



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Verkkosivuston toteutus sisällönhallintajärjestelmällä

Nuorva, Eetu

2015 Laurea Kerava

Laurea-ammattikorkeakoulu
Laurea Kerava

Verkkosivuston toteutus sisällönhallintajärjestelmällä

Eetu Nuorva
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Joulukuu, 2015

Eetu Nuorva

Verkkosivuston toteutus sisällönhallintajärjestelmällä

Vuosi 2015 Sivumäärä 32

Toiminnallisessa opinnäytetyössä suunniteltiin ja toteutettiin verkkosivusto Jouluapua-yhteisölle, käyttäen apuna WordPress-sisällönhallintajärjestelmää. Jouluapua on jouluisin toimiva vapaaehtoisten yhteisö, joka pyrkii saattamaan yhteen apua tarvitsevat perheet ja apua tarvitsevat henkilöt. Jouluapua on toiminut aikaisemmin Facebookin kautta, mutta suosion kasvaessa toiminnan toteuttaminen Facebookin kautta on käynyt hankalaksi, jolloin uudet verkkosivut tulevat tarpeeseen.

Opinnäytetyössä kuvataan verkkosivustoprojektin kulku ja tutustutaan siinä käytettyyn WordPress-sisällönhallintajärjestelmään. Sisällönhallintajärjestelmät ovat nykyaikainen tapa toteuttaa verkkosivuja etenkin sellaisissa projekteissa, missä loppukäyttäjällä ei ole vahvaa HTML-osaamista. Sisällönhallintajärjestelmä tarjoaa helpon käyttöliittymän sisällön luomiseen ja ylläpitämiseen, eikä sen käyttö vaadi laajaa teknistä osaamista.

Projektin lopputuloksena oli www-sisällönhallintajärjestelmää käyttävä verkkosivusto, mikä julkaistiin vuoden 2015 lokakuun lopussa.

Asiasanat: WordPress, sisällönhallintajärjestelmä, verkkosivu

Eetu Nuorva

Creating web site with content management system

Year	2015	Pages	32
------	------	-------	----

The aim of this functional thesis was to design and create a web page for the Jouluapua-community using WordPress content management system. Jouluapua is a charity community working during Christmas, which aims to connect those who need help and those who could give help. Jouluapua has previously operated via a Facebook group, but as their popularity has grown lately, Facebook is no longer a suitable platform for the community. Hence the new web page is highly needed.

The thesis describes the flow of the web page process and explores the WordPress content management system, which was used in project. Content management systems are a modern way to produce web pages, especially in those projects where the end user doesn't have strong skills in HTML-language. A content management system provides an easy user interface to create and maintain content and it doesn't require wide technical skills.

The result of the project was a website using content management system. Site was published at the end of October 2015.

Keywords: WordPress, Content Management System, Web page

Sisällys

1	Johdanto	8
2	Opinnäytetyön lähtökohdat	8
2.1	Toimeksiantajan ja asiakkaan esittely	8
2.2	Tutkimuksen lähtökohdat ja tavoitteet	9
2.3	Aihealueen rajaukset	9
3	Verkkosivuston kehittäminen	10
3.1	Verkkosivuston kehityksessä käytetyt tekniikat	10
3.1.1	HTML.....	10
3.1.2	CSS	11
3.1.3	Javascript.....	11
3.1.4	PHP ja MySQL	11
3.2	Verkkosivustokehityksen vaihejakomallit	12
3.2.1	Vesiputousmalli	12
3.2.2	Prototyypimalli	13
3.2.3	Ketterät menetelmät	14
3.3	Responsiivinen suunnittelu	14
4	WordPress-sisällönhallintajärjestelmä	14
4.1	Sivut ja artikkelit.....	15
4.2	Teemat	16
4.3	Lisäosat	17
4.4	Vimpaimet	17
5	Tutkiminen ja tutkimusmenetelmät	18
5.1	Toimintatutkimus	19
5.2	Kvalitatiivinen ja kvantitatiivinen tutkimus	19
5.3	Validiteetti ja reliabiliteetti	19
5.4	Lähdekritiikki	20
6	Kehitysympäristö ja käytetyt työkalut	20
7	Projektin kulku ja toteutus.....	20
7.1	Vaatimusmäärittely	21
7.2	WordPressin asentaminen	22
7.3	Hallintapaneeli	23
7.4	Lisäosat	23
7.5	Sivurakenne ja täytesisältö.....	24
7.6	Teeman luominen	25
7.7	Sisällönsyöttö ja testaus	27
8	Sivuston käyttöönotto	27
9	Oma arviointi	28

10	Yhteenveto	28
	Lähteet	30
	Kuvat	32

Käsiteluettelo

HTML	Hypertext Markup Language. Merkkauskieli, millä verkkosivustot useimmiten esitetään. Verkkoselain lukee HTML:ää ja kuvantaa verkkosivun sen perusteella. (Korpela 2009.)
CSS	Cascading Style Sheets. HTML-dokumentin esitysasua kuvaava tyyliohjekieli. (W3C 2015.)
JavaScript	Dynaaminen oliopohjainen ohjelmointikieli, jota voidaan käyttää verkkosivustoilla (Saarikumpu 2006).
PHP	Hypertext Preprocessor. Palvelimen ajama ohjelmointikieli, jolla voidaan toteuttaa erilaisia verkkosovelluksia. (Heinisuo & Rauta 2007, 12-13.)
MySQL	Relaatiotietokantojen hallintajärjestelmä (Heinisuo & Rauta 2007, 37).
Sisällönhallintajärjestelmä	Työkalu, millä käyttäjä voi luoda ja hallita verkkosivustolla olevaa sisältöä (Kline 2012).

1 Johdanto

Sisällönhallintajärjestelmien käyttö on yleistynyt kovasti viime vuosien aikana niiden helppokäyttöisyyden ja matalan oppimiskynnyksen takia. Sisällönhallintajärjestelmää käyttäessä ei tarvitse vahvoja teknisiä taitoja eikä sen käyttö vaadi kokemusta verkkosivusto-ohjelmoinnista, mikä madaltaa monen kynnyksen perustaa verkkosivusto esimerkiksi yritykselle. Tämän toiminnallisen opinnäytetyön tarkoituksena oli suunnitella ja toteuttaa verkkosivusto asiakkaalle käyttäen sisällönhallintajärjestelmää ja raportoida kehitysprosessi mahdollisimman kattavasti.

Projektin raportoinnin lisäksi opinnäytetyössä käydään lyhyesti läpi WordPress-sisällönhallintajärjestelmää, jota käytettiin myös opinnäytetyön asiakasprojektissa. Sen lisäksi opinnäytetyössä tutustutaan yleisesti verkkosivuston kehittämiseen ja sen takan oleviin tekniikkoihin. Verkkosivustojen kehittäminen muuttui rajusti älypuhelimien ja tablettien suosion myötä ja nykyään sivuston pitää olla luettavissa monilla erikokoisilla laitteilla. Opinnäytetyössä tutustutaan myös responsiiviseen suunnitteluun, minkä avulla yksi verkkosivu saadaan toimimaan esteettömästi kaikilla laitteilla.

Opinnäytetyön toimeksiantaja toimi digitaalisia ratkaisuja tarjoava W3 Group Finland Oy. Asiakkaana projektissa toimi Jouluapua-yhteisö, mikä pyrkii joulun alla yhdistämään apua tarvitsevat perheet ja halukkaat avunantajat. Verkkosivustoprojekti ajoitettiin alkusyksylle 2015 ja valmis verkkosivusto otettiin käyttöön lokakuun alussa.

2 Opinnäytetyön lähtökohdat

Aihe opinnäytetyöhöni tuli työpaikaltani W3 Groupilta. Tarkoitukseni oli toteuttaa verkkosivustoprojekti joka tapauksessa, joten opinnäytetyön tekeminen tästä samaisesta projektista kävi erittäin hyvin. Tämä oli myös ensimmäinen projekti, missä työskentelin täysin yksin ja vastasin itse sen kaikista osa-alueista.

2.1 Toimeksiantajan ja asiakkaan esittely

Opinnäytetyön toimeksiantaja oli työnantajani W3 Group Finland Oy (lyhyesti W3 tai W3 Group). W3 Group on sähköisen liiketoiminnan palveluita tarjoava yritys, joka tuottaa mm. tietojärjestelmiä ja verkkosivustoja.

Syksyllä 2014 W3 Groupilla oli ”Kuka tarvitsee uudet nettisivut” -niminen joululahjakampanja, missä ihmiset saivat ehdottaa kerhoja, yhdistyksiä, henkilöitä, yrityksiä tai vastaavia tahoja,

ketkä heidän mielestensä tarvitsevat uudet verkkosivut. Ehdotettuja verkkosivuja pystyi äänestämään W3 Groupin nettisivuilla ja eniten ääniä kerännyt ehdotus palkitaan hyväntekeväisyystyönä tehtyinä verkkosivuina. Äänestyksen voitti ylivoimaisesti Heidi Miettisen Jouluapua-ryhmä, mikä toimi tämän opinnäytetyön asiakkaana.

Jouluapua on tällä hetkellä Facebookissa toimiva ryhmä, mikä pyrkii yhdistämään sosiaalisen median kautta ne, jotka tarvitsevat apua ja ne, jotka voivat tarjota apua. Jouluapua-ryhmällä ei ole omia verkkosivuja, vaan kaikki toiminta on tapahtunut Facebookin kautta.

2.2 Tutkimuksen lähtökohdat ja tavoitteet

Tämän toiminnallisen opinnäytetyön tavoitteena oli suunnitella ja toteuttaa Jouluapua-ryhmän verkkosivustoprojekti siitä samalla raportoiden. Valmiissa opinnäytetyössä käydään projektin raportoinnin lisäksi lyhyesti läpi WordPress-sisällönhallintajärjestelmän toimintaa ja avataan erilaisia valintoja web-kehityksen taustalta.

Jouluapua-ryhmällä ei ole tällä hetkellä omia verkkosivuja, sillä ryhmä on toiminut Facebookissa. Ryhmä on saanut viime vuosina huomasti suosiota ja sen toiminnan ohjaaminen Facebookissa on alkanut käydä vaikeaksi. Projektissa toteutettavan verkkosivuston tavoitteena on helpottaa Jouluapua-ryhmän takana olevien henkilöiden taakkaa sekä tarjota käyttäjille entistä helpompi keino pyytää ja antaa apua joulun alla. Tarkoituksena on, että uudet verkkosivut saadaan käyttöön joulukuksi 2015.

Oma henkilökohtainen tavoitteeni oli syventää omaa WordPress- sekä web-kehitysosaamista ja oppia asiakasprojektin vähemmän teknisiä yksityiskohtia, kuten projektinhallintaa. Teknistä toteutusta olen tehnyt paljon aikaisemminkin, mutta projektinhallintaa en juuri ollenkaan.

2.3 Aihealueen rajaukset

Opinnäytetyössä keskitytään toteutetun asiakasprojektin kuvaamiseen. Vaikka WordPress-sisällönhallintajärjestelmä on siinä isossa osassa, työn ei ole tarkoitus olla kaiken kattava WordPress-manuaali, vaan siinä käydään ko. sisällönhallintajärjestelmää niiltä osin läpi, mitä koen tarpeelliseksi projektin kannalta. Toisin kuin monissa samaa aihetta käsittelevissä opinnäytetöissä, en käy tässä läpi eri sisällönhallintajärjestelmiä, vaan keskityn pelkästään WordPressiin.

Myös itse verkkosivustoprojektissa oli muutamia rajoituksia: en piirrä sivuston ulkoasua itse, vaan saan työpaikkamme graafikolta valmiiksi piirretyn ulkoasun. En myöskään tuota itse sisältöä, vaan saan valmiit tekstit asiakkaalta, mitkä syötän sivulle.

3 Verkkosivuston kehittäminen

Verkkosivuston kehittäminen (lyhyesti web-kehitys) käsittää verkkosivuston luomisen, rakentamisen ja ylläpitämisen. Prosessi voidaan jakaa erinäisiin osa-alueisiin, kuten verkkosivusto-ohjelmointiin, verkkosivustosuunnitteluun, sisällönjulkaisuun ja tietokantahallintaan. Verkkosivusto-ohjelmointi on erinäisten ohjelmointi- ja merkintäkielin kirjoittamista, mistä kerron tarkemmin seuraavassa luvussa. Verkkosivustosuunnittelu sisältää myös ohjelmointia, mutta keskittyy enimmäkseen sivuston ulkoasuun ja käytettävyyteen, eli siihen osaan joka näkyy loppukäyttäjälle. Sisällönjulkaisu keskittyy ainoastaan verkkosivulla olevan sisällön tuottamiseen ja ylläpitämiseen, eikä ota kantaa tekniseen toteutukseen. Tietokantahallinta taas huolehtii tietokannoista, mihin verkkosivuston dataa voidaan tallentaa tehokkaasti. (TechTerms 2009.)

3.1 Verkkosivuston kehityksessä käytetyt tekniikat

Verkkosivustokehitystä varten on olemassa liuta erilaisia tekniikoita. Tekniikoiden käyttö riippuu yleisesti siitä, mikä käyttötarkoitus sivulla on. Staattinen verkkosivusto voi olla tekniikaltaan varsin yksinkertainen, kun taas dynaaminen, käyttäjän kanssa kommunikoiva sivusto vaatii tiedon käsittelyyn taipuvampia tekniikoita.

Tässä alaluvussa käyn läpi yleisempiä ja mielestäni tärkeimpiä web-kehityksessä käytettyjä tekniikoita. Kaikkia luvussa esiteltyjä tekniikoita käytettiin myös tämän opinnäytetyön asiakasprojektissa.

3.1.1 HTML

Hypertext Markup Language eli HTML on vahvasti standardoitu kuvauskieli, jolla verkkosivustot useimmiten rakennetaan. HTML kuvaa verkkosivuston rakennetta, minkä perusteella verkkoselain luo loppukäyttäjälle näytetyn verkkosivuston. HTML ajetaan aina käyttäjän selaimessa, minkä takia verkkosivuston käyttäjä pystyy aina lukemaan selaamansa sivun lähdekoodin. (Korpela 2009.)

HTML5 on HTML:n uusin versio, minkä standardit hyväksyttiin virallisesti vuonna 2014. Uusi versio tarjoaa paljon uusia elementtejä etenkin mediaelementtien näyttämiseen ja tekstin parempaan jäsentelyyn. (Mozilla Foundation 2015.)

3.1.2 CSS

Cascading Style Sheets eli CSS on tyyliohjekieli, mikä kuvaa HTML-dokumentin esitysraakenteen. Käytännössä jokaisen verkkosivuston graafinen ulkoasu on CSS-pohjainen ja ilman sitä verkkosivustot olisivat vain mustaa tekstiä valkoisella pohjalla. CSS:n avulla voidaan vaikuttaa kaikkiin HTML-elementtien graafisiin piirteisiin, kuten esimerkiksi leipätekstin kokoon, palstojen leveyteen ja linkkien väreihin. (W3C 2015.)

CSS:n viimeisin versio on CSS3, minkä merkittävimpänä ominaisuutena lienee media query-ominaisuus, minkä avulla voidaan tehokkaasti määrittellä erilaisia ulkoasuyleilyjä erilaisille ja erikokoisille näytöille, mikä on responsiivisuuden kannalta merkittävä ominaisuus (Boudreaux 2012). Verkkosivuston elementit voidaan esimerkiksi määrittellä media queryn avulla niin, että kun sivustoa katsotaan tarpeeksi pieneltä näytöltä, rinnakkain olevat elementit muuttuvat pystysuuntaisiksi elementeiksi, jolloin sivusto mukautuu paremmin pienelle näytölle.

3.1.3 Javascript

Javascript on dynaaminen oliopohjainen ohjelmointikieli, jota voidaan käyttää niin verkkosivustoilla kuin palvelinpuolella. Javascript on kevyt ja käyttöjärjestelmäriippumaton, minkä takia se sopii hyvin myös mobiililaitteille. (Saarikumpu 2006.)

Javascriptillä verkkosivustoon saa luotua dynaamisuutta; esimerkiksi tässä asiakasprojektissa sivustolla olevan lomakkeen tekstikentät tarkastetaan javascriptillä ennen lähettämistä. Jos javascript huomaa, että esimerkiksi sähköpostiosoitteelle varatussa kentässä ei ole sähköpostiosoitetta, se tulostaa virheilmoituksen eikä lähetä lomaketta eteenpäin.

3.1.4 PHP ja MySQL

Hypertext Preprocessor eli PHP on ohjelmointikieli, jota voidaan javascriptin tapaan käyttää dynaamisten verkkosivujen luomisessa. Toisin kuin javascript ja HTML, PHP on palvelimella ajettava kieli ja se vaatii palvelimelle erikseen asennettavan PHP-tulkin. (Heinisuo & Rauta 2007, 12-13.)

MySQL on avoimen lähdekoodin relaatiotietokantaohjelmisto ja yleisesti käytetty tietokanta. Tietokannat toimivat laajemmissa verkkosivuissa ja -sovelluksissa paikkana, minne palvelun

käyttämää tietoa voidaan varastoida ja mistä sitä voidaan hakea. Tieto on järjestetty relaatiotietokannoissa järjestetty sarakkeista ja soluista koostuviin tauluihin, mitkä voivat muodostaa keskinäisiä suhteita eli relaatioita. (Heinisuo & Rauta 2007, 37-38.)

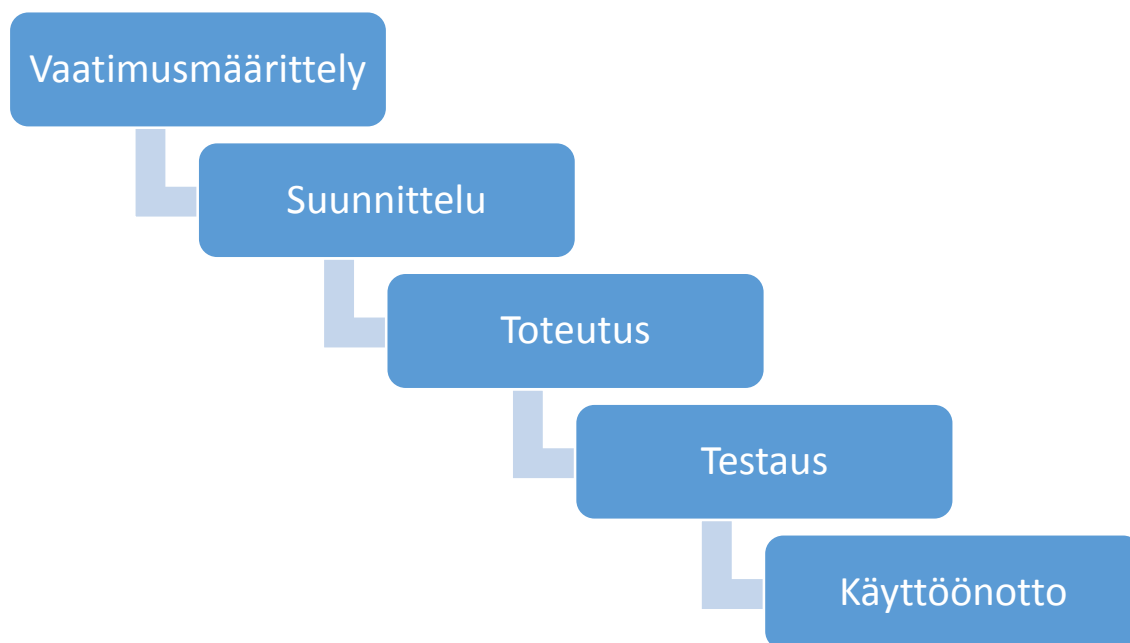
Sisällönhallintajärjestelmissä tietokantoja käytetään sisällön tallentamiseen, minkä ansiosta sivuston ulkoasu ja sisältö saadaan pidettyä erillään toisistaan.

3.2 Verkkosivustokehityksen vaihejakomallit

Verkkosivuston tai ohjelmiston kehitystyö voidaan jakaa pienempiin vaiheisiin. Tätä vaihejakoa kutsutaan vaihejakomalliksi. Tässä aluvuossa kerron vesiputousmallista, mitä käytettiin myös tämän opinnäytetyön projektissa, sekä muista yleisimmistä vaihejakomalleista.

3.2.1 Vesiputousmalli

Vesiputousmalli on yksi eniten käytetyistä vaihejakomalleista. Vesiputousmallissa kehitysvaiheet etenevät lineaarisesti vesiputouksen lailla, niin että seuraava kehitysvaihetta ei voida aloittaa, ennen kuin aikaisempi vaihe on saatettu loppuun. Kun yksi kehitysvaihe on toteutettu, siihen ei enää palata. Kuvassa 1 on esitetty tässä opinnäytetyössä tehdyn verkkosivustoprojektin vesiputousmalli. (Hughey 2007.)



Kuva 1: Jouluapua-projektin vesiputousmalli

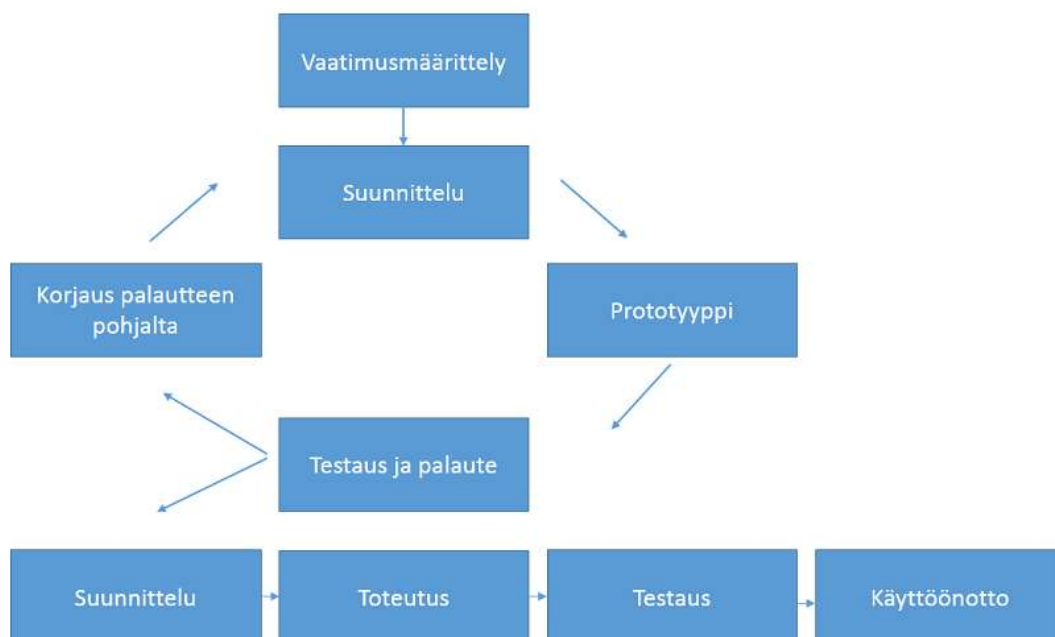
Vesiputousmallin vahvuus on sen yksinkertaisuudessa. Malli on selkeä ja helppo ymmärtää, kun jokaisella vaiheella on tarkka alku ja loppu. Yksisuuntaisen etenemisen ansiosta projektinhallinta on helppoa, kun eri vaiheet eivät mene päällekkäin. (Hughey 2007.)

Vesiputousmallin huonona puolena pidetään sen joustamattomuutta ja kykyä vastata huonosti muuttuneisiin vaatimuksiin. Kaikkia projektin vaatimuksia voidaan harvoin määrittellä ensimmäisessä vaatimusmäärittelyssä, minkä takia vaatimuksia joudutaan muuttamaan projektin myöhemmissä vaiheissa, mitä vesiputousmalli ei salli. Tämän takia vesiputousmalli soveltuu erityisesti pieniin projekteihin, missä vaatimukset ovat selkeät ja tarkoin määritellyt alusta asti. (Hughey 2007.)

Tämän opinnäytetyön projektissa käytetty vaihejakomalli oli myös vesiputousmalli, mitä ei tosin noudatettu aivan täydellisesti. Toteutus-, testaus- ja käyttöönottovaiheissa palattiin vä- lillä aikaisempiin vaiheisiin korjaamaan ilmenneitä virheitä, mitä ei pitäisi tapahtua täydelli- sessä vesiputousmallissa.

3.2.2 Prototyypimalli

Prototyypimallissa kehitettävästä tuotteesta luodaan toiminnallinen prototyyppi tiedossa olevien vaatimusten pohjalta. Vaatimusmäärittelyä täydennetään käyttämällä ja kokeilemalla prototyyppiä. Prototyypimalli etenee iteratiivisesti, kuten kuva 2 havainnollistaa: Ensin pro- totyyppi luodaan nopeasti, minkä jälkeen sitä testataan kehittäjien ja asiakkaan toimesta. Prototyyppiä korjataan testauksesta saadun palautteen perusteella ja testataan uudestaan. Prototyyppiä iteroidaan niin kauan, kunnes se saa asiakkaan hyväksynnän, minkä jälkeen se toteutetaan. (Bangwal 2012.)



Kuva 2: Esimerkki prototyypimallista

Nopea palautteen saaminen on yksi prototyypimallin eduista. Toisin kuin vesiputousmallissa, prototyypimallissa palautetta saadaan jokaisen iteraatiokierroksen jälkeen, jolloin siihen voidaan reagoida jo projektin varhaisissa vaiheissa. Konkreettinen prototyyppi auttaa asiakasta hahmottamaan paremmin, millainen lopullinen tuote tulee olemaan. Toisaalta asiakkaalle pitää myös korostaa, että prototyyppi ei ole valmis tuote. (Bangwal 2012.)

3.2.3 Ketterät menetelmät

Aikaisemmissa luvuissa kuvailtu vesiputousmalli ja prototyypimalli ovat yleisesti heikkoja reagoimaan nopeasti muuttuviin määrityksiin. Ketterät menetelmät ovat joukko menetelmiä, mitkä on kehitetty vaihtoehdoksi paljolti käytetylle vesiputousmallille. Ketterä menetelmissä kehitystyö tapahtuu tiukissa iteraatioissa niin, että jokaisen iteraationkierroksen lopussa arvioidaan seuraavan kierroksen vaatimukset. Tällaisella työskentelytavalla voidaan reagoida nopeasti kesken projektia muuttuviin vaatimuksiin. (Habib 2013.)

Ketterien menetelmien heikkoudeksi sanotaan niiden ennalta-arvaamattomuus. Kun projektia määritellään uudelleen iteraatio kerrallaan, sille voi olla hankala määritellä tarkkaa budjettia ja valmistumispäivämäärää. (Waters 2015.)

3.3 Responsiivinen suunnittelu

Responsiivinen suunnittelu on käyttöliittymän suunnittelutapa, missä otetaan huomioon kaikki erikokoiset laitteet ja näytöt. Responsiivista verkkosivua voidaan käyttää ilman tarvetta tehdä erillisiä versioita sivusta eri laitteille; sivustoa voidaan käyttää hyvällä käyttökokemuksella huolimatta siitä, onko käyttäjällä pöytätietokone, kännykkä tai tabletti. (Marcotte 2010.)

Mobiililaitteita käytetään nykypäivänä lähes yhtä paljon kuin pöytätietokoneita ja kannettavia tietokoneita. Joissakin maissa, esimerkiksi Yhdysvalloissa mobiilikäyttäjät ovat jo enemmistö, joten mistään pienestä ilmiöstä ei voida puhua. (Bosomworth 2015.) Mobiilikäyttäjät on pakko ottaa huomioon verkkosivua suunnitellessa. Mobiililaitteet eivät lataa sivuja yhtä nopeasti, kuin työpöytäkoneet, joten sivuston sisältö pitää optimoida tehokkaasti ja ylisuuret kuvat jättää pois. Laitteiden ruudut ovat pieniä, joten klikattavien elementtien, kuten navigaatiolinkkien, pitäisi olla mahdollisimman suuria. (Rocheleau 2012.)

4 WordPress-sisällönhallintajärjestelmä

Verkkosivuston sisällönhallintajärjestelmä (engl. Content Management System, CMS) on työkalu, minkä avulla käyttäjä (useimmissa tapauksissa sivuston ylläpitäjä) voi luoda ja hallita

sivustolla esitettävää sisältöä graafisen käyttöliittymän avulla (Kline 2012). Yksi keskeisimmistä sisällönhallintajärjestelmien tuomista hyödyistä on se, että sen yksinkertainen käyttäminen ei tarvitse erityistä HTML:n, CSS:n tai muiden web-teknologioiden tuntemusta. (Kline 2013).

Web-kehittäjän näkökulmasta sisällönhallintajärjestelmä vähentää sivuston ylläpitotaakkaa, kun asiakas, kenelle sivusto on tehty, pystyy itse lisäämään ja muokkaamaan sisältöä. Lisäksi, kun sisällönhallintajärjestelmä tarjoaa valmiin koodipohjan, kehittäjä voi keskittyä oleellisempiin asioihin, kuten graafisen ulkoasun tai yksilöidyn toiminnallisuuden kehittämiseen.

WordPress on avoimen lähdekoodin sisällönhallintajärjestelmä. Se julkaistiin alun perin vuonna 2003 blogialustaksi, mistä se kehittyi pian kaikenlaiset verkkosivustot kattavaksi sisällönhallintajärjestelmäksi. Kehityskaaren aikana WordPress sai monia nykyisen WordPress-kehityksen kannalta tärkeitä ominaisuuksia, kuten teemat ja lisäosat, mistä lisää seuraavaksi. (WordPress 2015a.)

4.1 Sivut ja artikkelit

WordPressin sisältö jaetaan kahteen osaan: sivuihin ja artikkeleihin. Sivut ovat staattista sisältöä, minkä ei oleteta päivittyvän kovin useasti. Kuvitteellisen yrityksen sivuja voisi olla esimerkiksi etusivu, tuotesivu, ajankohtaista-sivu sekä yritystietoa sisältävä sivu. WordPressin sivut ovat hierarkkisia, eli sivuilla voi olla useita alasivuja. Yritystiedot-sivulla voisi olla esimerkiksi Rekrytointi-alasivu. Sivujen hierarkia selkeyttää sivurakennetta ja auttaa esimerkiksi Googlea indeksoimaan sivut tehokkaammin. (Fylan 2015.)

Artikkelit ovat dynaamista, useasti päivittyvää sisältöä. Artikkelit voivat olla esimerkiksi tiedotteita, blogikirjoituksia tai uutiskirjeitä. Artikkeleita on mahdollista ryhmitellä kategorioiden avulla. Jos sivustolla on uutisia sekä blogikirjoituksia, uutisartikkeleita varten voidaan luoda uutiset-kategoria. Kun jossakin kohtaa sivustoa halutaan listata kaikki uutiset, riittää haetaan kaikki uutiskategoriaan lisätyt artikkelit. (Fylan 2015.)

WordPressissa on myös mahdollista luoda mukautettuja sisältötyyppejä (engl. custom post type). Mukautettuja sisältötyyppejä käytetään yleensä, kun luodaan dynaamista sisältöä, mikä ei sovi sivuksi, eikä sitä saada eriytettyä kategorioiden avulla tarpeeksi muista artikkeleista. Mukautettuja sisältötyyppejä luodaan `register_post_type()`-funktion avulla teeman `functions.php`-tiedostossa tai mahdollisesti jossakin lisäosan tiedostoista, jos mukautettu sisältötyyppi luodaan lisäosan kautta. Mukautetulla sisältötyypillä voi olla sekä sivun että artikkelin

ominaisuuksia. Esimerkiksi monet tapahtumakalenteri-lisäosat käyttävät mukautettua sisältötyyppiä tapahtumien tallentamiseen. Tällöin oletussisältötyyppien rinnalle tulee Tapahtumasisältötyyppi. (WordPress Codex 2015^c.)

4.2 Teemat

WordPress-sivuston ulkoasua ja sen toiminnallisuutta kutsutaan teemaksi (engl. theme).

Teema koostuu CSS-tyylitiedostosta, missä määritellään eri elementtien tyylit, esimerkiksi otsikon fontti, infolaatikon leveys tai leipätekstin marginaali. Tyylitiedoston lisäksi teemaan kuuluu sivupohjia (engl. template). Sivupohjat ovat PHP-tiedostoja, joiden pohjalta yksittäiset sivut luodaan, kun käyttäjä hakee niitä. Itse sisältöä sivupohjissa ei ole, vaan niiden paikka merkitään sivupohjaan omilla PHP-funktiolla. Teema sisältää myös functions.php-tiedoston, missä voidaan määritellä omia PHP-funktioita teemassa käytettäväksi. (Hedengren 2010, 58-59.)

Teema on yleensä pilkottu vähintään ylä-, ala- ja keskiosiin. Kuvassa 3 näkyy yksinkertainen pohjatiedosto. `get_header();` -funktio hakee teeman yläosan, mikä on määritelty `header.php` -nimisessä tiedostossa. `get_footer();` puolestaan teeman alaosan. Keskellä olevan `while`-loopin sisällä olevat funktiot `the_title()` ja `the_content()` tulostavat sen hetkisen sivun otsikon ja sisällön.

```

1  <?php get_header(); ?>
2
3  <div id="middle">
4  <div id="middle-content" class="max-width">
5
6      <?php while(have_posts()): the_post(); ?>
7
8          <article class="post">
9              <div class="title">
10                 <h1><?php echo the_title(); ?></h1>
11             </div>
12             <div class="content">
13                 <?php echo the_content(); ?>
14             </div>
15         </article>
16
17     <?php endwhile; ?>
18
19 </div>
20 </div>
21
22 <?php get_footer(); ?>
23
24

```

Kuva 3: Yksinkertainen pohjatiedosto

WordPressissa on mahdollista käyttää pääteeman lisäksi alateemoja (engl. Child theme). Alateema on oma itsenäinen teema, mutta se perii pääteeman (engl. Parent theme) ominaisuudet, ellei toisin määritellä. Jos pääteemassa on määritetty, että h1-otsikon koko on 24 pikseliä ja sen väri on harmaa, niin lapsiteema käyttää tätä samaa tyylimäärittelyä. Tyylimäärittelyitä voi kuitenkin yliajaa ja alateeman h1-otsikon väriksi voidaan määrittellä vaikkapa musta. Tällöin alateeman otsikko käyttää vanhemmalta perittyä fonttikokoa 24px, mutta omaa fonttiväriä musta. (Hedengren 2010, 104-107.)

4.3 Lisäosat

WordPress itsessään on melko kevyt julkaisujärjestelmä eikä se sisällä juurikaan erityistä lisätoiminnallisuutta. Lisäosat (engl. Plugin) ovat pieniä koodipaketteja, joita asentamalla WordPressiin voidaan tuoda haluamiaan ominaisuuksia. (WordPress Codex 2015a.) Esimerkiksi yksi WordPressin lisäosahakemiston suosituimmista lisäosista on Contact Form 7 -lisäosa, jonka avulla käyttäjä saa helposti luotua haluamiansa lomakkeita sivulle. (WordPress 2015b.)

WordPressin hallintapaneelista löytyy yksinkertainen käyttöliittymä lisäosien hallintaan, minkä kautta lisäosia voi lisätä, poistaa ja aktivoida. Tuoreen WordPress-asennuksen mukana tulee kaksi lisäosaa: Akismet, mikä auttaa torjumaan sivustolle suuntautuvaa roskapostia, sekä Hello Dolly, mikä tulostaa Louis Armstrongin Hello Dolly -kappaleen sanoja hallintapaneelin ylälaitaan.

Kehittäjän näkökulmasta lisäosat virtaviivaistavat työprosessia ja säästävät aikaa. Palautelomaketta tai muuta vastaavaa toiminnallisuutta ei aina kannata lähteä koodaamaan itse, kun saatavilla on ilmainen ja satoja kertoja hyväksi todettu lisäosa. Lisäosat ovat myös täysin avointa lähdekoodia, eli kuka tahansa voi tutkia tarkasti, miten lisäosa toimii ja minkälaista koodia sen taustalta löytyy.

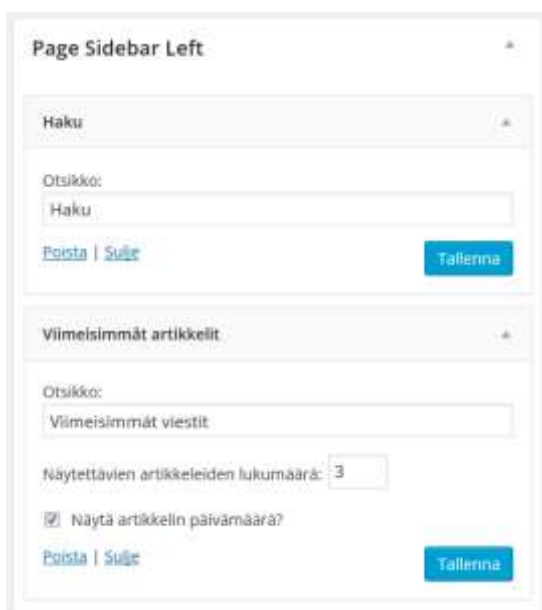
Lisäosia käytettiin hyödyksi myös tässä verkkosivustoprojektissa. Erityisesti PageBuilder-lisäosan ansiosta kehitystyössä säästettiin lukuisia, mahdollisesti jopa kymmeniä henkilötyötunteja vähentyneen ohjelmointitarpeen ansiosta.

4.4 Vimpaimet

Vimpaimet (engl. Widget) ovat WordPressin sisältöön upotettavia koodinpätkiä, mitkä toteuttavat tietyn, ennalta määrätyn tehtävän. WordPressin oletusasennuksen mukana tulee lukuisia vimpaimia sekä monet lisäosat tuovat niitä mukanaan lisää. Vimpain voi esimerkiksi tulostaa viimeisimmän artikkelin tai näyttää hakupalkin. (WordPress Codex 2015^b.)

Vimpaimia varten pitää määrittellä vimpainalue. Vimpainalue voidaan määrittellä mihin tahansa kohtaan teemasta esimerkiksi sivuston alaosaan, jolloin vimpainalueelle lisätyt vimpaimet näkyvät jokaisen sivun alalaidassa. Määrittely tapahtuu lisäämällä teeman pohjatiedostoon PHP-funktio `dynamic_sidebar("vimpainalueen_id")`. Määritellyt vimpainalueet listautuvat WordPressin hallintapaneeliin, missä niihin voi lisätä uusia vimpaimia. (WordPress Codex 2015^b.)

Kuvassa 4 on ”Page Sidebar Left” -niminen vimpainalue, mihin on lisätty Haku- ja Viimeisimmät artikkelit-vimpaimet. Tässä tapauksessa vimpaimissa oli tarjolla muutamia asetuksia, minkä avulla niiden toimintaa sai tarkennettua. Lopputuloksena jokaisella sivulla, missä sivupalkki on käytössä, näkyy hakukenttä sekä kolme viimeisintä artikkelia päivämäärineen. Mikäli sivupalkkia halutaan muokata esimerkiksi näyttämään viisi uusinta artikkelia kolmen sijaan, se onnistuu helposti vimpaimen asetuksia muuttamalla ilman, että muutoksia tarvitsee tehdä manuaalisesti jokaiselle sivulle.



Kuva 4: Kuva vimpainalueesta, mihin on lisätty hakukentän ja kolmen viimeisimmän artikkelin näyttävät vimpaimet

5 Tutkiminen ja tutkimusmenetelmät

Tämä opinnäytetyö on toiminnallinen opinnäytetyö, eikä se sisällä varsinaista tutkimusta tai tutkimuskysymystä. Toiminnallinen opinnäytetyö on kehittämistyö, missä tavoitteellaan ammatillisessa kentässä käytännön toiminnan ohjeistamista ja opastamista (Airaksinen 2009). Opinnäytetyössä tehtävä tutkimus on luonteeltaan toimintatutkimus, missä on kvalitatiivisen tutkimuksen piirteitä.

5.1 Toimintatutkimus

Toimintatutkimus on prosessi, mikä tähtää konkreettisten ongelmien ratkaisuun ja paremman tuotteen, palvelun tai toiminnon kehittämiseen. Tässä tapauksessa kehitettävä tuote on asiakkaan verkkosivusto.

Toimintatutkimuksessa tutkimuksen tekijä on tiiviissä yhteistyössä yhteistyökumppanin kanssa; toimintatutkimusta ei pysty tekemään yksin. Toimintatutkimuksen välivaiheita arvioidaan yhdessä yhteistyökumppanin kanssa ja valmis tutkimus esittää uutta tietoa, jonka pätevyyttä arvioidaan keskustelun ja kokeilun kautta sekä rinnastamalla sitä aikaisempaan tietoon. (Ruohonen 2015.)

5.2 Kvalitatiivinen ja kvantitatiivinen tutkimus

Kvantitatiivinen eli määrällinen tutkimusmenetelmä on tutkimuksessa käytettävä menetelmä, mikä perustuu rajatun aineiston mitattavissa olevien muuttujien tutkimiseen ja laskemiseen. Muuttujia voivat olla esimerkiksi ikä, sukupuoli tai bruttopalkka. Määrällinen tutkimusmenetelmä pyrkii vastamaan sellaisiin kysymyksiin kuin ”kuinka paljon” tai ”miten usein”. (Vilka 2007.) Määrällinen aineisto voi olla esimerkiksi tilasto siitä, kuinka monta klikkausta minuutissa erilaiset verkkosivulla olevat mainoskuvat keräävät. Määrällisen tutkimuksen aineistoa voidaan kerätä esimerkiksi mittauksilla ja tietokantahauilla. (Taanila 2014.)

Kvalitatiivinen eli laadullinen tutkimusmenetelmä pyrkii puolestaan vastaamaan sellaisiin kysymyksiin, kuin ”miksi” ja ”miten”. Tutkimus kuvaa sen kohderyhmät ja ilmiöt kokonaisvaltaisesti ja kertoo niiden syy- ja seuraussuhteista. (Inspirans 2015.) Mittaamisen ja laskemisen sijaan laadullisessa tutkimuksessa pyritään ymmärtämään syitä tutkittavan ilmiön takana (Hirsjärvi & Huttunen 1995, 174, 201). Laadullisen tutkimuksen aineistoa voidaan kerätä mm. haastattelemalla ja havainnoimalla (Taanila 2014).

5.3 Validiteetti ja reliabiliteetti

Validiteetti eli pätevyys kertoo, miten hyvin jossakin tutkimuksessa käytetyt mittaus- ja tutkimusmenetelmät mittaavat tutkittavan ilmiön ominaisuutta. Hyvä validiteetti saavutetaan oikeilla kohderyhmillä ja kysymyksillä. Validiteetin puuttuminen tekee tutkimuksesta arvottoman. (Hiltunen 2009.)

Reliabiliteetti eli luotettavuus kertoo, miten luotettavasti tutkimuksessa käytetyt mittaus- ja tutkimusmenetelmät mittaavat tutkittavaa ilmiötä ja miten vähän tuloksissa on sattumanvaraisuutta. Reliabiliteetin osatekijät ovat stabiliteetti ja konsistenssi. Stabiliteetti kertoo mittarin

tai menetelmän pysyvyydestä ajassa. Sitä voidaan tutkia vertailemalla ajallisesti perättäisiä mittauksia. Tulosten konsistenssi eli yhtenäisyys kertoo tulosten riippumattomuudesta, kun useammasta väittämästä koostuva mittari jaetaan kahteen väittämäjoukkoon. (Hiltunen 2009.)

5.4 Lähdekritiikki

Lähdekritiikki tarkoittaa kriittistä suhtautumista esimerkiksi tutkimuksessa käytettäviin lähteisiin, ennen kuin niitä käytetään lähteinä. Lähdeaineistoa tutkiessa on syytä miettiä, kuinka luotettava aineiston tuottaja on, missä aineisto on alun perin julkaistu ja mitä lähteitä itse aineisto on käyttänyt. (Hjelt 2010.)

Lähdekritiikki on tärkeää erityisesti internetaineistoa käytettäessä, missä aineiston julkaiseminen on helppoa ja vailla minkäänlaista ennakkosensuuria. Internetainestoa on myös helppo kopioida ja uudelleenjulkaista muokattuna, eikä kävijä välttämättä erota muokattua aineistoa alkuperäisestä. Verkkolähteissä on hyvä käyttää ensisijaisesti sellaisia lähteitä, missä julkaisijan nimi ja muut tiedot ovat saatavilla. (Hjelt 2010.)

6 Kehitysympäristö ja käytetyt työkalut

Tässä projektissa ei käytetty erillistä kehityspalvelinta, vaan verkkosivu luotiin suoraan W3:n Jelastic-palvelimelle, minne asiakkaan hankkima jouluapua.fi verkkotunnus ohjattiin sivuston valmistuttua. Tässä oli etuna se, että valmista sivustoa tai sen tietokantoja ei tarvinnut lähteä siirtämään kehityspalvelimelta muualle, mikä erityisesti WordPressin kanssa voi olla välillä hankalaa.

HTML-, CSS- ja PHP-koodin kirjoittaminen hoidettiin ilmaisella Sublime Text 2 -tekstieditorilla oman henkilökohtaisten mieltymykseni takia. Projektissa siirrettiin paljon tiedostoja FTP-yhteyden kautta, mikä onnistui Kubuntu-nimisen Linux-jakelun oletustiedostoselaimella.

Sisällönhallintajärjestelmäksi verkkosivustolle valittiin WordPress sen helppokäyttöisyyden ja suuren käyttäjäyhteisön takia sekä ennen kaikkea siksi, että se on todettu toimivaksi ja tarpeet kattavaksi aikaisemmissa projekteissa.

7 Projektin kulku ja toteutus

Projekti toteutettiin noin kuukauden aikana. Projektin aikana asiakkaaseen oltiin yhteydessä sähköpostitse. Projekti alkoi verkkotapaamisella asiakkaan kanssa ja päättyi valmiin verkkosivuston julkaisuun.

7.1 Vaatimusmäärittely

Projekti alkoi asiakkaan kanssa käydylä Skype-tapaamisella, missä käytiin läpi Jouluavun nykyistä toimintaa, mitä ongelmia Facebookissa toimivassa ryhmässä on ja mitä ominaisuuksia uudella verkkosivustolla pitäisi olla.

Päätarkoitus verkkosivustolla on, että avun tarvitsijat ja antajat voivat ottaa yhteyttä sen kautta. Aikaisemmin yhteydenottoja on tullut niin paljon Facebookin kautta, että niiden hallinnointi Facebookin oman viestijärjestelmän kautta alkaa muuttua vaikeaksi. Toiveena oli, että verkkosivustolle saisi yhteydenottolomakkeen, minkä data ohjattaisiin Jouluapua varten perustettuun sähköpostiin.

Muita vaatimusmäärittelyssä esille tulleita asioita olivat:

- Selkeä yhteydenottotapa avun tarvitsijoille ja avun antajille.
- Yksityisten henkilöiden ja yritysten yhteydenottokanavat erikseen.
- Kiitossivu, minne apua saaneet voivat lähettää kiitoksiaan.

Seuraavia vaatimuksia ei tapaamisessa määritelty, mutta asetin ne omiksi henkilökohtaisiksi vaatimuksiksi projektin kannalta:

- Responsiivinen ulkoasu, eli ulkoasu skaalautuu näytölle sopivaksi riippumatta siitä, minkä kokoiselta näytöltä sivua selataan.
- Mahdollisimman helppokäyttöinen asiakkaan näkökulmasta.
- Mahdollisimman kevyesti latautuva, jotta käytettävyys mobiililaitteilla on hyvä.

Graafikko loi vaatimusmäärittelyn ja asiakkaan toiveiden pohjalta alustavan ulkoasun, mikä näkyy kuvassa 5. Asiakas hyväksyi ulkoasun ja sivustoa lähdettiin rakentamaan sen pohjalta. Ulkoasuun voi tulla pieniä graafisia muutoksia, mutta pääosin rakenne tulee olemaan ulkoasu-suunnitelman mukainen.



Kuva 5: Sivuston alustava ulkoasu

7.2 WordPressin asentaminen

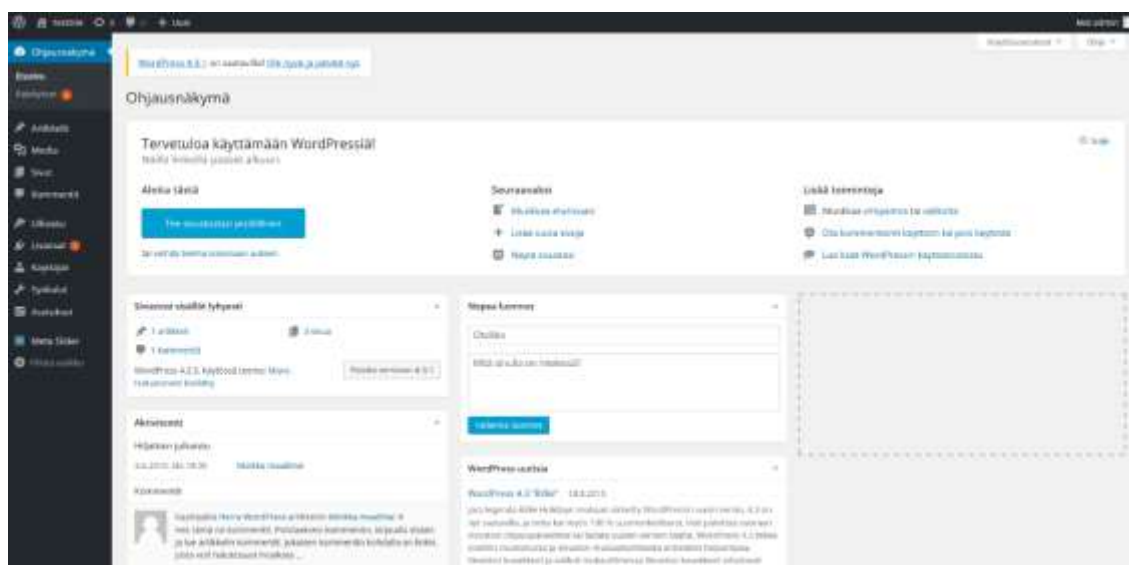
Sivuston kehitysprojekti aloitettiin lataamalla WordPressin uusin versio, 4.3 ja siirtämällä se kehityspalvelimelle. Samalla luotiin myös MySQL-tietokanta, mihin WordPress tallentaa kaiken sivuston datan, kuten sisällön ja asetukset. Tietokannalle luotiin myös oma MySQL-käyttäjä, millä on tarvittavat oikeudet WordPressin tietokantaan.

WordPress mainostaa asentamisen kestävän vain viisi minuuttia, mikä pitikin hyvin paikkansa. Asennusvaiheessa annettiin projektissa käytettävän tietokannan tiedot ja määriteltiin sivuston nimi sekä pääkäyttäjän tunnus ja salasana. Asennuksen yhteydessä oli myös ominaisuus estää sivuston näkyminen hakukoneissa, mikä otettiin käyttöön, ettei sivuston kehitysversio indeksoidu vielä hakukoneisiin.

7.3 Hallintapaneeli

WordPressin sisällön ja asetusten hallinta tapahtuu hallintapaneelin kautta, mihin kirjaudutaan ensimmäisen kerran asennusvaiheella luoduilla käyttäjätunnuksilla. Hallintapaneelissa käyttäjä voi muun muassa luoda uusia sivuja ja artikkeleita, hallita lisäosia ja teemoja, luoda uusia käyttäjiä sekä hallita WordPressin omia asetuksia. Kuvassa 6 on esitetty kuvankaappaus WordPressin hallintapaneelin etusivusta. WordPress ilmoittaa hallintapaneelissa selkeästi saatavilla olevista päivityksistä.

Tässä vaiheessa WordPressin asetuksiin tehtiin tarvittavat muutokset. Sivuston osoiterakenne muutettiin oletusasetuksesta selkeämpään artikkelimuotoon. Oletusasetuksilla sivuston osoitteet olivat muodossa <http://esimerkki.com/?p=24>, kun taas uusi asetus näyttää osoitteessa sivun tai artikkelin nimen, esimerkiksi <http://esimerkki.com/uutiset>. Jälkimmäinen osoiterakenne on huomattavasti oletusrakennetta selkeämpi. Asetuksista määritettiin myös sivuston staattinen etusivu.

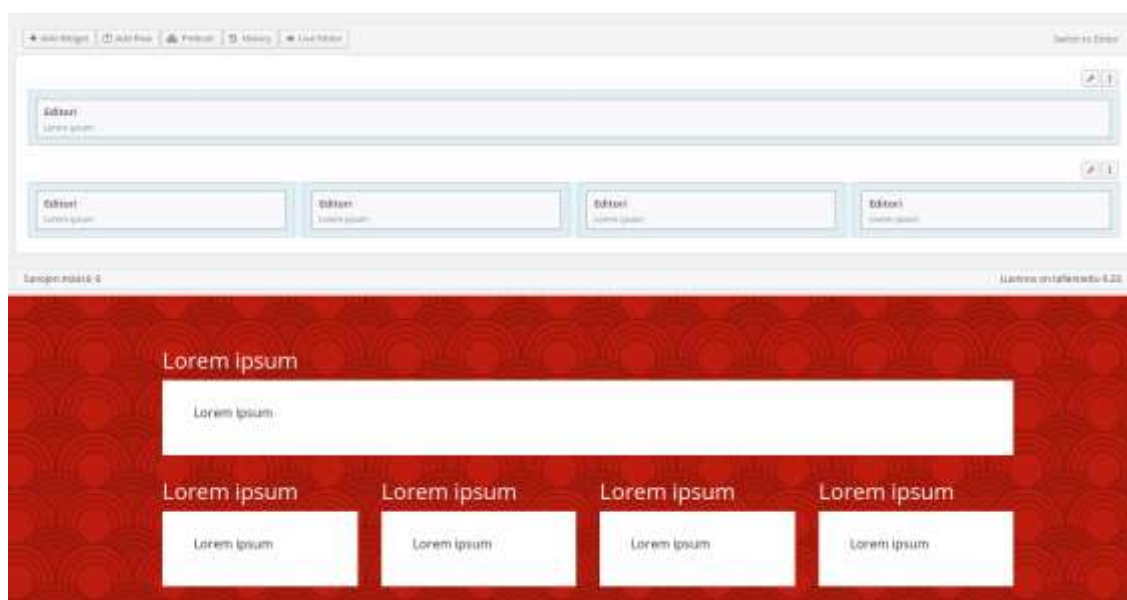


Kuva 6: WordPressin hallintapaneeli. Sivupalkin oranssit huomiopallot kertovat, että sivun lisäosiin tai teemoihin on saatavilla päivityksiä

7.4 Lisäosat

WordPressin asentamisen jälkeen sivustolle asennettiin muutamia lisäosia, mitkä koettiin tarpeelliseksi sisällyttää alusta alkaen. Lisäosia pystyy tarpeen tullen asentamaan lisää, mikäli kehityksessä ilmenee tarvetta jollekin tietylle ominaisuudelle. Sivustolle asennettiin seuraavat lisäosat:

- Page Builder by SiteOrigin - Mahdollistaa sivuston sisältöalueen jakamisen erikokoisiin ja responsiivisiin palstoihin helposti. Kuten kuva 7 havainnollistaa, sivu voidaan jakaa useisiin erikokoisiin palstoihin käyttäjän toiveiden mukaan.
- Contact Form 7 - Antaa helpon käyttöliittymän lomakkeiden luomista ja lähettämistä varten. Tämän avulla luodaan lomake, millä käyttäjät voiva pyytää apua. Lisäosan asetuksista voidaan määritellä sähköposti, mihin täytetyt lomakkeet lähetetään.
- Velvet Blues Update URLs - Lisäosa, millä saa päivitettyä kaikki sivuston osoitteet nopeasti. Tätä käytetään sivuston julkaisun jälkeen, kun jouluapua.fi verkkotunnus on otettu käyttöön.
- Yoast SEO - Optimoii sivuston hakukonenäkyvyyttä paremmaksi.



Kuva 7: Page Builderilla luotu sivurakenne WordPressin editorissa sekä luodulla sivulla.

7.5 Sivurakenne ja täytesisältö

Sivuston sivurakenne ja sisältö ei tässä vaiheessa ollut vielä täysin selvillä, mutta sivustolle luotiin muutama alasivu ja lisättiin hieman Jouluavun Facebook-sivulta otettua sisältötekstiä. Alustava sivurakenne, mikä ilmenee kuvasta 8, voi hyvinkin jäädä pysyväksi ja vaikka ei jäisi, kehitystyö on paljon helpompaa, kun sisältö ja alisivut vastaavat mahdollisimman paljon haluttua lopputulosta.



Kuva 8: Alustava sivurakenne

Alustava sivustorakenne oli varsin yksinkertainen; kaikki sivut olivat päätason sivuja karttasivua lukuun ottamatta, mikä on ”Haluan auttaa” -sivun alasisivu. Päätason sivut tulevat näkyviin sivuston ylälaudassa olevaan navigaation. Alatasen sivut tulevat näkyviin joko silloin, kun käyttäjä vie hiiren päätason linkin päälle tai vasta silloin, kun alatasen sisältävä pääsivu on avattu.

7.6 Teeman luominen

Teeman luominen on ehdottomasti projektin aikaa vievin osuus. Teemaa ei kuitenkaan lähdetty luomaan aivan tyhjästä, vaan projektissa käytettiin aikaisemmin rakennettua pohjateemaa. Pohjateema on luotu niin, että sen päälle on suhteellisen helppo lähteä rakentamaan omaa teemaa. Se ei sisällä varsinaista graafista ulkoasua, mutta tarjoaa valmiin teemarakenteen ja alkeellisen CSS-rakenteen, mitä on helppo lähteä kehittämään haluttuun suuntaan.

Tässä tapauksessa pohjateema tarjosi kelvollisesti mallinnetun HTML-pohjan, mikä sopii hyvin sivustoa varten. Päätehtäväksi jäi siis graafisen ulkoasun toteuttaminen CSS-tyylittelyn avulla.

Sivun responsiivisuus toteutettiin käyttämällä CSS:n media query -ominaisuutta, minkä avulla voidaan CSS-tyylejä voi määritellä niin, että tietyt tyylit näytetään vain tietyillä näyttöleveyksillä. Tyylä luodessa käytettiin ns. ”mobile first” -ratkaisua eli sivusto lataa ensin vain kaikkein kriittisimmät tyylit ja hakee sen jälkeen käyttäjän näytön koosta riippuen lisää tyylejä. Näin mobiilikäyttäjät eivät turhaan joudu lataamaan isommille näytöille tarkoitettuja tyylejä, mikä näkyy sivuston nopeampana latausaikana mobiililaitteella.

Kaikkea responsiivisuutta ei tarvinnut lähteä koodaamaan itse, vaan aikaisemmin esitelty Page Builder -lisäosan luomat elementit ovat automaattisesti responsiivisia. Kuten kuvasta 9 näkyy, rinnakkain olevat laatikot asemoituvat automaattisesti päällekkäin, kun näytön leveys

on tarpeeksi pieni. Kuvassa näkyy myös mobiilinavigaatio, millaiseksi navigaatiopalkki muuttuu pienellä näytöllä. Mobiilinavigaatio klikataan erikseen auki tarvittaessa, minkä ansiosta se ei vie turhaa tilaa pieneltä näytöltä.



Kuva 9: Verkkosivun etusivu 400px leveällä näytöllä

CSS-toteutuksen lisäksi sivustolle kirjoitettiin pieni JavaScript -koodinpätkä, millä luotiin ns. "sticky navigation" eli navigaatiopalkki, mikä pysyy aina sivun yläaidassa, vaikka sivua selaisi alaspäin. JavaScript tarkkailee, miten paljon käyttäjä on rullannut sivua alaspäin ja tietyn rullauspisteen saavutettua antaa navigaatioelementille CSS-attribuutin "position: fixed", jolloin navigaatiopalkki jää näkyviin sivun ylälaitaan, vaikka sivua rullaisi alemmaksi.

7.7 Sisällönsyöttö ja testaus

Asiakas lähetti sivustolle laitettavat tekstit sähköpostitse. Ne kopioitiin ja liitettiin sivustolle. Asiakkaalle luotiin myös omat tunnukset sisällönhallintajärjestelmään sekä annettiin nopea käyttöohjeistus, jotta asiakas voi tarvittaessa itse muokata tekstejä tai muuta sisältöä. Kun sisältö oli syötetty, verkkosivu oli valmis testattavaksi.

Valmis sivusto ei ollut kovin laaja. Erillisiä sivuja oli vain kuusi kappaletta, joten sivuston testaaminen onnistui varsin nopeasti. Testaamisessa kiinnitettiin erityisesti huomiota siihen, miten sivusto renderöityy eri selaimilla ja erikokoisilla ruuduilla. Tavoitteena oli pitää selainkohtaiset erot mahdollisimman pieninä ja säilyttää hyvä käytettävyys myös pienemmillä ruutukoilla. Testaus suoritettiin vuoden 2015 suosituimmilla selaimilla eli Google Chromella, Mozilla Firefoxilla ja Internet Explorerilla (W3Counter 2015). Mobiilitestaus suoritettiin Android- ja iOS-mobiilikäyttöjärjestelmillä.

Sivusto toimi hyvin työpöytäkoneella eri selaimilla, mutta mobiililaitteilla selatessa sivuston elementit asemoituivat toistensa päälle, jos puhelinta piti pystyasennossa. Virhe johtui sisältöelementeille annetuista staattisista korkeuksista ja korjaantui muuttamalla ne dynaamisiksi. Positiivisena yllätyksenä tuli se, että sivusto toimi hyvin myös Internet Explorerilla,

8 Sivuston käyttöönotto

Sivusto otettiin käyttöön lokakuun 2015 alussa, kun testausvaiheessa ilmenneet ongelmat saatiin korjattua. Koska sivusto oli luotu suoraan tuotantopalvelimelle, sivustoa ei tarvinnut lähteä siirtämään erikseen, vaan pelkkä verkkotunnuksen uudelleenohjaus uudelle sivustolle riitti.

Verkkotunnuksen uudelleenohjauksen jälkeen sivuston linkit päivitettiin vastaamaan uutta osoitetta. Tämä onnistui Velvet Blues Update URL -lisäosalla, mikä asennettiin valmiiksi jo projektin alkuvaiheilla. Linkkien päivittämisen lisäksi sivustoin hakukoneilta piilottava asetus, mikä otettiin käyttöön sivustoa luodessa, otettiin pois, jotta hakukoneet voivat indeksoida sivuston. Lisäksi sivustolle upotettiin Google Analytics -skripti, minkä avulla sivuston kävijädataa voidaan tilastoida ja tutkia.

Sivuston käyttöönotto tapahtui onnistuneesti. Julkaistu sivu täytti aikaisemmin määritellyt vaatimukset, oli visuaalisesti mieluisan näköinen ja toimi myös mobiililaitteilla. Vielä julkaisun jälkeen sivustolta korjattiin muutamia bugeja, mitä ei havaittu testausvaiheessa.

9 Oma arviointi

Projektin ensisijainen tavoite oli luoda asiakkaalle uudet verkkosivut. Pää tavoite onnistui laaditun aikataulun puitteissa, joten projektia voidaan pitää onnistuneena. Myös projektille asetetut henkilökohtaiset tavoitteet onnistuivat hyvin ja kaiken kaikkiaan pääsin käyttämään projektissa hyvin aiempaa osaamistani ja oppimaan uutta.

Merkittävimpänä projektista opittuna asiana jäi mobiilitestauksen tärkeys. Mobiilitoimivuutta pystyi testaamaan helposti pöytätietokoneella muuttamalla selaimen ikkunakokoa, mutta näin tehdessä paljastui, että työpöytäselain ei renderöi sivua täysin samalla tavalla kuin mobiiliselain, jolloin jotkin bugit voivat jäädä huomaamatta. Näin ollen täydellinen mobiilitestaus vaatii aina kunnollisen testilaitteen.

10 Yhteenveto

Opinnäytetyössä luotiin verkkosivusto Jouluapua -yhteisölle käyttäen apuna WordPress sisälönhallintajärjestelmää. Sen lisäksi työssä käytiin läpi kyseisen järjestelmän toimintaa. Jouluapua äänestettiin voittajaksi W3:n jouluna 2014 pitämässä kilpailussa, missä voittajalle tehdään uudet verkkosivustot hyväntekeväisyystyönä. Jouluavulla ei ollut aikaisempia verkkosivuja, vaan yhteisö oli toiminut Facebookissa.

Projekti alkoi verkkotapaamisella asiakkaan kanssa, missä kartoitettiin mitä kaikkea sivustolle tarvitaan sekä haettiin toiveita sivuston graafisesta ulkoasusta. Näiden pohjalta luotiin sivuston graafinen ulkoasu, mikä hyväksyminen jälkeen koodattiin WordPress-teemaksi ja otettiin käyttöön projektille luodussa WordPress-asennuksessa. Teeman käyttöönoton jälkeen sivustolle syötettiin asiakkaan antama sisältö ja sivuston toimivuus eri laitteilla ja selaimilla testattiin. Valmis verkkosivusto, minkä lopullinen ulkoasu käy ilmi kuvasta 10, julkaistiin lokakuun 2015 alussa ja löytyy osoitteesta <http://jouluapua.fi>

JOULUAPUA
Joulu on antamisen aikaa

JOULUAPUA | PALJUN KUTTEN | TAVITSEN APJA | KARTTI | KUTTOVUOKSE | TIEDOJOULUUUNSTA

JOULU ON ANTAMISEN AIKAA

Jouluaika on yhtiössä, jonka tavoitteena on toteuttaa lapsille ja nuorille joulujuhlat tarjoamalla huolellista palvelua lämpimällä ja turvallaisella tilalla. Ohjelmassa on mukana vierailevien perheiden ja avustajien väliä.

Tavoitteena on saada julkistettujen hakemusten yhdistelmä vuosina 2013 ja 2014 noin 350 perheelle vapaaehtoisten avulla kassaa.

Kassan myynti ja lahjoittaminen on paljolti Jouluaika-yhtiöiden perustajien, Heidi Mattilan työkavereiden 2013 varamurhekkajoukko.

SUOSITUMMAT LAHJATOIVEET

I. Joulunäkki
II. Pyjamatossit
III. Wii-ohjelma

KATSO LAHJATOIVEET

MITEN VOIN TOIVOA?

Voit lähettää jouluaika-hakemusten OYTYÖSSÄ sivustalle alla olevan lähtölaatikon.

LÄHETÄ LAHJATOIVEET

YHTEISTYÖSSÄ MUKANA

MAPLET®
WE GROUP

Kuva 10: Kuvankaappaus julkaistusta verkkosivusta

Lähteet

Kirjallisuus

Hedengren, T. 2010. Smashing WordPress - Beyond the blog. West Sussex : John Wiley & Sons.

Heinisuo, R. & Rauta, I. 2007. PHP ja MySQL - Tietokantapohjaiset verkkopalvelut. Helsinki : Talentum.

Hirsjärvi, S. & Huttunen, J. 1995. Johdatus kasvatustieteeseen. Helsinki, Porvoo : WSOY.

Sähköiset lähteet

Airaksinen, T. 2009. Toiminnallisen opinnäytetyön kirjoittaminen. Viitattu 13.10.2015.
<http://www.slideshare.net/TiinaMarjatta/toiminnallinen-opinnytety-tekstin>

Bangwal, K. 2012. PPT on Prototyping Model. Viitattu 29.11.2015.
<http://www.slideshare.net/shuisharma/prototype-model>

Bosomworth, D. 2015. Mobile Marketing Statistics 2015. Viitattu 25.10.2015.
<http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>

Boudreaux, R. 2012. How to create media queries in Responsive Web Design. Viitattu 25.10.2015
<http://www.techrepublic.com/blog/web-designer/how-to-create-media-queries-in-responsive-web-design/>

Fylan, J. 2015. Understanding WordPress Posts and Pages and Knowing When to Use Them For the Best Results. Viitattu 29.11.2015.
<https://theme-fusion.com/wordpress-posts-and-pages-explained/>

Habib, M. 2013. Agile software development methodologies and how to apply them. Viitattu 27.11.2015.
<http://www.codeproject.com/Articles/604417/Agile-software-development-methodologies-and-how-t>

Hiltunen, L. 2009. Validiteetti ja reliabiliteetti. Viitattu 29.9.2015.
http://www.mit.jyu.fi/ope/kurssit/Graduryhma/PDFt/validius_ja_reliabiliteetti.pdf

Hjelt, T. 2010. Lähdekritiikki. Viitattu 30.3.2015.
http://www.pori.tut.fi/kirjasto/web/tty_info/tty_lahdekritiikki.pdf

Hughey, D. 2009. The Traditional Waterfall Approach. Viitattu 27.11.2015.
<http://www.umsl.edu/~hugheyd/is6840/waterfall.html>

Inspirans. 2015. Kvalitatiivinen tutkimus. Viitattu 8.2.2015.
<http://www.inspirans.fi/kvalitatiivinen-tutkimus>

Kline, J. 2012. Web Content Management 101. Viitattu 10.8.2015.
<http://www.accrinet.com/blog/web-content-management-101/>

Kline, J. 2013. 5 Reasons You Should Be Using A Content Management System. Viitattu 10.8.2015.
<http://www.accrinet.com/blog/5-reasons-you-should-be-using-a-content-management-system/>

- Korpela, J. 2009. Rakennetta mukaan (HTML). Viitattu 25.10.2015.
<https://www.cs.tut.fi/~jkorpela/webjulk/1.2.html>
- Marcotte, E. 2010. Responsive Web Design. Viitattu 25.10.2015.
<http://alistapart.com/article/responsive-web-design>
- Mozilla Foundation. 2015. HTML5. Viitattu 25.10.2015.
<https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>
- Rocheleau, J. 2012. A look into: Designing for Mobile Devices. Viitattu 23.20.2015.
<http://www.hongkiat.com/blog/designing-for-mobile-devices/>
- Rouse, M. 2011. Content management system (CMS) definition. Viitattu 10.8.2015.
<http://searchsoa.techtarget.com/definition/content-management-system>
- Ruohonen, S. 2015. Toimintatutkimus opinnäytetyössä. Viitattu 14.10.2015.
http://www2.amk.fi/digma.fi/www.amk.fi/material/attachments/van-haamk/etuotanto/030906/5gxjQD2CJ/toimintatutkimus_opinnaytetyossa.ppt
- Saarikumpu, O. 2006. JavaScript-kielen alkeet - osa 1. Viitattu 25.20.2015.
<http://weppipakki.com/js/opas/alkeet1.htm>
- Taani, A. 2014. Määrällisen aineiston kerääminen. Viitattu 2.9.2015.
<http://my.haaga-helia.fi/-taaak/t/suunnittelu.pdf>
- Tappura, S. 2009. Toimintatutkimus. Viitattu 27.9.2015.
http://matriisi.ee.tut.fi/hmopetus/hmjatko-opintosemma/2008/Tappura_Toimintatutkimus090109.pdf
- TechTerms. 2009. Web Development Definition. Viitattu 25.10.2015.
http://techterms.com/definition/web_development
- Vilka, H. 2007. Tutki ja mittaa. Viitattu 2.9.2015.
<http://hanna.vilka.fi/wp-content/uploads/2014/02/Tutki-ja-mittaa.pdf>
- Waters, K. 2007. Disadvantages of Agile Development. Viitattu 29.11.2015.
<http://www.allaboutagile.com/disadvantages-of-agile-development/>
- WordPress 2015^a. History. Viitattu 3.9.2015.
<https://codex.wordpress.org/History>
- WordPress, 2015^b. Plugin Directory. Viitattu 2.9.2015.
<https://wordpress.org/plugins/browse/popular/>
- WordPress Codex, 2015^a. Plugins. Viitattu 2.9.2015.
<http://codex.wordpress.org/Plugins>
- WordPress Codex, 2015^b. WordPress Widgets. Viitattu 29.11.2015.
https://codex.wordpress.org/WordPress_Widgets
- WordPress Codex, 2015^c. Post Types. Viitattu 29.11.2015.
https://codex.wordpress.org/Post_Types
- W3C, 2015. Cascading Style Sheets home page. Viitattu 24.10.2015.
<http://www.w3.org/Style/CSS/>
- W3Counter, 2015. Global Web Stats. Viitattu 14.10.2015
<http://www.w3counter.com/globalstats.php>

Kuvat

Kuva 1: Jouluapua-projektin vesiputousmalli.....	12
Kuva 2: Esimerkki prototyypimallista	13
Kuva 3: Yksinkertainen pohjatiedosto.....	16
Kuva 4: Kuva vimpainalueesta, mihin on lisätty hakukentän ja kolmen viimeisimmän artikkelin näyttävät vimpaimet	18
Kuva 5: Sivuston alustava ulkoasu	22
Kuva 6: WordPressin hallintapaneeli. Sivupalkin oranssit huomiopallot kertovat, että sivun lisäosiin tai teemoihin on saatavilla päivityksiä	23
Kuva 7: Page Builderilla luotu sivurakenne WordPressin editorissa sekä luodulla sivulla. ..	24
Kuva 8: Alustava sivurakenne	25
Kuva 9: Verkkosivun etusivu 400px leveällä näytöllä.....	26
Kuva 10: Kuvankaappaus julkaistusta verkkosivusta.....	29