



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Markus Makkonen

Web-kehitys Drupal-sisällönhallintajärjestelmällä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Software engineering

Insinöörityö

11.10.2019

Tekijä Otsikko	Markus Makkonen Web-kehitys Drupal-sisällönhallintajärjestelmällä
Sivumäärä Aika	25 sivua 11.10.2019
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tietotekniikan koulutusohjelma
Ammatillinen pääaine	Software engineering
Ohjaajat	Osaamisaluepäällikkö Janne Salonen Lehtori Jussi Alhorinne
<p>Tämä opinnäytetyö käsittelee avoimeen lähdekoodiin perustuvaa Drupal-sisällönhallintajärjestelmää ja sen hyödyntämistä verkkokehityksessä. Aihetta kuvataan käsittelemällä sen määrittelyä, historiaa ja teknisiä ominaisuuksia.</p> <p>Opinnäytetyön projektiosuus toteutettiin Karhu Helsinki Oy:n kanssa yhteistyössä ja työn tarkoituksena oli toteuttaa vuosikatsaussivu vakuutusalan asiakkaalle Drupal 7 -sisällönhallintajärjestelmää hyödyntäen. Vuosikatsaussivun pääasiallinen tehtävä on kuvastaa menneen vuoden talouslukuja sekä kuulumisia.</p> <p>Lopputuloksena projektista oli toimiva, teknisesti ja visuaalisesti koherentti kokonaisuus, johon asiakas oli erittäin tyytyväinen. Sivusto vietiin onnistuneesti palvelimelle ja sen jatkokehitysmahdollisuudet ovat avoinna.</p>	
Avainsanat	Drupal, web-kehitys, sisällönhallintajärjestelmä

Author Title	Markus Makkonen Web development with Drupal content management framework
Number of Pages Date	25 pages 11 October 2019
Degree	Bachelor of Engineering
Degree Programme	Information technology
Professional Major	Software engineering
Instructors	Janne Salonen, Head of Department (ICT) at Metropolia Jussi Alhorinne, Senior Lecturer
<p>The purpose of this thesis is to introduce Drupal content management system in its entirety. Introduction is divided into general information, history and technical specifications which all together leaves the reader with a good understanding of how Drupal works.</p> <p>The project part of this bachelor's thesis is to create a working website based on the specifications set by a client of Karhu Helsinki Oy. The client is an insurance company which requires a website to display annual review of different economic numbers and news from the previous year. The website uses Drupal 7 as a platform.</p> <p>The results of this project were pleasing for both the client and Karhu Helsinki Oy. Everything that was set to be done within the time constraints of the project was finished. The website is running on a previously set-up web server and is ready to be further developed if needed.</p>	
Keywords	Drupal, web development, content management system

Sisällys

1	Johdanto	1
2	Sisällönhallintajärjestelmät	2
2.1	Sisällönhallintajärjestelmän perusominaisuudet	2
2.2	Sisällönhallintajärjestelmän hyödyt	2
3	Drupal	4
3.1	Miksi Drupal?	4
3.2	Drupalin historia	5
3.3	Drupalin asennus	6
3.4	Drupal core	8
3.5	Teemat	9
3.6	Sisältötyypit	9
3.6.1	Sisältötyyppien luominen	10
3.6.1	Sisältötyyppien julkaisuasetukset	10
3.7	Moduulit	11
3.8	Lohkot	12
3.9	Taksonomia	14
3.10	Näkymät	14
3.11	Metatiedot	16
3.12	Teemaaminen	17
4	Drupal-sivuston suunnittelu ja toteutus	18
4.1	Rakenteen suunnittelu	18
4.2	Tekninen toteutus	19
4.2.1	Sisältötyypit	19
4.2.2	Näkymät	21
4.2.3	Käyttäjäroolit	22
5	Yhteenveto	23
	Lähteet	24

Lyhenteet ja termit

Composer Drupalin kanssa käytettävä pakkaushallintajärjestelmä. Usein käytetty Drupal 8:n kanssa.

CMS Content Management System. Sisällönhallintajärjestelmä, jolla rakennetaan verkkosivuja.

CSS Cascading Style Sheets. Web-ohjelmoinnissa käytetty kieli, jolla HTML-elementeille voidaan lisätä tyyliohjeita.

Drush Drupalin kanssa käytettävä pakkaushallintajärjestelmä. Usein käytetty Drupal 7:n kanssa.

HTML Hypertext Markup Language. Web-ohjelmoinnissa käytetty kieli, jolla kuvataan sivustojen sisältöä.

JavaScript Web-ohjelmoinnissa käytetty kieli, jolla voidaan lisätä sivuille dynaamista toiminnallisuutta.

Moduuli Drupalissa käytettävä lisäosa, joka lisää tai muokkaa järjestelmän toiminnallisuutta.

Node Solmu. Yleensä yksittäinen sivuston sivu, joka koostuu siihen määritellystä sisällöstä.

PHP PHP: Hypertext Preprocessor. Web-ohjelmoinnissa käytetty kieli, jolla kuvataan web-sivujen sisältöä.

WYSIWYG What-you-see-is-what-you-get. Drupalin tekstieditori.

1 Johdanto

Verkkopalveluita tarjoavilta yrityksiltä vaaditaan teknisillä ominaisuuksillaan vakuuttavia sekä eri standardeilla kilpailukykyisiä ratkaisuja. Erilaiset sisällönhallintajärjestelmät ovat erinomainen ratkaisu hyvään verkkokehittämiseen sekä verkkosivujen kehittäjän että asiakkaan näkökulmasta. Yksi näistä sisällönhallintajärjestelmistä on Drupal, jolla voidaan rakentaa ja ylläpitää verkkosivuja sekä kuten sisällönhallintajärjestelmästä nimen perusteella voidaan päätellä, hallita ja valvoa siellä näytettävää sisältöä sekä asetuksia.

Tämän opinnäytetyön tavoitteena on esitellä Drupal-järjestelmän hyödyntämistä yritysmaailmassa, jossa sekä kehittäjä että asiakas saavat verkkosivuston käyttämisestä mahdollisimman paljon irti. Kehittäjälle tämä tarkoittaa tehokkuutta ennen kaikkea käyttöönotossa, rakentamisessa ja ylläpidossa. Asiakas puolestaan hyötyy järjestelmästä sen helpon sisällönluonnin ja valvonnan takia.

Opinnäytetyössä tutkitaan Karhu Helsinki Oy:n toimintatapoja ja -tekniikoita Drupalin kanssa työskennellessä. Näitä hyödyntäen yrityksen vakuutusalan asiakkaalle rakennettiin vuosikatsaussivu, jonka alustana toimii Drupal 7. Asiakkaan motiivi projektin valmistukselle oli saada oma sivustokokonaisuus, jossa esitellään yrityksen edellisen vuoden talouslukuja ja kuulumisia.

2 Sisällönhallintajärjestelmät

Sisällönhallintajärjestelmällä (engl. Content Management System tai CMS) tarkoitetaan verkkosivujen kanssa työskentelyyn tarkoitettua ohjelmistopakettia, joka tarjoaa mahdollisuuden sisällön luomiseen, julkaisemiseen ja ylläpitoon.

2.1 Sisällönhallintajärjestelmän perusominaisuudet

Sisältöä verkkosivuilla ovat kaikki tekstiä, grafiikkaa, kuvia, ääntä tai videota yhdessä tai yksin ilmentämät näkymät. Tyypillisesti sisällönhallintajärjestelmät tarjoavat myös tietokantayhteyden sisällön tallentamiselle sekä erilaisia työkaluja sivustolla käytettävien elementtien, teemojen, turvallisuuden ja käyttäjäryhmien muokkaamiseen.

2.2 Sisällönhallintajärjestelmän hyödyt

CMS-paketit on suunniteltu tarjoamaan sopivan rungon ja työkalut sivuston kanssa työskentelyyn, mutta niiden käyttäminen pienissä, muutaman sivun kattavassa sivustossa ei ole tarkoituksenmukaista. Sivuston mittakaavassa raskaan hallintajärjestelmän ylläpitäminen ei ole varsinkaan järkevää, jos sivuston sisältöä tai asetuksia ei tarvitse jatkuvasti muokata ja päivittää. [2, s. 1-2.]

Sisällönhallintajärjestelmiä käytetään web-kehityksessä pääasiallisesti, koska se tekee kaikesta työskentelystä yksinkertaisempaa suuremmilla sivustoilla. Ensimmäinen motiivi järjestelmien kehittämiseen oli juurikin verkkokehityksen helpottaminen tarjoamalla työkalun teknisesti osaamattomille käyttäjille. Tästä syystä verkkopalveluja tarjoavien yritysten on huomattavasti helpompi toteuttaa asiakkaalle sivusto, jota asiakas pystyy itse suurissa määrin muokkaamaan sisällön osalta. Sisällönmuokkauksen helppoudesta hyvänä esimerkkinä on Drupalin WYSIWYG-editori (what-you-see-is-what-you-get), joka vastaa hyvin paljon Microsoftin Wordia (ks. kuva WYSIWYG).

Ennen sisällönhallintajärjestelmien olemassaoloa verkkosivuja kehitettiin käyttämällä vain staattisia Hypertext Markup Language (lyh. HTML) -tiedostoja, jotka lisättiin web-palvelimelle. Tähän kömpelöön tapaan haluttiin muutos, jonka takia sisällönhallintajärjestelmät, kuten Drupal, kehitettiin. [3, s. 5-6.]

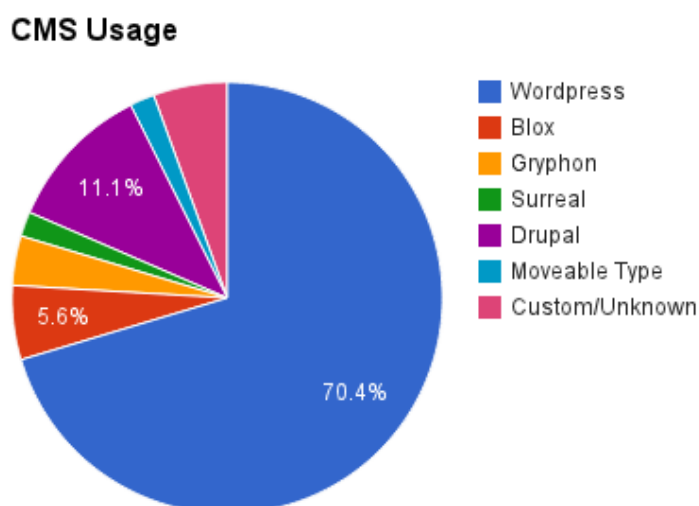
"OK, we can do this the hard way, or we can do this the Drupal way" -Forest Mars
[1, s. 3.]

3 Drupal

3.1 Miksi Drupal?

Drupalin käyttö sisällönhallintajärjestelmänä on hyvin usein toimiva valinta, mutta sen avulla toteutettavan projektin tulee soveltua järjestelmään: kaikki sivustot eivät ole rakenteeltaan samanlaisia.

Drupal on käytössä maailmanlaajuisesti ja on avoimen lähdekoodin järjestelmä. Tämä tarkoittaa sitä, että Drupal-yhteisöllä on vapaa oikeus muokata ja julkaista omia moduuleita omaan ja muiden käyttöön. Drupalin suosion takia moduuleita muokataan jatkuvasti käyttäjäystävällisemmiksi, toimivammiksi ja tietoturvalisemmiksi. Drupal on maailman kolmanneksi suosituin sisällönhallintajärjestelmä (kuva 1) ja jatkaa kasvuaan digitalisoituvassa maailmassa. [2, s. 2.]



Kuva 1. Ympyrädiagrammi käytetyimmistä sisällönhallintajärjestelmistä. [6.]

Drupalin toimintatapa, joskin paikoin monimutkainen, on paljon kattavampi kuin kilpailevilla järjestelmillä. Kehittäjiltä kehittäjille -tyyppinen Drupalin eteenpäinvienti tekee siitä hyvin ajan tasalla olevan kokonaisuuden, koska kehitys on jatkuvaa. Erityisesti monimutkaisten sivustojen rakentaminen taitavalle Drupalin käyttäjälle on helppoa, kun muilla järjestelmillä se saattaa olla jopa mahdotonta. Drupalin hyödyntäminen kehityksessä on myös ilmaista. [6.]

3.2 Drupalin historia

Drupalin kehittivät hollantilaiset opiskelijat Dries Buytaert ja Hans Snijder. Alun perin pieneksi tarkoitettu projekti kehitettiin auttamaan Buytaertia, Snijderiä ja heidän ystäviään jakamaan uutisia ja tapahtumia toisilleen uudenlaisen kanavan kautta. Antwerpen yliopistossa ei siihen aikaan ollut kaikille tarjottuja pysyviä internetyhteyksiä, joten opiskelijat käyttivät ohjelmistoa ensin sillatulla ADSL-modeemiyhteydellä.

Opiskelijoiden kehittämä ohjelmisto julkaistiin ensin nimellä ”drop.org”, mutta laajemman kiinnostuksen herättyä sitä kohtaan nimi muuttui ennen pitkää Drupaliksi. Nimi on johdettu hollanninkielisestä sanasta ”druppel”, jonka pohjalta myös ohjelmistolle laadittiin pisanamuotoinen logo.



Kuva 2. Drupalin pisanamuotoinen logo.

Drupal julkaistiin yleiseen käyttöön vuonna 2001. Julkaisu otettiin myöhemmin verkkokehityksen yhteisössä hyvin vastaan ja pian yksityishenkilöt, tiimit ja yhteisöt alkoivat rakentaa järjestelmän päälle jatkuvasti uusia ominaisuuksia, kuten keskustelufoorumeita

ja äänestyksiä. Pian Drupalista tulikin yksi tehokkaimmista ja ominaisuuksiltaan monipuolisimmista sisällönhallintajärjestelmistä. [2, s. 2; 11.]

3.3 Drupalin asennus

Drupalin käyttämiseen tarvitaan verkkopalvelin ja tietokantayhteys ja sitä suositellaan käytettävän niin sanotun AMP-pinon päällä, joka kattaa seuraavat palvelut:

- A – Apace, verkkopalvelin
- M – MySQL, tietokanta
- P – PHP, kieli, jota Drupal käyttää.

Drupalin asentaminen alkaa lataamalla asennustiedostot osoitteesta drupal.org/download ja purkamalla ne palvelimelle. Asennusohjelma tarkistaa, että vaaditut palvelut ovat asennettuna, jonka jälkeen lisätään tietokantayhteys. Tämän jälkeen lisätään enää sivuston tiedot, kuten sivuston nimi, sähköpostiosoite, käyttöliittymän kirjautumistiedot sekä aikavyöhyke. [9.]

Taulukko 1. Drupalin asennuksen minimivaatimukset ja -versiot. [7,8.]

	Drupal 7			Drupal 8		
Levytila	15Mb, 50Mb jos asennettuna on teema ja moduuleja			100Mb		
PHP	7.1			7.0.7		
Tietokanta	MySQL 5.0.15	Postgre 8.3	SQLite 3.3.7	MySQL 5.5.3	Post-Gre 9.1.2	SQLite 3.6.8

3.4 Drupal core

Drupalin perusasennuksessa asennetaan Drupalin niin kutsuttu core. Siihen sisältyvät kaikki Drupalin perustarpeet, joihin kuuluvat pääasiallisesti seuraavat asiat:

- sisällön luominen, muokkaaminen ja poistaminen
- tiedostojen lataus ja vienti palvelimelle
- valikot
- käyttäjät
- käyttäjäroolit- ja luvat
- taksonomiatermiit
- blogit
- keskusteluforumiit
- online-kyselyt.

Sivuston käyttäminen vain Drupal coren asennuksen jälkeen on mahdollista, mutta usein sivustolle asennetaan erinäisiä moduuleita, joiden avulla monimutkaisetkin tietoraken-
teet ja toiminnallisuudet saadaan toimimaan (ks. lukua 3.2.4). [2, s. 2.]

3.5 Teemat

Kun Drupal on asennettu ja toimii palvelimella normaalisti, voidaan Drupalille valita teema. Drupalin perusasennuksessa teemoja on valittavana neljä: Baritk, Garland, Seven ja Stark. Monia muita on ladattavissa valmiina paketteina, mutta usein näitä teemoja ei käytetä yksin, vaan ne räätälöidään tarkoituksenmukaisiksi. Valittu teema määrittää muun muassa seuraavat sivun ulkonäköön vaikuttavat asiat:

- kuinka moneen sarakkeeseen sivun näkymä jaetaan
- sivuston ala- ja ylätunnisteen näkyvyys
- sivuston päävalikon näkyvyys
- sivuston päävärit ja fontit.

Teema määrittää myös ylläpito näkymän tyylin eli sen, mitä sisään kirjautunut ylläpitäjä näkee asetusvalikoissa. [2, s. 5.]

3.6 Sisältötyypit

Sisältötyyppejä käytetään hyväksi, kun sisältöä lisätään normaalisti sivustolle. Sisältötyyppi toimii pohjana tietyntyyppiselle sisältöpaketille ja sen sisältämät kentät luovat julkaistuihin sivuihin sisältöelementit. Esimerkkinä sisältötyypeistä ovat esimerkiksi artikkeli, uutinen ja etusivu.

Drupalin perusasennukseen sisältyy muutamia valmiita sisältötyyppejä, mutta sivustojen suunnittelussa päädytään lähes aina tekemään mukautettuja sisältötyyppejä tarpeiden mukaan. Jos kaikki Drupal coren mukana tulevat moduulit otetaan käyttöön, valmiita sisältötyyppejä löytyy yhteensä kuusi. [2, s. 119-120; 3, s. 160.]

3.6.1 Sisältötyyppien luominen

Drupalin käyttöliittymästä voidaan lisätä omia, mukautettuja sisältötyyppejä sivujen pohjiksi. Lisääminen tapahtuu käyttöliittymän asetuksista Add content type -painikkeella, jonka jälkeen sisältötyypille annetaan nimi ja kuvaus. Sisältötyyppiin voi sitten lisätä haluamansa määrän kenttiä, jotka sisältävät sivustolla näytettävän datan. Erilaisia kenttiä voivat olla esimerkiksi pääkuva, otsikko, body-teksti ja päivämäärä. [2, s. 120-122.]

3.6.2 Sisältötyyppien julkaisuasetukset

Sisältötyypin määrittämisen jälkeen ja sitä pohjana käyttäen voidaan sisältöä alkaa luomaan. Julkaisuun liittyviä asetuksia sisältötyypissä on usein monia:

- Sisältö voidaan lisätä sivustolle ja tietokantaan julkaisemattomana, jolloin dataa ei näytetä, mutta se on olemassa.
- Sisältö voidaan lisätä näkymään etusivulle.
- Sisältö voidaan lisätä kaikkien listausten ensimmäiseksi. Esimerkiksi uutinen-sisältötyypillä tehdyt sisällöt voidaan näin nostaa uutislistauksen kärkeen.
- Sisällöstä voidaan luoda uusi versio. Tämä on erittäin tärkeä julkaisuun liittyvä asetus, jos jo aikaisemmin julkaistua sisältöä muokataan uudestaan ja vanha sisältö halutaan jättää sivustolle.
- Sisältö voidaan tehdä monikieliseksi. Tämä asetus otetaan usein käyttöön monikielisillä sivustoilla, jotta navigointi ja sisällön organisointi helpottuisi.
- Sisältöön voidaan lisätä sen kirjoittajan tiedot sekä julkaisu-aika. Tähän asetukseen voi lisätä myös halutessaan tekstile maksimipituuden.

- Sisältö voidaan lisätä valikkoon. Sivustoilla on usein monia eri valikoita (esim. päävalikko, alatunnisteen valikko, sivuvalikko), joihin sisältö halutaan näkymään.

Sisällön luomien ei ole siis niin yksinkertaista kuin voisi luulla. Sisältötyypin kenttien nimeäminen ja kuvauksen lisääminen ovat tärkeitä osia sisältötyyppejä luodessa, jotta kaikille sisältöä lisääville on selkeää, miten se toimii. [2, s.124-127.]

3.7 Moduulit

Moduulit tuovat coren päälle asennettuna toiminnallisuuden Drupaliin. Niitä voidaan pitää ikään kuin palapelin paloina, joista valmiina rakentuu valmis ja toimiva kokonaisuus. Coren asennuksessa on mukana 44 moduulia, jotka kaikki eivät ole automaattisesti käytössä asennuksen jälkeen. [3, s. 105.]

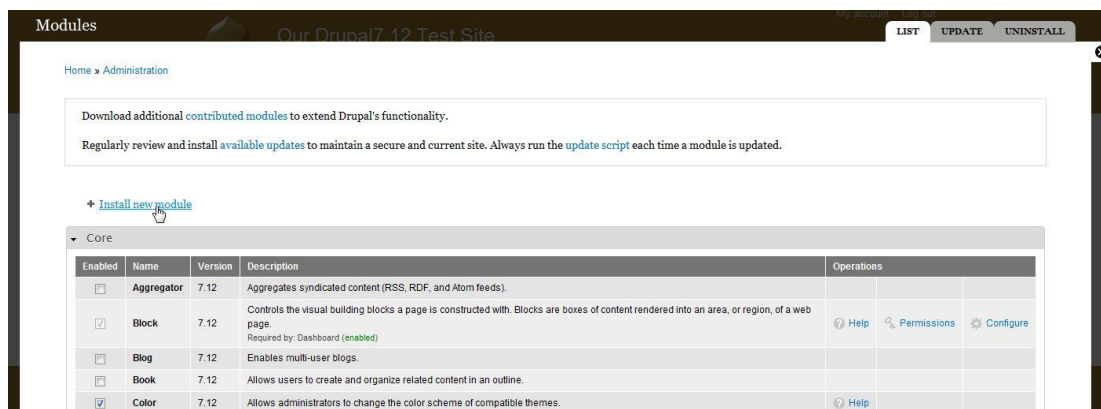
Moduulit asennetaan yleisesti komentoriviä käyttäen. Suositeltua on käyttää joko drushia tai composeria moduulin lataukseen. Composerilla asennus toimii seuraavasti:

```
composer require drupal/[moduulin_nimi]
```

Drush ei ole enää suositeltu Drupal 8 asennuksissa käytettävä pakkauksienhallintatyökalu, vaan sitä käytetään lähinnä Drupal 7 -sivustoilla. Drushilla moduulin asennus on seuraavanlainen:

```
drush dl [moduulin_nimi]
```

Moduulien lataamisen ja käyttöönoton jälkeen niillä on usein omat asetuksensa Drupalin ylläpito näkymässä (ks. kuva 2). [10.]



Kuva 3. Ylläpitäjän moduulilistausnäkymä Drupalissa.

3.8 Lohkot

Drupalin lohkot (engl. block) ovat joko käyttöliittymästä suoraan lisättyjä tai koodissa määritettyjä rakennuspaloja, joilla on oma sisältö. Ne ovat usein pieniä kokonaisuuksia ja sisältävät vähän dataa. Esimerkiksi valikot, sisäänkirjautumisenäkymä ja hakupalkki toteutetaan yleisesti lohkona. Sivustoilla on usein monia lohkoja jaoteltuina sivun eri alueisiin (engl. region), ja lohkon alueen voi valita raahaamalla ja pudottamalla. Muokkausnäköisestä voi myös poistaa lohkon käytöstä (ks. kuva 4). [2, s. 71.]

Lohko	Alue	Custom Theme	Toimenpiteet	
Language select				
+	Kielivalikko (static links)	Language select	Custom classes (advanced)	Muokkaa poista
Ylätunniste				
+	View: Sections: (block) First item	Ylätunniste		Muokkaa
Navigation bar				
+	Menu block (main-menu)	Navigation bar	Custom classes (advanced)	Muokkaa poista
+	Tiedostolinkit	Navigation bar	Custom classes (advanced)	Muokkaa poista
Korostettu				
Tällä alueella ei ole lohkoja				
Ohje				
+	Järjestelmäohje	Ohje		Muokkaa
Sisältö				
+	Pääsivun sisältö	Sisältö		Muokkaa
+	Takaisin ylös -linkki	Sisältö		Muokkaa poista

Kuva 4. Lohkojen ylläpito näkymä.

Usein lohkon sisällöt ovat hyvin spesifejä, pieniä kokonaisuuksia ja niiden näkyvyyttä tulee rajoittaa, jotta lohko näytetään vain sille suunnitellulla paikalla. Drupalin käyttöliittymässä on paljon erilaisia kriteerejä, joita voidaan lisätä lohkon näkyvyydelle:

- sivut
- sisältötyypit
- roolit
- käyttäjät.

Sivu-kriteerissä voidaan luoda lohkolle sääntö, joka rajoittaa sen näkymään vain tietyillä, yksittäisillä sivuilla. Sääntö voi olla myös käänteinen, jolloin listataan sivut, joilla lohkon ei tule näkyä.

Sisältötyyppi-kriteerissä voidaan lisätä lohko näkyviin vain tietyillä sisältötyypeillä tehtyihin sivuihin. Samoin kuin sivu-kriteerissä, tämäkin sääntö voi olla käänteinen.

Roolit-kriteeri antaa mahdollisuuden lisätä lohkoon näkyviin vain tietyille käyttäjäryhmille. Esimerkiksi monet vain järjestelmävalvojille näkyvät lohkot vaativat kirjautumisen sivustolle käyttäjällä, jolla on järjestelmävalvojan oikeudet.

Käyttäjät-kriteeri on hyvin paljon kuin roolit-kriteeri, mutta pätee yksittäisiin sivustolla oleviin käyttäjätileihin. [2, s. 76.]

3.9 Taksonomia

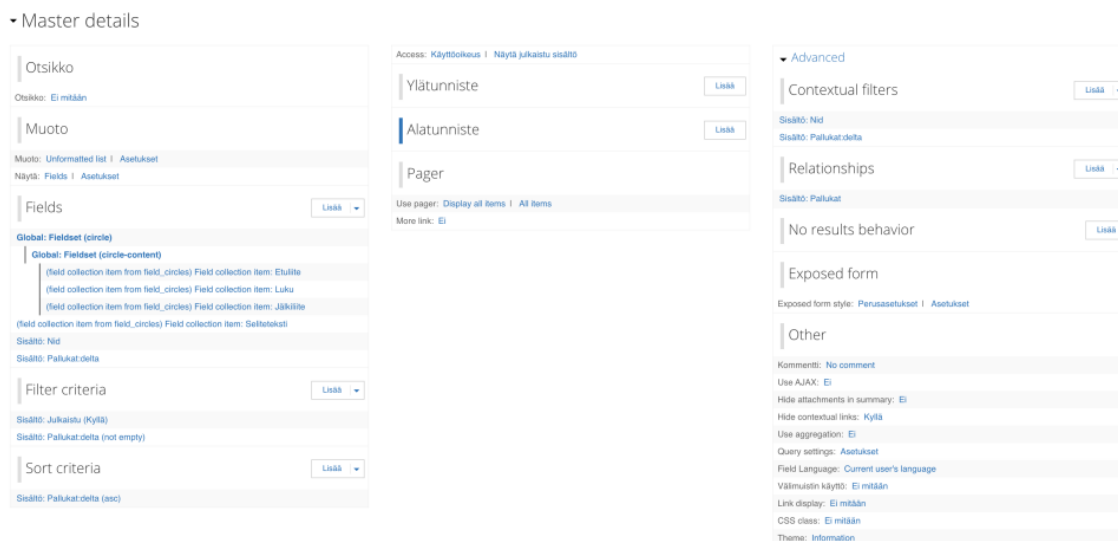
Drupalissa tarvitaan taksonomiatерmejä luokittelemaan ja suodattamaan sisältöä. Taksonomia on kuin kategoriamerkintä jollekin sivuston sisällölle, ja niitä lisätään erillisiin kirjastoihin. Taksonomialla on oma moduuli, jonka lataamalla saa käyttöön nämä kirjas-tot.

Uuden taksonomian määrittelyssä Drupaliin lisätään uusi sanasto, johon lisätään haluttu määrä termejä. Myös jostain sanaston termistä voidaan tehdä uusi sanasto, jolla on taas kategorisoituna termejä. Esimerkiksi sivustolle voidaan luoda sanasto puut, joka sisältää termit koivu, mänty ja kuusi. Tästä sanastosta voidaan ottaa termi koivu, josta rakennetaan edelleen uusi sanasto, jolla on termit vaivaiskoivu ja rauduskoivu.

Hyvinkin monimutkaiset taksonomiatерmeillä toteutetut viittaukset ovat mahdollisia Drupalissa, kun niitä käytetään näkymissä (ks. 3.2.7). [3, s. 201-204.]

3.10 Näkymät

Näkymät (engl. views) ovat laajan konsensuksen mukaan yksi Drupalin hienoimmista asioista. Se on vastaus, kun kysytään, miten tiettyä sisältöä voidaan näyttää oikeassa muodossa, oikea määrä ja monesta paikasta samaan aikaan haettuna. Näkymät ovat sekä nuoren että vanhan Drupal-kehittäjän parhaita ystäviä niiden helppokäyttöisyyden takia; ne eivät tarvitse riviäkään kirjoitettua koodia toimiakseen.



Kuva 5. Näkymät-sivun asetukset.

Näkymät toimivat graafisen käyttöliittymän kautta ja rakentavat hyvinkin monimutkaisia tietokantakutsuja. Niissä voidaan näyttää sisältöä kokonaisuudessaan listana tai valita tiettyjä kenttiä näytettäväksi. Listauksissa on paljon erilaisia vaihtoehtoja listauksen tyyliin ja suodatukseen. Esimerkiksi suodatusasetuksiin halutaan usein lisättävän kriteeri, joka tarkistaa, että näytettävä sisältö on julkaistu (ks. kuva 5 ja lukua 4.2.3).

Kaikkia näkymät-moduulin hienouksia on mahdotonta käydä läpi, mutta esimerkiksi seuraavissa esimerkkitilanteissa se on kiistatta loistava valinta tarkoitukseen:

- kuukausittainen sisällön arkistointi
- kaikki tietyn henkilön lisäämät blogikirjoitukset
- kaikkien uutinen-sisältötyypillä tehtyjen sisältöjen otsikko, yhteenveto ja pääkuva listattuna omalle sivulleen vanhin julkaisu ensin. Tämä sivujakajalla, joka listaa kerrallaan 10 vanhinta uutista sekä CSS-luokka lisättyinä. Päivitys 5 minuutin välein.

Näkymät tarjoaa siis hyvin paljon erilaisia tapoja hakea dataa tietokannasta ja esittää sitä eri tavoin. Vaikeus sen käytössä onkin suurimmaksi osin oikeiden olemassa olevien työkalujen löytäminen. [1, s. 49-50.]

3.11 Metatiedot

Verkkosivujen sisällössä keskitytään pääsääntöisesti siihen, mitä ihminen saa siitä irti. Enenevässä määrin nykykehityksessä keskitytään myös sivustolla esiintyvään meta-dataan. Metadatalta tarkoitetaan pääasiallisesti hakukoneita (esim. Google, Bing, DuckDuckGo) ja sosiaalisen median kanavia (esim. Twitter, Facebook, Instagram) varten lisättyä tietoa, joka antaa näkymättömässä muodossa kohteelle informaatiota sisällöstä. Metadatan tehokkuus riippuu monesta asiasta, joihin verkkokehityksessä tulee kiinnittää tarkkaan huomiota.

Verkkosivustolla esiintyvän tekstisisällön monipuolisuus, omapuolisuus ja termien oikeanlainen käyttö lisäävät hakukonenäkyvyyttä. Monipuolisuus sisällössä tarkoittaa erilaisten elementtien mahdollisimman kattavaa käyttöä; välitsikot, luettelot, tekstin lihavointi ja kursivointi sekä taulukot ovat yhdessä käytettynä hakukoneiden näkyvyyttä huomattavasti lisääviä. Sisällön omapuolisuus ja termien oikeanlainen käyttö kontekstissa lisäävät hakukonenäkyvyyttä sulkemalla pois muita vaihtoehtoja, koska hakukoneet pystyvät tunnistamaan tekstin alkuperäisen kirjoittajan, jos samaa sisältöä löytyy monelta sivustolta.

Vielä tarkemmin katseltuna hakukonenäkyvyyttä parantavia seikkoja sivuston sisällössä ovat sen pääotsikon, selainotsikon ja URL-osoitteen muoto, joista hakukone saa selkokielisiä otsikoita sivujen listaukseen. Myös linkkien käyttäminen, varsinkin ristiinlinkitykset, ovat hyvä tapa näyttää hakukoneelle, mikä sisältö on tärkeää.

Mediatiedostojen ja liitteiden kannalta tiedostojen nimeäminen ja alt-tekstit ovat optimaalisimmassa muodossa, kun ne on lisätty oikein. Alt-tekstillä tarkoitetaan tiedostoon erikseen lisättyä informaatiokenttää, jossa on tietoa tiedoston sisällöstä. Koska hakukoneet

eivät pysty ”näkemään” mediatiedostoja tai kuvaamaan ilman tiedoston nimeä ja alt-tekstiä lisättyä liitettä, on ensiarvoisen tärkeää lisätä nämä tiedot huolella. Esimerkiksi paljon käytetty kuvahaku nostaa esiin tuloksissa oikeaa sisältöä vain, jos kuvan tiedot ovat kunnossa. [12.]

Jokaiselle verkkosivuston yksittäiselle sivulle on mahdollista lisätä metakuvauskenttä. Se on noin 155 merkkiä pitkä tekstijono, jota hakukoneelle tarjotaan ensisijaisesti auttamaan hakua. Drupalissa metakuvauskenttiä voi hallita helposti Metatag-moduulilla. Se tarjoaa mahdollisuuden lisätä kuvaavan tekstijonon lisäksi jokaiselle yksittäiselle sivulle sosiaalisen median jakoon tarkoitetut metatiedot. Facebookilla ja Twitterillä on molemmilla omat tyylinsä hakea kuva, otsikko ja teksti omiin palveluihinsa jaetun linkin tiedoista, ja rakentaa siitä esikatselu. [12; 13.]

3.12 Teemaaminen

Yksi käyttöliittymän front-end-kehityksen suurimmista haasteista on sivuston teemaaminen. Tällä tarkoitetaan sivuston kokonaisvaltaista visuaalista ilmettä mukaan lukien yksittäisten elementtien näyttämistä.

Kuten aikaisemmin sanottu (ks. luku 3.2.3), lähtökohtana teemaamiseen on usein Drupalin perusasetuksissa määritetty, valmiiksi räätälöity teema. Jos Drupalin tarjoamista valmiista teemapakkauksista ei löydy sopivaa pohjaa teemalle, voi sen luoda kokonaan itse. Tämä vaatii toisaalta paljon enemmän töitä kehittäjältä.

Kaikki kustomoitu teemaaminen, jota sivustolle halutaan lisätä, vaatii käsin tehtyä koodia, minkä takia siinä vaaditaan taitoa käsitellä kieliä HTML, CSS ja PHP. Kaikki sivustolla näytettävä sisältö on PHP-tiedostojen aikaansaamaa, ja se esitetään HTML-elementteinä. Näille HTML-elementeille lisätään CSS-koodilla sääntöjä, jotka muokkaavat elementtien esitystapaa sivustolla. Jokaisella Drupalissa käytettävällä teemalla on olemassa seuraavat tärkeimmät tiedostot, joista säännöt visuaaliseen ilmeeseen tulevat:

- <teeman_nimi>.info, joka sisältää informaatiota teemasta, kuten teeman nimen, Drupal-version, jota teema tukee, JavaScript-tiedostojen polun sekä alueet (ks. luku 3.2.5), joita teema käyttää.
- page.tpl.php, joka sisältää HTML- ja PHP-koodina tiedon sivuston ulkoasusta ja rakenteesta. Tässä tiedostossa määritetään paljon eri riippuvuuksia, joita teemalla on muihin yksittäisiin tiedostoihin.
- node.tpl.php, joka määrittää, mitä kenttiä sivuston sisältötyypeistä nostetaan näkyviin sivuille.
- style.css sisältää kaikki CSS-säännöt, joita sivustolle on lisätty. CSS-tiedostoja on usein monia, ja ne kootaan yhteen style.css-tiedostoon.

[2, s. 197-200.]

4 Drupal-sivuston suunnittelu ja toteutus

Asiakkaalle toteutetun sivuston vastuualueet jaettiin projektin alussa selkeästi. Karhu Helsinki Oy:n vastuulla projektissa oli visuaalisen ilmeen määrittäminen layoutilla sekä teknisen toteutuksen suunnittelu sekä toteutus. Asiakkaan vastuulle jäi sivustolla näytettävän sisällön suunnittelu ja toimitus.

4.1 Rakenteen suunnittelu

Sivuston suunnittelun lähtökohtana oli selkeän vuosikatsaussivun toteuttaminen vakuutusalan asiakkaalle. Sivuston tuli olla käännettävä kolmelle kielelle suomeksi, ruotsiksi ja englanniksi. Vuosikatsaussivun tuli olla lähtökohtaisesti yksisivuinen näkymä yhtiön talousluvuista ja kuulumisista vuodelta 2018, päädyttiin rakenteeseen sisällyttämään kaksi

sisältötyyppiä: etusivun osio ja tavallinen sivu. Etusivun koko sisältö tuli rakentua osioista, jotka listataan näkymällä listautumaan allekkain.

Muu sivustolla näytettävä sisältö on ylä- ja alatunnisteissa. Ylätunnisteeseen päätettiin lisätä valikko, josta voi navigoida sivun eri osioihin, yhtiön logo ja kielenvaihtaja. Alatunnisteessa näkyvät linkit yhtiön verkkosivuille sekä sosiaalisen median kanavat.

Karhu Helsinki Oy:n graafisen suunnittelun tiimi valmisti näillä suunnitelluilla määritteillä asiakkaalle layoutin valmiin sivuston näkymistä. Tämän layoutin pohjalta sivuston tekninen toteuttaminen toteutui.

4.2 Tekninen toteutus

Karhu Helsinki Oy on toteuttanut samalle asiakkaalle myös aikaisemmin vuosikatsaus-sivun, joten luonnollisena jatkumona vanhoille sivuille myös tähän valittiin alustaksi Drupal 7. Myös sama palvelin ja tietokanta tulivat vanhalta sivustolta käyttöön uudelle sivustolle, joten uuden sivuston käyttöönotto oli helppoa.

4.2.1 Sisältötyypit

Koska etusivulla näytettävän sisällön määrä oli niin suuri, päätettiin sisältö jakaa myös rakenteellisesti osioihin. Etusivun osio -sisältötyyppiin valittiin tarkoitusta varten seuraavat kentät:

- Rabbit Hole, jolla voi määrittää osion näkymisen tietyillä sivuilla. Esimerkiksi jos sivua ei löydy tai se antaa virheilmoituksen, osio piilotetaan.
- Exclude Node Title, joka antaa mahdollisuuden poistaa osion otsikon näkymästä.

- Kieli, eli alavetovalikko, josta valitaan käyttöön yksi kolmesta käytettävästä kielestä: suomi, ruotsi, englanti.
- Title, eli osion otsikko, joka näytetään kaikkialla yksittäisiä osioita listaavilla sivuilla.
- Kuva, joka näytetään osion taustalla. Kuvan päällä näytetään muu sisältö.
- Body, eli leipäteksti sisältää WYSIWYG-kentän, johon voi lisätä osiossa näytettävää sisältöä.
- Linkit, eli linkkilaatikat, joiden kohde on mm. vakuutusyhtiön julkaisemiin raportteihin.
- Pallukat, joissa näytetään avainlukuja yhtiön menneeltä vuodelta.
- Edistyneet asetukset: CSS Classes, jossa voidaan määrittää erillisiä osiossa käytettäviä CSS-luokkia, kuten text-white, jonka avulla sivuston teemassa määritellään osion tekstit valkoisiksi.
- Polkujen asetukset, jossa voidaan valita osiolle oma osoitealias tai generoida se automaattisesti.
- URL Redirects, jossa voidaan määrittää uudelleenohjaus tältä sivulta muualle.
- Meta tags, jossa voidaan määrittää osion metatiedot. Automaattisesti metatietoihin lisätään sivun otsikko, kuvaus sekä sosiaalisen median jaossa käytettävät tiedot.

Tavallinen sivu -sisältötyyppi päädyttiin lisäämään sivustolle käytettäväksi sivuilla, jotka haluttiin näyttää omina kokonaisuuksinaan etusivun ulkopuolella. Tällä sisältötyypillä

tehtyjä sivuja on huomattavasti vähemmän ja niihin lisättävä sisältökin on suppeampaa. Siihen lisättiin seuraavat kentät, jotka ovat kaikki myös käytössä etusivun osio -sisältötyypissä:

- Rabbit Hole
- kieli
- title
- kuva
- Body
- polkujen asetukset
- URL Redirects
- Meta tags.

4.2.2 Näkymät

Sivustolle lisättiin kaksi näkymää etusivulle. Sections-nimellä oleva näkymä listaa etusivun osio -sisältötyypillä tehdyt sisällöt etusivulle kokonaisuudeksi. Kaikista sisällöistä näytetään kentät otsikko, kuva, body ja linkit allekkain.

Toinen sivulle lisätty näkymä on pallukat, jossa sisältötyyppiin lisätyt kentät etuliite, luku, jälkiliike ja seliteteksti näytetään yhdessä kokonaisuudessa (ks. kuva 2). Esimerkkikuvassa sisältöön on syötetty kenttiin seuraavat arvot: etuliite: noin, luku: 70, jälkiliite: %, seliteteksti: kuntoutujista palasi takaisin työelämään.



Kuva 5. Esimerkki opinnäytetyössä käytetystä yksittäisestä pallukkanäkymästä.

4.2.3 Käyttäjäroolit

Sivustolla on asennuksen yhteydessä lisätty admin-käyttäjä, jolla on järjestelmävalvojan oikeudet. Tämän tason käyttäjällä on oikeus muokata kaikkea, mitä Drupalin käyttöliittymässä voi muokata. Myös asiakkaalle tarvittiin päivittäjä-tason käyttäjä, jotta myös heidän päästään sivuston sisältöä pystytään muokkaamaan, mutta pääsyä arkaluontoiisiin asetuksiin ei ole.

5 Yhteenveto

Insinööriyössä pyrittiin esittämään kattavasti Drupal-sisällönhallintajärjestelmää eteneväällä loogisesti sen määritelmästä, historiasta, hyödyistä ja teknisistä ominaisuuksista aina yritykselle valmistetun valmiin projektin läpikäymiseen.

Projektina toteutettiin teknisesti ja visuaalisesti koherentti kokonaisuus vakuutusalan yritykselle vuosikatsaussivuston muodossa. Julkaisujärjestelmänä toimi Drupal 7, joka osoittautui tähän projektiin erittäin hyvin sopivaksi. Jo projektin suunnitteluvaiheessa määritetty tekninen alusta vei toteutusta aina oikeaan suuntaan sen onnistumisen kannalta. Karhu Helsinki Oy:n projektin toteutumisesta vastaava projektipäällikkö sekä yrityksen graafisen suunnittelun tiimi tekivät erinomaista työtä tuodakseen projektin tekniset ja visuaaliset tarpeet ilmi.

Sivuston tilannut asiakas on ollut aikaisemminkin tyytyväinen tilaamiinsa Drupal 7 -sivustoihin, eikä tämä kerta ollut poikkeus; kaikki projektiin liittyvät asiat saatiin valmiiksi määräajassa, ja sivuston käyttöönotto sujui mutkitta. Projektin teknisen toteutuksen jälkeen sivuston lisääminen valmiina odottavalle palvelimelle tietokantayhteyksineen toimi hyvin. Lopputulemana projektista oli toimiva sivusto, jota voidaan jatkokehittää mahdollisesti myöhemmin, jos tarpeen tulee.

Lähteet

- 1 Melancon, Benjamin; Luisi, Jacine; Negyesi, Karoly; Anderson, Greg; Somers, Bojhan; Corlosquet, Stephane; Freudenberg, Stefan; Lauer, Michelle; Carlevale, Ed; Loretan, Florian; Nordin, Dani; Szrama, Ryan; Stewart, Susan; Strawn, Jake; Travis, Brian; Hakimzadeh, Dan; Scavarda, Amye; Albala, Albert; Micka, Allie; Douglass, Robert; Monks, Robin; Scholten, Roy; Boyer, Sam; Gifford, Mike & Sarahe, Claudina. 2011. The Definitive Guide to Drupal 7. New York: Springer Science+Business Media.
- 2 Tomlinson, Todd. 2010. Beginning Drupal 7. New York: Springer-Verlang.
- 3 Shreves, Ric & Dunwoodie, Brice. 2011. Drupal 7 Bible. Indianapolis, Indiana: Wiley Publishing Inc.
- 4 Managing users. Verkkoaineisto. <<https://www.drupal.org/docs/7/managing-users>>. Luettu 2.9.2019.
- 5 Bryan Murley, 2013. WordPress again most frequently used CMS among Online Pacemaker finalists. Verkkoaineisto. <<https://collegemediainnovation.wordpress.com/2013/04/18/wordpress-again-most-frequently-used-cms-among-online-pacemaker-finalists/>>. 18.4.2013. Luettu 1.9.2019.
- 6 5 Reasons to use Drupal vs. WordPress, 2017. Verkkoaineisto. <<https://www.duoconsulting.com/blog/5-reasons-use-drupal-vs-wordpress>>. 4.1.2017. Luettu 3.9.2019.
- 7 System requirements. Verkkoaineisto. <<https://www.drupal.org/docs/8/system-requirements>>. Luettu 3.9.2019.
- 8 System requirements. Verkkoaineisto. <<https://www.drupal.org/docs/7/system-requirements>>. Luettu 3.9.2019.
- 9 Documentation. Verkkoaineisto. <<https://www.drupal.org/docs/8/install>>. Luettu 3.10.2019.
- 10 Installing a Drupal 8 module. Verkkoaineisto. <<https://docs.acquia.com/resource/module-install-d8/>>. Luettu 3.10.2019.

- 11 Our history. Verkkoaineisto. <<https://www.drupal.org/about/history>>. Luettu 7.10.2019.
- 12 Sanna Lindholm, 2017. Näin tuotat hakukoneystävällistä sisältöä. Verkkoaineisto. <<https://www.karhuhelsinki.fi/blogi/nain-tuotat-hakukoneystavallista-sisaltoa>>. 14.11.2017. Luettu 8.10.2019.
- 13 Metatag. 2019. Verkkoaineisto. <<https://www.drupal.org/project/metatag>>. Luettu 8.10.2019.