType DEFLATE application/x-font-opentype
AddOutputFilterByType DEFLATE application/x-font-otf
AddOutputFilterByType DEFLATE application/x-font-truetype
AddOutputFilterByType DEFLATE application/x-font-ttf
AddOutputFilterByType DEFLATE application/x-javascript
AddOutputFilterByType DEFLATE application/xhtml+xml
AddOutputFilterByType DEFLATE application/xml
AddOutputFilterByType DEFLATE font/opentype
AddOutputFilterByType DEFLATE font/otf
AddOutputFilterByType DEFLATE font/ttf
AddOutputFilterByType DEFLATE image/svg+xml
AddOutputFilterByType DEFLATE image/x-icon
AddOutputFilterByType DEFLATE text/css
AddOutputFilterByType DEFLATE text/html
AddOutputFilterByType DEFLATE text/javascript
AddOutputFilterByType DEFLATE text/plain
AddOutputFilterByType DEFLATE text/xml

# Remove browser bugs (only needed for really old browsers)
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4.[0-9] no-gzip
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
Header append Vary User-Agent
</IfModule>
```

Liite 2. Sivuston Expire headerien lisäkseen käytetty koodi

```
<IfModule mod_expires.c>
# Enable expirations
ExpiresActive On
# Default directive
ExpiresDefault "access plus 1 month"
# My favicon
ExpiresByType image/x-icon "access plus 1 year"
# Images
ExpiresByType image/gif "access plus 1 month"
ExpiresByType image/png "access plus 1 month"
ExpiresByType image/jpg "access plus 1 month"
ExpiresByType image/jpeg "access plus 1 month"
# CSS
ExpiresByType text/css "access plus 1 month"
# Javascript
ExpiresByType application/javascript "access plus 1 year"
</IfModule>
```

Liite 3. Alapalveluiden listaaminen ja lisääminen shortcodeen

```
323 // =====
324 // Alapalveluiden lisäys shortcodeen [wpb_childpages] (käytetty sivupalkissa)
325 // =====
326 function list_child_pages() {
327     global $post;
328     if ( is_singular( 'palvelut' ) ) {
329         if ( $post->post_parent )
330             $children = wp_list_pages( 'menu_order&title_li=&child_of=' . $post->post_parent . '&post_type=palvelut&echo=0' );
331         else
332             $children = wp_list_pages( 'menu_order&title_li=&child_of=' . get_the_ID() . '&post_type=palvelut&echo=0' );
333         if ( $children ) {
334             $string = '<ul><b>' . $children . '</b></ul>';
335         }
336         return $string;
337     }
338 }
339 add_shortcode( 'childpages', 'list_child_pages' );
340
```

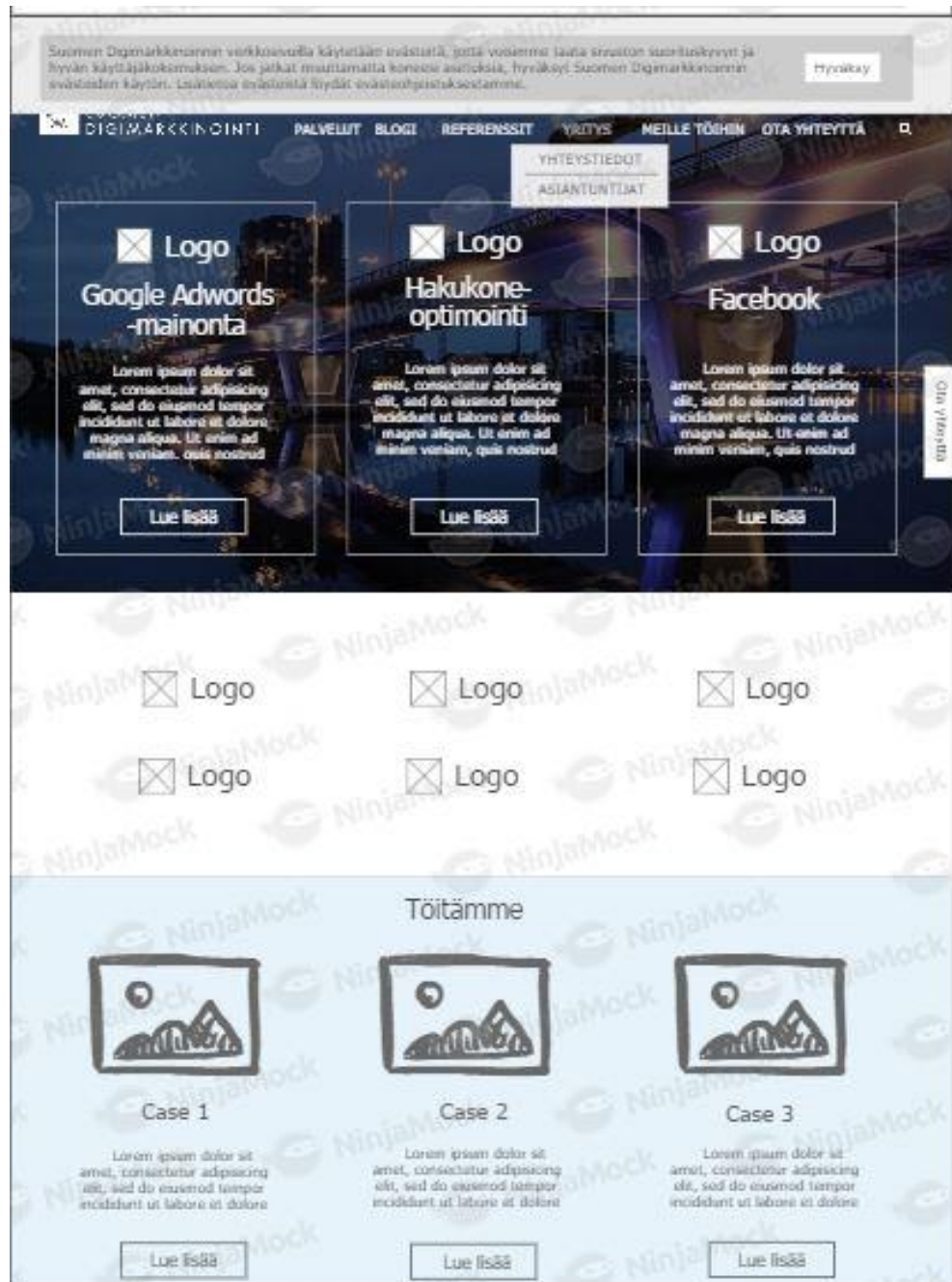
Liite 4. Palvelu-sisältötyypille oman kategorian rekisteröinti

```
34 function palvelu_kategoria_init() {
35     // create a new taxonomy
36     register_taxonomy(
37         'kategoria',
38         'palvelut',
39         array(
40             'label' => __( 'kategoria' ),
41             'rewrite' => array( 'slug' => 'kategoria' ),
42             'capabilities' => array(
43                 'manage_terms' => 'edit_posts',
44                 'edit_terms' => 'manage_categories',
45                 'delete_terms' => 'manage_categories',
46                 'assign_terms' => 'edit_posts'
47             )
48         )
49     );
50 }
51 add_action( 'init', 'palvelu_kategoria_init' );
```

Liite 5. Uuden sisältötyypin rekisteröinti

```
93 function palvelut () {
94
95     $labels = array (
96         'name' => 'Palvelut',
97         'singular_name' => 'Palvelu',
98         'add_new' => 'Lisää palvelu',
99         'all_items' => 'Kaikki palvelut',
100        'add_new_item' => 'Lisää palvelu',
101        'edit_item' => 'Muokkaa palvelua',
102        'new_item' => 'Uusi palvelu',
103        'view_item' => 'Katso palvelua',
104        'search_item' => 'Etsi palvelua',
105        'not_found' => 'Palvelua ei löydy',
106        'not_found_in_trash' => 'Ei löydy roskakorista',
107        'parent_item_colon' => 'Vanhempi
108
109    );
110    $args = array(
111        'menu_icon' => 'dashicons-format-aside',
112        'labels' => $labels,
113        'public' => true,
114        'has_archive' => true,
115        'publicly_queryable' => true,
116        'query_var' => true,
117        'rewrite' => array('slug' => '', 'with_front' => false, 'pages' => false),
118        'capability_type' => 'post',
119        'hierarchical' => true,
120        'supports' => array (
121            'title',
122            'editor',
123            'page-attributes',
124            'excerpt',
125            'thumbnail',
126            'revisions',
127
128        ),
129        'taxonomies' => array('post_tag', 'Palvelu-kategoria'),
130        'menu_position' => 2,
131        'exclude_from_search' => false,
132    );
133    register_post_type('palvelut', $args);
134
135 }
136 add_action('init', 'palvelut');
137
```

Liite 6. Etusivun rautalankamalli



Liite 7. Blogi-sivun rautalankamalli



Liite 8. Yksittäisen blogin rautalankamalli



Liite 9. Drupalin blogikirjoitusten siirto Wordpressintietokantaan

```

// Mae kaikki Drupalin postit WP:n tietokantaan
$sqlQuery = apply_filters( 'DrupalImport_query_nodes',
    "SELECT
    DISTINCT n.nid AS id,
    n.uid AS post_author,
    FROM_UNIXTIME(n.created) AS post_date,
    r.body_value AS post_content,
    n.title AS post_title,
    r.body_summary AS post_excerpt,
    n.type AS post_type,
    IF(n.status = 1, 'publish', 'draft') AS post_status,
    a.alias
    FROM ".$this->dbSettings['prefix'].".node n
    LEFT JOIN ".$this->dbSettings['prefix'].".field_data_body r ON (r.entity_id = n.nid)
    LEFT JOIN ".$this->dbSettings['prefix'].".url_alias a ON (REPLACE(a.source, 'node/', '') = n.nid)
    ORDER BY n.nid" );
$drupalContent = $this->drupalDB->results($sqlQuery);
if ( !empty($drupalContent) ) {
    // Jokaisen postin tallennus
    foreach($drupalContent as $dp) {
        $this->savePost2WP($dp);
    }
    // Drupalin kannassa tyhjiä välilehtiä ja tagien piisuminen
    $wpdb->query("UPDATE {$wpdb->posts} SET post_content = REPLACE(post_content, '<p> </p>', '');");
    $wpdb->query("UPDATE {$wpdb->posts} SET post_content = REPLACE(post_content, '<p class=\\"italic\\" > </p>', '');");
    print '<p><span style="color: green;">'.__( 'Content Imported', 'DrupalImport' ).'</span></p>';
} else {
    print '<p><span style="color: maroon;">'.__( 'Content Failed to Import', 'DrupalImport' ).' - '.__( 'Everything failed', 'DrupalImport' ).'</span></p>';
}
ob_flush(); flush(); // Output
}

```

Liite 10. Käyttäjien haku Drupalintietokannasta

```
// Käyttäjien haku Drupaalista
public function importUsers() {
  if (empty($this->options['users'])) {
    $totalUsers = $this->drupalDB->row("
      SELECT COUNT(1) as total
      FROM " . $this->dbSettings['prefix'] . "users u
      WHERE u.uid > 1
      " . (empty($this->options['enabled_users_only']) ? 'AND u.status = 1' : '') . "
      ORDER BY u.uid
    ");
    if (empty($totalUsers) && empty($totalUsers['total'])) {
      for($i=0; $i<=$totalUsers['total']; $i++) {
        $this->importUsersData($i);
      }
      print '<p><span style="color: green;">' . sprintf(__('Users Import Complete (%d imported)', 'DrupalImport'), $totalUsers['total']).'</span></p>';
      ob_flush(); flush(); // Output
    }
  }
  return $this;
}
?>
```