

WordPress Gutenberg – perustoiminnallisuudet ja niiden laajentaminen

Jimi Heitto

OPINNÄYTETYÖ
Huhtikuu 2019

Tietojenkäsittely
Web-palvelut

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely
Web-palvelut

HEITTO, JIMI:

WordPress Gutenberg – perustoiminnallisuudet ja niiden laajentaminen

Opinnäytetyö 29 sivua
Huhtikuu 2019

Opinnäytetyö tutki ja analysoi WordPressin 5.0 version mukana julkaistun Gutenberg-muokkaimen käyttöä ja laajennusmahdollisuuksia. Opinnäytetyön tavoitteena oli tuottaa toimeksiantajayrityksen WordPress-kehittäjille ohjeisto, joka kuvaa, miten Gutenbergia laajennetaan WordPressin omilla toiminnoilla ja Advanced Custom Fields -lisäosan tuomilla toiminnolla. Opinnäytetyössä tutkittiin ja esitettiin, miten Advanced Custom Fields -lisäosa yhdistetään Gutenberg-muokkaimen. Tarkoituksena oli kuvailla uutta muokkainta ja sen ominaisuuksia sekä tutkia ja esittää, miten Gutenberg-muokkainta muokattiin ja laajennettiin luomalla omia lohkoja ja tyyliä muokkaimen.

Opinnäytetyössä selvitettiin omien lohkojen ja lohkotyylien luomista Gutenberg-muokkaimen sisään. Lisäksi opinnäytetyössä esitettiin erilaisia tapoja, joilla voidaan hallita Gutenberg-muokkaimen lohkojen näkyvyyttä, hallita muokkaimen väripalettiä ja poistaa käytöstä muokkaimen valmiita ominaisuuksia.

Opinnäytetyössä havaittiin, että Gutenberg-muokkaimen laajentaminen vaatii ohjelmointiosaamista ja WordPressin perustoimintojen ymmärtämistä. Gutenbergia muokkaamalla ja laajentamalla saadaan kehitettyä omia sisällönhallintamahdollisuuksia WordPress-asennukseen ja tapoja hallita sisältöä paremmin.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Business Information Systems
Web Services

HEITTO, JIMI:
WordPress Gutenberg – Basic Functionality and Extensions

Bachelor's thesis 29 pages
April 2019

This thesis examines and analyzes the usage of Gutenberg editor and its expansion capabilities released with the WordPress version 5.0. The aim of the thesis was to provide instructions for WordPress developers in the client company on how to extend Gutenberg with WordPress's own functions and the functions included in the Advanced Custom Fields plugin. The purpose of the thesis report was to study how to combine the Advanced Custom Fields plugin with the Gutenberg editor, to describe the new editor and its features, as well as how to edit and expand the editor by creating own blocks and styles inside it.

The results were a variety of options for managing the visibility of the Gutenberg editor blocks, creating one's own blocks and styles inside Gutenberg, overriding the color palette in the Gutenberg editor, and disabling Gutenberg's redundant properties.

The thesis study showed that expanding the Gutenberg editor requires programming expertise and understanding the basic functions of WordPress. By customizing and expanding Gutenberg, it is possible to develop individual content management capabilities for WordPress installation and manage the content of the WordPress environment more easily.

SISÄLLYS

1	JOHDANTO	6
2	SISÄLLÖNHALLINTAJÄRJESTELMÄ	7
3	WORDPRESS	8
4	KLASSINEN MUOKKAIN.....	9
5	GUTENBERG-MUOKKAIN.....	11
	5.1 Kuvaus	11
	5.2 Lohkot	11
6	ADVANCED CUSTOM FIELDS PRO	14
	6.1 Kuvaus	14
	6.2 Lisäkentät.....	15
	6.3 Kenttien rekisteröinti ja esittäminen.....	15
7	GUTENBERG-MUOKKAIMEN LAAJENTAMINEN	21
	7.1 Lohkojen luominen ACF-lisäosalla	21
	7.2 Lohkojen näkyvyyden määrittäminen	22
	7.3 Lohkojen tyylien luominen ja poistaminen.....	23
	7.4 Tyylit Gutenbergiin	24
	7.5 Uuden väripaletin luominen Gutenbergiin	26
8	POHDINTA	28
	LÄHTEET.....	29

ERITYISSANASTO

ACF	Advanced Custom Fields - WordPress lisäosa
FTP	Tiedonsiirtoprotokolla
Funktio	Tietyn toiminnon suorittava käskysarja
Parametri	Funktiolle välitettävä yksittäinen tieto
Sivupohja	Koodillisesti toistettava mallitiedosto
Sivurakentaja	Verkkosivun rakennustyökalu
Vimpain	WordPressin sisällön ja ominaisuuksien esittelytyökalu
Workflow	Työnkulku
Wysiwyg	“What you see is what you get”-sisältökenttä

1 JOHDANTO

WordPressin uusi versio 5.0 "Bebo" julkaistiin ladattavaksi ja asennettavaksi 6. joulukuuta 2018. Tämän version mukana julkaistu Gutenberg-muokkain korvaa jatkossa WordPressin alkuperäisen klassisen muokkaimen ja tuo samalla paljon uusia ominaisuuksia monipuolisen sisällön luomiseen, kehittämiseen ja esittämiseen. Gutenberg-muokkain tulee vähintäänkin muuttamaan kehittäjien työskuvaa ja tapoja kehittää WordPress-sivustoja.

Uusi lohkopohjainen Gutenberg-muokkain tuo mukanaan joustavuutta ja vapautta sisällönhallintaan ja tämän esittämiseen. Gutenberg ei suoraan muuta sivuston sisällön ulkoasua, vaan sisällön asettelua. Gutenberg-muokkaimen on tarkoitus korvata klassinen muokkain vuoteen 2021 mennessä, mutta tähän asti klassista muokkainta voi käyttää lisäosan "Classic Editor" avulla tai Gutenberg-muokkaimen sisällä omana lohkonaan.

Gutenberg itsessään tuo kattavan määrän uusia ominaisuuksia sisällönhallintaan, mutta näitä voidaan vielä laajentaa entisestään ja Gutenberg-muokkainta ja sen käyttöä voidaan muokata paremmin hallittavaksi kokonaisuudeksi. Advanced Custom Fields -lisäosa tuo WordPressiin asennettuna kattavat työkalut Gutenberg-muokkaimen muokkaamiseen ja laajentamiseen.

Opinnäytetyön tarkoituksena on tutkia, analysoida ja esittää Gutenberg-muokkaimen käyttöä, mukauttamista ja laajennusmahdollisuuksia Advanced Custom Fields -lisäosalla ja WordPressin omilla toiminnoilla. Opinnäytetyön tavoitteena toimii toimeksiantajayrityksen WordPress-kehittäjille suunnatun ohjeen toteuttaminen. Opinnäytetyön lähtökohtana on aloitteleva WordPress-kehittäjä, joten lähestymistapa ja näkökulma opinnäytetyössä on tämän mukainen.

2 SISÄLLÖNHALLINTAJÄRJESTELMÄ

Sisällönhallintajärjestelmä on sovellus, jota käytetään web-sisällön hallintaan. Sisällönhallintajärjestelmän perusominaisuuksia ovat sisällön luominen, muokkaaminen, tallennus ja julkaisu järjestelmän sisältä esitettäväksi. Yleisiä ominaisuuksia ovat myös järjestelmän hallinnalliset työkalut käyttäjien oikeuksien asettamiseen.

Sisällönhallintajärjestelmä mahdollistaa sivuston sisällönhallinnan ilman ohjelmointiosaamista tarjoamalla automaatiota ja työkaluja tehokkaaseen sisällönhallintaan. Järjestelmä koostuu erilaisista osista, joihin kuuluvat muokkausliittymä, tiedon arkisto ja erilaiset mekanismit, kuten julkaisumekanismit. Järjestelmä koostuu useista yksittäisistä komponenteista, jotka voivat toimia itsenäisesti erillisinä osina järjestelmässä, mutta jotka muodostavat yhdessä yhden yhtenäisen kokonaisuuden, sisällönhallintajärjestelmän (Barker, 2016, 32).

Yleisesti sisällönhallintajärjestelmän ominaisuuksiin kuuluvat juuri sisällönhallinnan työkalut. Järjestelmä seuraa, missä sisältö on, missä tilassa se on ja kennellä on pääsy kyseiseen sisältöön. Järjestelmän ydintoimintoja ovat sisällön käyttöoikeudet, statuksenhallinta, workflow, versiointi, riippuvuudenhallinta, haku- ja organisointitoiminnot.

Sisällönhallintajärjestelmän ominaisuuksiin kuuluu, että järjestelmään lisätty data tallennetaan yleisesti tietokantaan ja esitetään loogisesti. Tunnettuja sisällönhallintajärjestelmiä ovat WordPress, Drupal, Joomla ja Squarespace.

3 WORDPRESS

Vuonna 2003 julkaistu WordPress on PHP-ohjelmointikielellä kirjoitettu avoimen lähdekoodin sisällönhallintajärjestelmä, joka luotiin aluksi blogisivustojen julkaisua ja ylläpitoa varten, mutta järjestelmän käyttökohteet ovat laajentuneet huomattavasti tämän kehityksen aikana. WordPress on vapaasti ladattavissa ja asennettavissa, mikä on yksi tekijä siinä, että yli 33 % internetin sivustoista on valinnut sisällönhallintajärjestelmäkseen WordPressin.

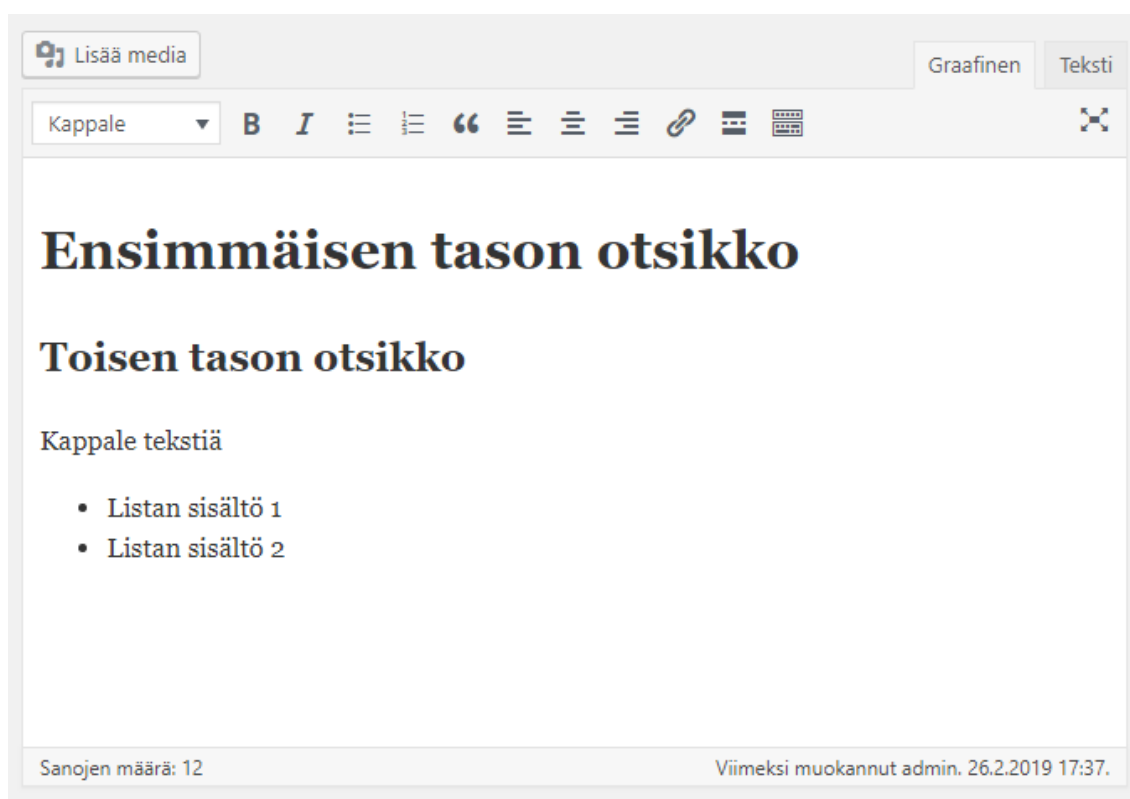
Yksi WordPressin pääominaisuuksista on käyttäjän syöttämän sisällön esittäminen. Sen sijaan, että käyttäjä loisi oman verkkosivun alusta, antaa tämä WordPressille raaka-aineet, joita ovat teksti ja kuvat, ja näistä aineksista WordPress kokoaa esitettävän verkkosivun (MacDonald, 2014, 20).

WordPress on tehokas sisällönhallintajärjestelmä, jolla on useita eri käyttökohteita. WordPressin avulla käyttäjä voi luoda haluamansa tyyppisen sivuston, kuten yritys-, blogi- tai uutissivuston, ansioluettelon tai portfolion. Myös monimutkaisemmatkin kokonaisuudet, kuten verkkokaupat, materiaalikirjastot, intra- ja extranetit ovat toteutettavissa WordPressillä.

WordPress on tehty laajennettavaksi ja mukautettavaksi erilaisilla teemoilla, jotka pääosin määrittävät miltä sivusto näyttää ja lisäosilla, jotka määrittävät ja luovat sivustolle toiminnallisuuksia. WordPress on GPLv2-lisensoitu, mikä tarkoittaa sitä, että sitä voi vapaasti muokata ilmaiseksi. Tämän vuoksi WordPressillä on useita eri kehittäjiä, jotka osallistuvat itse WordPressin kehittämiseen ja kehittävät tähän myös teemoja ja lisäosia. WordPressiin on julkaistu yli 50 000 ilmaista lisäosaa ja yli 5 000 ilmaista teemaa.

4 KLASSINEN MUOKKAIN

WordPressin sisältökenttä, eli klassinen muokkain (kuva 1), on WordPressin alkuperäinen oletusmuokkain ennen Gutenberg-päivitystä. Klassinen muokkain antaa järjestelmän käyttäjälle yksinkertaiset työkalut sisällön luomiseen ja hallintaan. Muokkaimen syötetään sivuston sivujen ja artikkelien olennainen sisältö, kuten teksti- ja kuvasisällöt.



KUVA 1. WordPressin klassinen muokkain

Klassinen muokkain toimii kuten tavallinen tietokoneen tekstinkäsittelyohjelma. Muokkaimen avulla tekstiä voidaan asemoida, muotoilla ja tyyllitellä. Muokkaimen avulla sisältöön voidaan myös lisätä elementtejä, kuten otsikot, listat, sitaattit, linkit ja kuvat, jotka järjestelmä muuttaa HTML-elementeiksi muun sisällön ohella esitettäväksi.

Klassinen muokkain on yhdistetty avoimen lähdekoodin HTML-muokkaimen, TinyMCE-tekstinkäsittelyohjelmaan. TinyMCE:n ominaisuutena on HTML-elementtien muuntaminen muokkausinstansseiksi ja esittää nämä Wysiwyg-tyyliin. Klassisessa muokkaimessa on kaksi eri puolta, graafinen ja teksti, joiden välillä

käyttäjää voi vaihdella halutessaan helposti. Graafinen- eli Wysiwyg-muokkain esittää sisällön, miten se sivustolla ilman CSS-tyylimäärittelyjä näyttää. Enemmän kehittäjille suunnattu tekstimuokkain esittää tekstin HTML-elementteinä.

5 GUTENBERG-MUOKKAIN

5.1 Kuvaus

Gutenberg-muokkaimen nimi tulee WordPressin kehittäjien Gutenberg-projektista, jonka tarkoituksena oli luoda kokonaan uusi muokkauskokemus WordPressille. Tätä tarkoitusta noudattaen klassisen muokkaimen tilalle kehitettiin modulaarisempi Gutenberg-muokkain, jonka avulla voidaan tehdä näyttävämpiä ja monipuolisempia asetelmia sivujen ja artikkelien sisällöille. Gutenberg tuo mukanaan uudistetun kirjoituskokemuksen ja on ensimmäinen askel kohti yhtenäistä sivustonhallintakokemusta mahdollistaen vapaamman sisällön tuottamisen ja asettelun (Wahalahti. 2018). Ennen monipuoliseen sivu- tai artikkeliasetteluun tarvittiin avuksi WordPressin vimpaimia, lisäkenttiä tai lisäosia.

Gutenbergin julkaisun ansiosta monipuolinen sisällönhallinta ja -asemointi onnistuu nyt käyttäen vain ydinmuokkainta. WP-palvelun Toni Laakson mukaan Gutenbergin kehitystyön lähtökohtana ovat yksinkertaiset periaatteet ja tarkoituksena on tehdä sisällöntuottamisesta WordPress-sivustoille vaivatonta, helpottaa erilaisten lisäosien käytön oppimista, sekä poistaa yleinen ilmiö: aina ei voi olla varma mistä mitään kohtaa sivustosta päivitetään (Laakso, 2018).

5.2 Lohkot

Gutenberg käyttää sisällön esittämiseen ja järjestämiseen lohkoja. Klassinen muokkain toimi yksittäisenä tekstisisältöikkunana, kun taas lohkot ovat toistettavia modulaarisia elementtejä sisältöalueen sisällä, joiden yksittäinen sisältötyyppi määrittyy lohkon tyyppin mukaan. Lohkot (kuva 2) ovat helposti ymmärrettäviä, toistettavia ja siirrettäviä sisältöalueen sisällä. Jokainen Gutenbergin lohko on luotu palvelemaan samaa tarkoitusta, joka on tietynlaisen sisällön esittämistä. Yhden lohkon tarkoituksena on yleisesti yksittäisen HTML-elementin esittäminen, kuten otsikko- tai listaelementin.

The screenshot shows the Gutenberg editor interface. On the left, there is a document with two heading levels: "Ensimmäisen tason otsikko" (H1) and "Toisen tason otsikko" (H2). Below the H2 heading, there is a text block "Kappale tekstiä" followed by a bulleted list with two items: "Listan sisältö 1" and "Listan sisältö 2". Below the list, there is a prompt: "Aloita kirjoittaminen tai kirjoita / valitaksesi lohkon". On the right, there is a sidebar with a "Lohko" (Block) menu. The menu is open, showing options for "Asiakirja" (Document) and "Lohko" (Block). Under "Lohko", there are sections for "Tila ja näkyvyys" (Status and Visibility) with a dropdown arrow, "Näkyvyys" (Visibility) with a link to "Julkinen" (Public), "Julkaise" (Publish) with a timestamp "26.2.2019 17:40", a "Siirrä roskakoriin" (Move to trash) button, "Kestolinkki" (Permanent link) with a dropdown arrow, "Artikkelikuva" (Article image) with a dropdown arrow, and "Sivun ominaisuudet" (Page settings) with a dropdown arrow.

KUVA 2. Gutenberg-muokkain

Lohkotyytit on Gutenbergin sisällä jaoteltu neljään eri kategoriaan: yleiset lohkot, muotoilu, asettelun elementit ja vimpaimet. Gutenbergin sisällä on myös käytetyimmät lohkot -kategoria, jonka sisään kerääntyvät käyttäjän yhdeksän käytettyä lohkoa. Yleisiä Gutenberg-lohkoja ovat:

Kappale

Lisää kappale-elementin, mikä vastaa tavallista tekstikappaletta.

Lainaus

Lisää lainauksen ja sitaatin.

Lista

Lisää listan, eli luettelon, joka voi olla järjestetty tai järjestämätön.

Kuva

Lisää kuvaelementin ja tähän määritetyn kuvan.

Otsikko

Lisää otsikkoelementin, joka voi olla tasoltaan H1, H2, H3, H4, H5 tai H6.

Video

Lisää videoelementin ja tähän määritetyn videon.

Gutenbergin lohkojen käyttö vastaa karkeasti sivurakentajan käyttöä. Jokainen lohko on oma yksittäinen osa kokonaisuutta. Lohkojen asemaan Gutenbergissa voidaan vaikuttaa raahaamalla niitä toisten lohkojen ylä- tai alapuolelle. Lohkojen aseman avulla määritetään esitettävän sisällön asemaa ja ulkoasua sivustolla. Lohkoja voi siirtää, muokata ja mukauttaa vapaasti muokkaustilassa, mikä on suurin erottava tekijä klassisen muokkaimen ja Gutenberg-muokkaimen välillä.

6 ADVANCED CUSTOM FIELDS PRO

6.1 Kuvaus

Advanced Custom Fields, lyhyesti ACF, on tehokas kehittäjille suunnattu WordPress-lisäosa, joka lisää WordPressiin mahdollisuuden rekisteröidä ja esittää mukautettuja lisäkenttiä ja näille asetettuja arvoja.

ACF-lisäosaa voidaan käyttää WordPressin sisällön ylläpitämiseen ja sen hallinnan monipuolistamiseen. Lisäosan mahdollisia käyttötapoja on useita, sillä ACF sisältää yli 30 erilaista kenttätyyppiä. Kenttiä voidaan lisätä, ylläpitää ja näyttää missä vain sivuston osiossa. Lisäkenttien tarkoitus on mahdollistaa mukautetun sisällön luomisen ja esittämisen sivustolla.

ACF on tärkeä lisäosa WordPress-kehityksessä. Tehokas ja toimiva sisällönhallinta saadaan aikaan, kun WordPressin oletustoimintoja laajennetaan lisäkentillä. Lisäkenttien luominen ja näyttäminen sivustolla vaatii ohjelmointiosaamista, mutta antaa kehittäjille mahdollisuuden rakentaa kenttien avulla käyttäjäystävällisiä ja turvallisia ylläpitomahdollisuuksia käyttäjille, joilta ei ohjelmointiosaamista löydy.

ACF:n avulla voidaan myös automatisoida sisällön tulostamista, jolloin samaa koodia ei tarvitse kirjoittaa useaan otteeseen ja itse sisällön ylläpitäminen tapahtuu sisällönhallintajärjestelmässä, eikä palvelimen puolella esimerkiksi FTP-yhteydellä. ACF mahdollistaa myös samojen lisäkenttien käytön eri sivuilla, sivupohjilla tai artikkeleilla, jolloin järjestelmään voidaan luoda helposti yhtenäisiä loogisia sisältöalueita vaihtelevilla sisällöillä.

Advanced Custom Fields Pro on maksullinen lisäosa, joka laajentaa tavallista ACF-lisäosaa uusilla kentillä ja ominaisuuksilla. Tämä opinnäytetyö käsittelee lisäosan Pro-version käyttöä, joten kaikki esitellyt toiminnot eivät välttämättä toimi lisäosan perusversiolla.

6.2 Lisäkentät

Kentät ovat lisäosan ominaisuuksia, joiden avulla voidaan yksinkertaisesti luoda sisällönhallintakenttiä WordPressin omien kenttien lisäksi ja tuoda niiden sisältöä näkyviin eri puolilla sivustoa. Yleensä tietyt kentät kohdistetaan lisäosan asetuksista näkymään tietyssä sivupohjassa, mutta niistä voidaan tehdä myös ”globaaleja”, tarkoittaen, että niiden sisältö näkyy yhdestä paikasta asetettuna sivustolla riippumatta missä sivupohjassa tai millä sivulla ollaan.

ACF:n kenttiä on erilaisia, kuten tekstikentät: yhden rivin teksti, tekstialue tai WordPressin sisältö- eli Wysiwyg-kenttä. Kenttien avulla voidaan sivustolla näyttää muun muassa kuvia, tiedostoja, kalenteriaikoja, valintalistoja, artikkelien ja sivujen sisältöjä sekä erilaisia toistin- ja sisältökenttiä joustavan sisällön tulostamiseen. Kenttien määrittämistyyppiä on siis useita, ja niiden sisällön tulostamislogiikkaa ja ulkoasua voidaan vapaasti muokata PHP-ohjelmointikielellä ja CSS-tyylimäärittelyillä.

6.3 Kenttien rekisteröinti ja esittäminen

WordPressin teema määrittää sivuston ulkoasua ja rakennetta. Lisäkenttiä voidaan rekisteröidä joko asennuksen hallintapuolelta lisäosan graafisesta liittymästä tai PHP-ohjelmointikielellä muokkaamalla teemaa. Teeman muokkaaminen tapahtuu palvelimen puolelta sivuston hakemistokansioista teeman omasta kansioista. Muokkaamalla `style.css`-tiedostoa voidaan sivustolle määrittää uusia tyyliä ja muokkaamalla `functions.php`-tiedostoa voidaan sivustolle lisätä uusia toiminnallisuuksia, kuten lisäkenttiä. Kenttien rekisteröinti PHP-ohjelmointikielellä tapahtuu lisäämällä rekisteröintifunktio teeman `functions.php`-tiedostoon. Tämä opinnäytetyö käsittelee kenttien rekisteröintiä ainoastaan PHP:lla. Havainnollistavana esimerkkinä käsitellään ”Team Member”-kentän rekisteröintiä ja esittämistä. Yleinen ohjelmointikieli on englanti, joten esitetyt funktiot ovat englanniksi.

Lisäkentät rekisteröidään PHP:lla kutsumalla ”`acf_add_local_field_group`”-funktiota. Kenttiä rekisteröitäessä täytyy niille määrittää oleelliset tiedot (kuva 3), kuten avain, nimi, tyyppi ja sijainti. Kenttiä kutsutaan avain- tai nimiparametrin avulla

ja tyyppiparametri määrittää, mitä kyseisen kentän avulla halutaan esittää. Lo-
kaatioparametri määrittää, missä rekisteröity kenttä esitetään.

```
if (function_exists('acf_add_local_field_group')) {
    acf_add_local_field_group(array(
        'key'      => 'group_team_member',
        'title'    => 'Team Member',
        'fields'   => array(
            array(
                'key'      => 'field_team_member_name',
                'label'    => 'Name',
                'name'     => 'member_name',
                'type'     => 'text',
            ),
            array(
                'key'      => 'field_team_member_image',
                'label'    => 'Image',
                'name'     => 'member_image',
                'type'     => 'image',
                'return_format' => 'array',
                'preview_size' => 'thumbnail',
                'library'   => 'all',
            ),
        ),
        'location' => array(
            array(
                array(
                    'param'    => 'post_type',
                    'operator' => '==',
                    'value'    => 'page',
                ),
            ),
        ),
    ));
}
```

KUVA 3. ACF-lisäkenttien rekisteröinti

Ryhmän "Team Member" kentät on rekisteröity näkymään "Page"-artikkelityy-
pissä, eli sivuston sivuilla. Lisäkentät näkyvät rekisteröinnin jälkeen WordPressin
hallintapuolella tämän artikkelityypin muokkaimen alapuolella (kuvat 4 ja 5).

⊕ ↶ ↷ ⓘ ☰

Tallenna luonnos Esikatselu Julkaise... ⚙️ ⋮

Team member

Aloita kirjoittaminen tai kirjoita / valitaksesi lohkon

Team member ^

Member name

Member image

Ei kuvia valittu

Asiakirja Lohko x

Tila ja näkyvyys ^

Näkyvyys [Julkinen](#)

Julkaise [Valittomästi](#)

Tarkastettavana

Artikkelikuva v

Sivun ominaisuudet v

KUVA 4. "Team member"-lisäkenttien sijainti Gutenberg-muokkaimen alapuolella

Team member ^

Member name

Member image

KUVA 5. "Team member"-lisäkentät tiedot syötettynä

Rekisteröidyt kentät halutaan myös esittää sivustolla. Koska yllä olevat kentät on rekisteröity artikkelityypille "Sivu", kutsutaan niitä teeman hakemistossa tämän

artikkelityypin sivupohjassa, yleensä "page.php", kutsumalla "get_field()" -funktiota (kuva 6). Funktiolle "get_field()" syötetään esitettävän kentän nimi, jonka avulla funktio tunnistaa, mitä kenttää kyseisessä kohtaa halutaan esittää.

```
<div class="team-member">
  <?php
    $name = get_field('member_name');
    $image = get_field('member_image');

    if (!empty($image)) {
      echo wp_get_attachment_image(
        $image['ID'],
        'thumbnail',
        null,
        array('class' => 'team-member-image')
      );
    }
    if (!empty($name)) {
      echo '<h4>' . esc_html($name) . '</h4>';
    }
  ?>
</div>
```

KUVA 6. ACF-kenttien esittäminen sivupohjassa

Kenttien tietoja hallitaan sivun muokkaustilassa ja esitetään sivustolla. Sivupohjan avulla samaa kenttää voidaan esittää samaa sivupohjaa käyttävillä sivuilla yksittäisin sivun muokkaustilassa määritetyin tiedoin. Kun kenttään on sivun muokkaustilassa asetettu tietoa ja kenttä kutsutaan sivupohjassa, tulee se esiin sivustolla (kuva 7). Kuvassa näkyvät sivun muokkaustilassa asetetut kuva ja teksti.



Member name

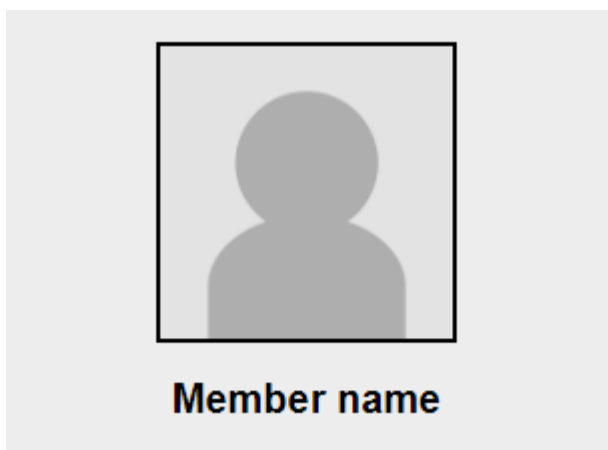
KUVA 7. "Team member"-kenttä sivustolla

Esitettävän kentän tiedot perivät tyylinsä WordPressin tyylitiedostosta. Kentän ympäröivälle div-elementille on sivupohjassa asetettu luokka "team-member", joten tämän luokan avulla voidaan kentän tiedoille tehdä CSS-tyylimäärittelyjä style.css-tiedostossa (kuva 8).

```
.team-member {  
    background: #ededed;  
    text-align: center;  
}  
  
.team-member img {  
    margin: 0 0 15px 0;  
    border: 2px solid black;  
}  
  
.team-member h4 {  
    font-size: 20px;  
}
```

KUVA 8. "Team member"-tyylimäärittelyt

Tyylimäärittelyiden avulla voidaan yliajaa teeman asettamia tyylejä kentän elementeille, kuten "Team member"-kentän kuvalle ja otsikolle, jolloin näiden ulkoasuun voidaan vaikuttaa halutulla tavalla. Tyylitiedostossa style.css asetetut tyylimäärittelyt muuttavat esitettävän kentän ulkoasua halutuksi (kuva 9). Kuvassa näkyvät "Team member"-lisäkentän elementtien muokatut välistykset, fontin koot ja taustaväri.



KUVA 9. "Team member"-kenttä tyyliteltyinä sivustolla

7 GUTENBERG-MUOKKAIMEN LAAJENTAMINEN

7.1 Lohkojen luominen ACF-lisäosalla

Oletuksena ACF-kentät sijoittuvat Gutenberg-muokkaimen alapuolelle, mutta Gutenbergia laajentamalla voidaan kenttiä rekisteröidä Gutenbergin sisään ja näin kentistä tulee modulaarisia ja helposti toistettavia. ACF mahdollistaa Gutenberg-lohkojen luomisen PHP:lla. Lohkojen luominen lisäosalla vaatii lisäosasta vähintään version 5.8.0. Gutenberg-lohko luodaan kutsumalla "acf_register_block"-funktioita (kuva 10). Funktion sisällä määritetään vähintään lohkon nimi, sivupohja, jonka avulla lohko esitetään ja lohkon kategoria Gutenberg-muokkaimen sisällä, mikä voi olla yleiset lohkot, muotoilu, asettelun elementit tai vimpaimet.

```
function register_block() {
    if (!function_exists('acf_register_block'))
        return;
    acf_register_block( array(
        'name'           => 'team-member',
        'title'          => 'Team Member',
        'render_template' => 'template-parts/layouts/block-team-member.php',
        'category'       => 'formatting',
        'icon'           => 'admin-users',
        'mode'           => 'preview',
        'keywords'       => array('profile', 'user', 'author')
    ));
}
add_action('acf/init', 'register_block');
```

KUVA 10. Uuden Gutenberg-lohkon luominen

Lisäkenttien rekisteröinti Gutenbergin sisällä käytettäväksi tapahtuu samalla tavalla kuin tavallisten kenttien rekisteröinti, mutta kenttää rekisteröitäessä sijaintiparametriksi asetetaan lohko eli "block". Valmiin kentän yhdistäminen Gutenberg-lohkoon tapahtuu vaihtamalla sijaintiparametrin uuden luodun lohkon nimen mukaan "acf/"-etuliitteellä (kuva 11).

```

'location' => array(
    array(
        array(
            'param'     => 'block',
            'operator' => '==',
            'value'     => 'acf/team-member',
        ),
    ),
),

```

KUVA 11. ACF-lisäkentän sijainnin asettaminen Gutenbergiin

Oman sivupohjan luominen yksittäiselle ACF-lohkolle mahdollistaa tälle määritetyn kentän tietojen esittämisen identtisenä Gutenberg-muokkaimessa ja sivustolla. ACF-lohkojen sivupohjat luodaan teeman hakemistoon. Sivupohjan sijainnin tulee täsmätä "acf_register_block"-funktiossa määritetyn "render_template"-parametrin kanssa. ACF-lohkon omassa sivupohjassa kutsutaan kentän tietoja kuten missä tahansa sivupohjassa "get_field()"-funktioilla.

7.2 Lohkojen näkyvyyden määrittäminen

Gutenbergin oletus- eli ydinlohkoja on kattava määrä ja kaikkia näitä ei todennäköisesti yhdellä sivustolla tarvita. Lisäämällä functions.php-tiedostoon uuden toiminnon, voidaan määrittää mitä lohkoja halutaan milläkin osaa sivustolla näyttää. Määrittääksemme, mitä lohkoja sivustolla näytetään, lisätään functions.php-tiedostoon suodatin "allowed_block_types", jonka sisällä voidaan ehtolauseella määrittää, missä mitäkin lohkoa halutaan näyttää (kuva 12). Kuvassa määritetään globaalisti näkyvät lohkot: kuva, tekstikappale, otsikko ja lista. Artikkelityypin ollessa "Sivu", näytetään myös lyhytkoodin lohko.

```

function allowed_block_types_pages($allowed_blocks, $post) {
    $allowed_blocks = array(
        'core/image',
        'core/paragraph',
        'core/heading',
        'core/list'
    );
    if ($post->post_type === 'page') {
        $allowed_blocks[] = 'core/shortcode';
    }
    return $allowed_blocks;
}
add_filter('allowed_block_types', 'allowed_block_types_pages', 10, 2);

```

KUVA 12. Gutenberg-lohkojen näkyvyyden määrittäminen functions.php-tiedostossa

Kun Gutenberg-muokkaimen sisään luodaan uusi lohko ACF:llä, täytyy se myös määrittää näkyviin "allowed_block_types"-suodattimen sisällä (kuva 13). Gutenbergin ydinlohkoista poiketen käytetään ACF-lohkojen näkyviin määrittämisessä etuliitteen "core" sijaan etuliitettä "acf". Määritettäessä lohkoa näkyviin käytetään lohkon nimenä "acf_register_block"-funktion sisällä nimiparametrin avulla määritettyä nimeä.

```

$allowed_blocks = array(
    'core/image',
    'core/paragraph',
    'core/heading',
    'core/list',
    'acf/team-member',
);

```

KUVA 13. ACF-lohkon näkyvyyden määrittäminen

7.3 Lohkojen tyylien luominen ja poistaminen

Uusia tyyliä Gutenberg-lohkoille voidaan luoda JavaScript-ohjelmointikielellä muun muassa painike- ja sitaattilohkoille. Tätä varten täytyy Gutenberg-muokkaimen lisätä oma JavaScript-tiedosto tyylien luomista varten functions.php-tiedostossa (kuva 14). Kuvan funktio hakee teeman juuresta tiedostoa register-styles.js, joten juureen täytyy lisätä tämän niminen JavaScript-tiedosto.

```
function add_gutenberg_scripts() {
    wp_enqueue_script(
        'gutenberg-scripts',
        get_template_directory_uri() . '/register-styles.js',
        array('wp-blocks')
    );
}
add_action('enqueue_block_editor_assets', 'add_gutenberg_scripts');
```

KUVA 14. JavaScript-tiedoston lisääminen Gutenberg-lohkotyylejä varten

Uuden tyylin luominen lohkolle tapahtuu tiedostossa register-styles.js kutsumalla ”registerBlockStyle”-funktioita, jossa määritetään tyylin kohde, eli lohko, ja tyylin nimi (kuva 15). Kuvassa alla luodaan painike-lohkolle liukuväriytyyli ”Gradient”.

```
wp.blocks.registerBlockStyle('core/button', {
    name: 'gradient',
    label: 'Gradient'
});
```

KUVA 15. Liukuväripainikkeen lisääminen tyylinä Gutenberg-muokkaimen

Luotu tyyli täytyy vielä tyylitellä CSS-tyylimäärittelyillä. Lisätään teeman tyylitiedostoon (style.css) liukuväripainikkeen tyylit. Lohkotyylin rekisteröinti lisää lohkolle automaattisesti lohkotyylin nimen mukaisen HTML-luokan ”is.style-”-etuliitteellä, jonka avulla lohkolle voidaan tehdä CSS-tyylimäärittelyjä (kuva 16). Tässä tapauksessa HTML-luokka on ”is.style-gradient”.

```
.wp-block-button.is-style-gradient .components-autocomplete > div,
.wp-block-button.is-style-gradient > a {
    background: linear-gradient(to bottom, #00528d 0%, #abdcd6 100%);
    padding: 15px 45px;
}
```



KUVA 16. Liukuväripainikkeen CSS-määrittelyt ja painike sivustolla

7.4 Tyylit Gutenbergiin

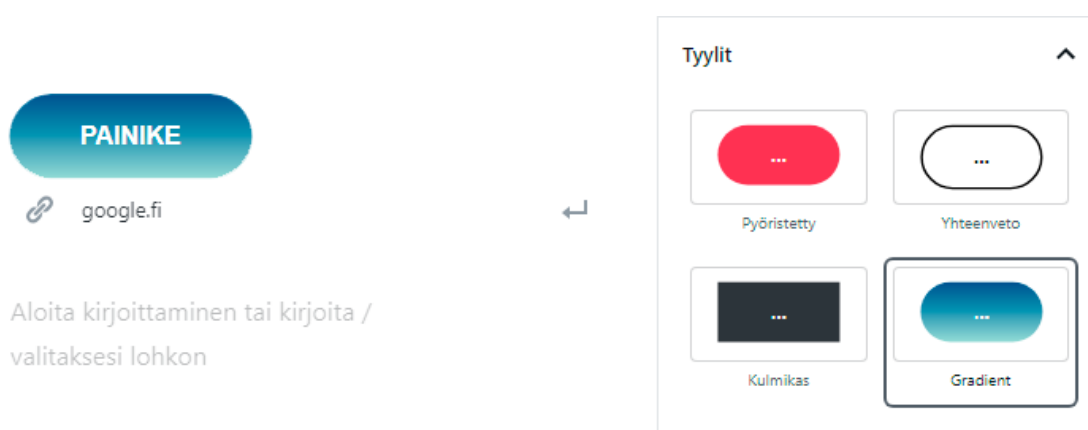
Sivuston tyylitiedostossa asetetut tyylit eivät näy Gutenberg-muokkaimessa. Gutenberg-tyylejä varten luodaan teeman juureen tyylitiedosto editor-styles.css ja

rekisteröidään tämä functions.php-tiedostossa latautumaan Gutenberg-muokkaimessa (kuva 17). Myös style.css-tiedosto voitaisiin rekisteröidä latautumaan Gutenberg-muokkaimessa, mutta kyseinen tiedosto sisältää usein ylimääräisiä tyylimäärittelyjä, joiden lataaminen ei ole tarpeellista ja nämä voivat sekoittaa muokkaimen oletustyyliä ja haitata muokkaimen käyttöä.

```
function block_editor_styles() {
    wp_enqueue_style(
        'editor-styles',
        get_template_directory_uri() . '/editor-styles.css',
        false,
        '1.0',
        'all'
    );
}
add_action('enqueue_block_editor_assets', 'block_editor_styles');
```

KUVA 17. Tyyli-tiedoston asettaminen Gutenberg-muokkaimen

Uuteen editor-styles.css-tiedostoon tulisi lisätä ACF-lohkojen ja Gutenberg-tyylien tyylimäärittelyt. Kun nämä tyylimäärittelyt asetetaan latautumaan myös Gutenberg-muokkaimen sisällä, näyttävät sen tyylit, lohkot ja niiden sisällöt samalta kuin sivustolla (kuva 18).



KUVA 18. Liukuväripainike Gutenberg-muokkaimessa

Jos asetettuja tyyliä halutaan poistaa, tulee register-styles.js-tiedostosta poistaa kyseisen tyylin rekisteröintifunktio. Gutenberg-lohkojen oletustyylien poistaminen tapahtuu tyylien luomisen lailla tiedostossa register-styles.js kutsumalla "unregisterBlockStyle"-funktiota. Tällä funktiolla voidaan poistaa käytöstä Gutenbergin oletustyyliä, kuten yksi sitaattielementin oletustyyleistä "Suuri" (kuva 19).

```
wp.blocks.unregisterBlockStyle('core/quote', 'large');
```

KUVA 19. Gutenberg-muokkaimen sitaattityyliin ”Suuri” poistaminen käytöstä

7.5 Uuden väripaletin luominen Gutenbergiin

Gutenbergin väripaletti sisältää oletuksena 11 eri väriä valittavaksi tiettyjen lohkojen teksti- tai taustaväriksi ja vapaan värivalintavaihtoehdon. Oletusväripaletin tilalle voidaan rekisteröidä uusi väripaletti functions.php-tiedostossa (kuva 20). Jos vapaa värivalinta halutaan poistaa käytöstä, voidaan väripaletin rekisteröintifunktioon lisätä ”add_theme_support(‘disable-custom-colors’);”.



KUVA 20. Väripaletin rekisteröinti ja värien asettaminen tekstikappaleelle Gutenbergissä

Tiettyillä Gutenberg-lohkoilla on valmiiksi asetettuja tyyliä, joiden avulla lohkoja voidaan tyyllitellä valmiiksi ilman CSS-tyylimäärittelyjä. Esimerkiksi Lainaus-lohko on valmiiksi kaksi eri tyyliä ”Tavallinen” ja ”Suuri” (kuva 21).

Lisää otsikko

Suuri lainaus

Sitaatti

Aloita kirjoittaminen tai kirjoita / valitaksesi lohkon

Asiakirja
Lohko
×

Lainaus

Korosta lainausta visuaalisesti.
 "Hyvä sitaatti on timantti
 mielevän miehen sormeen,
 mutta kivi tyhmän kouraan." —
 Joseph Roux

Tyylit

Suuri lainaus
Sitaatti

Tavallinen

*Suuri
lainaus*

Suuri

Edistyneet ominaisuudet

KUVA 21. Gutenberg-muokkaimen "Lainaus"-lohkon tyyli

8 POHDINTA

Opinnäytetyön tuloksena saatiin selvitettyä ja esitettyä, että WordPress itsessään tarjoaa hyvän pohjan Gutenberg-muokkaimen mukauttamiselle. Mukauttaminen kuitenkin vaatii ohjelmointiosaamista ja WordPressin perusteiden ymmärtämistä. WordPressin omilla toiminnoilla saadaan vaikutettua muokkaimen lohkojen näkyvyyteen, väripalettiin ja lohkotyyleihin.

Opinnäytetyöprosessin aikana saatiin perehdyttyä tapoihin laajentaa Gutenbergia ja selvitettiin optimaalisia tapoja toteuttaa tätä. Advanced Custom Fields -lisäosa tuo mukanaan hyödyllisiä ominaisuuksia Gutenberg-muokkaimen laajentamiseen. Lisäosan avulla voidaan rekisteröidä lisäkenttiä Gutenberg-muokkaimen sisään omiksi lohkoikseen. Näiden kenttien avulla saadaan toteutettua toistettavia mukautettuja sisältöalueita sivustolle, joita ei Gutenberg-muokkaimella itsellään voisi saada aikaan.

Gutenbergin laajentaminen on vasta alkutekijöissään ja toimintatavat tämän toteuttamiseen ovat vielä hieman raskaita. Opinnäytetyön raportissa esitetyllä tavalla saadaan kuitenkin mukautettua ja laajennettua muokkainta tehokkaasti. Uuden muokkaimen kehittyessä ja kehittäjien yhteistyöllä saadaan aikaan varmasti jatkossa myös tehokkaampia tapoja muokkaimen mukauttamiseen ja laajentamiseen.

LÄHTEET

Barker, D. 2016. Web Content Management. O'Reilly Media.

Laakso, T. 2018. Gutenberg mullistaa WordPressin – vai mullistaako?
<https://wp-palvelu.fi/blogi/gutenberg-mullistaa-wordpressin/>

MacDonald, M. 2014. WordPress: The Missing Manual, 2nd Edition. O'Reilly Media.

Wahalahti, T. 2018. WordPress 5.0 "Bebo".
<https://fi.wordpress.org/2018/12/06/wordpress-5-0-bebo/>