

OHJESIVUSTON TOTEUTTAMINEN SISÄLLÖNHALLINTAJÄRJESTELMÄLLÄ

Tiivistelmä

Tekijä(t) Tuominen, Emma	Julkaisun laji Opinnäytetyö, AMK Sivumäärä 29	Valmistumisaika Syksy 2020
Työn nimi Ohjesivuston toteuttaminen sisällönhallintajärjestelmällä		
Tutkinto Insinööri, tieto- ja viestintätekniikka		
Tiivistelmä <p>Työssä toteutettiin WordPress-pohjainen verkkosivusto LAB-ammattikorkeakoululle. Verkkosivuston tarkoitus oli korvata aiemmin käytössä ollut Wiki-pohjainen Peppi-opiskelutietojärjestelmän ohjekokoelma. Työn tavoitteena oli toteuttaa asiakkaalle mahdollisimman hyvin sisällöntuottajat ja sivuston ylläpitäjät huomioiva ratkaisu.</p> <p>Verkkosivun toteuttamiseen käytetty WordPress-sisällönhallintajärjestelmä valittiin sekä aiemman kokemuksen, että sen laajan muokattavuuden vuoksi. Lisäksi valintaan vaikutti WordPressin hyvä käytettävyys. Erityisesti WordPressin laajennettavuus ja hyvä käytettävyys osoittautuivat tärkeiksi ominaisuuksiksi verkkosivun toteutuksessa. Toteutetun verkkosivun sisällöntuottajat ja ylläpitäjät huomioitiin mahdollisimman yksinkertaisilla ratkaisulla ja selkeällä ohjeistolla, jossa käytiin ylläpitämiseen ja sisällöntuotantoon liittyvät ydintoiminnot läpi. Asiakkaan ohjeistamisella haluttiin taata verkkosivun päivittämisen sujuvuus ja parantaa sisällön yhdenmukaisuutta.</p> <p>Verkkosivun suunnittelussa huomioitiin käytettävyyteen liittyviä asioita, joiden avulla pyrittiin luomaan verkkosivun käyttäjäkokemuksesta toimiva ja miellyttävä. Käyttäjälähtöiseen suunnittelun teorioita hyödynnettiin sivuston suunnittelussa myös saavutettavuuden huomioimisessa. Erityisesti saavutettavuus ja käytettävyys pyrittiin huomioimaan selkeällä, rauhallisella käyttöliittymällä, sekä huolehtimalla kuvien seliteteksteistä ja riittävästä tekstin kontrastieroista.</p> <p>Työn lopputuloksena LAB Ammattikorkeakoulu sai käyttöönsä verkkosivun, jonka päivitettävyys ja käytettävyys on aiempaa Wiki-pohjaista sivua parempi ja joka huomioi myös saavutettavuuskysymykset paremmin.</p>		
Asiasanat käyttäjäkokemus, WordPress, käytettävyys, käyttöliittymä, sisällönhallintajärjestelmä		

Abstract

Author(s) Tuominen, Emma	Type of publication Bachelor's thesis	Published Autumn 2020
	Number of pages 29	
Title of publication Designing a website with content management system		
Name of Degree Bachelor of Engineering (Information technology)		
Abstract <p>The objective was to create a website for the instructions of the Peppi study information system. The study was commissioned by LAB University of Applied Sciences. The website was intended to replace the old Wiki-based guide. The goal was to create a solution that would take the administrators and the content producers into account as well as possible.</p> <p>WordPress was selected as the content management system based on the earlier experience with it. Other reasons for choosing WordPress for the project were WordPress' customization possibilities and its good usability. The content producers and administrators were taken into account by making the design solutions as clear and straightforward as possible. After the project they were given a guide where the key features and functions were instructed. The objective of the guide was to guarantee that the updating of the site would be easy and that the style of the content would stay consistent.</p> <p>User experience was an important part of the design process. The focus was in creating usable solutions that would create an enjoyable and functioning user experience for the users. The user interface was designed to be simple and calm. Alternate texts for images and the contrast of the text were designed with accessibility in mind.</p> <p>As a result of the study LAB University of Applied Sciences got a new website for sharing the instructions of the Peppi system. New website is more accessible and it has better usability compared to the old guide.</p>		
Keywords User Experience, WordPress, usability, user interface, content management system		

SISÄLLYS

1	JOHDANTO	1
2	KÄYTTÄJÄLÄHTÖINEN SUUNNITTELU.....	2
2.1	Käyttäjäkokemus	2
2.2	Käyttöliittymä	2
2.3	Käytettävyys	3
2.4	Käyttäjät	6
2.5	Saavutettavuus.....	7
3	SISÄLLÖNHALLINTAJÄRJESTELMÄT	12
3.1	Toiminta ja ominaisuudet	12
3.2	Avoin lähdekoodi	13
3.3	Arkkitehtuurityypit	14
3.4	Wordpress	15
3.5	Joomla.....	19
4	OHJESIVUSTON TOTEUTTAMINEN	21
4.1	Suunnittelu	21
4.2	Toteutus	22
4.3	Asiakkaan ohjeistaminen	23
5	YHTEENVETO	28
	LÄHTEET	29

1 JOHDANTO

Opinnäytetyön tavoitteena oli toteuttaa LAB-ammattikorkeakoululle verkkosivusto, jossa voitaisiin jakaa Peppi-opintojärjestelmään liittyviä ohjeita ja tärkeitä linkkejä aihealueeseen liittyen. Lisäksi työssä pyrittiin huomioimaan mahdollisimman hyvin myös asiakkaan sisälöntuottajien käyttäjäkokemus.

Työn teoriaosuudessa käydään läpi käyttäjälähtöistä suunnittelua sen eri osa-alueiden avulla. Saavutettavuuden osalta käsitellään eri asteisten aistillisten ja neurologisten haasteiden merkitystä verkkosivuston suunnittelussa ja miten näiden asioiden huomioiminen voi merkittävästi parantaa sivun tai palvelun käytettävyyttä. Lisäksi työssä käydään läpi käytettävyyteen vaikuttavia tekijöitä yleisesti, sekä käyttäjäliittymäsuunnittelun merkitystä käyttäjäkokemuksen kannalta.

Verkkosivuston toteuttamisen kannalta olennaisten sisällönhallintajärjestelmien toimintaperiaatteita selvitetään sekä yleisesti, että kahden esimerkkijärjestelmän kautta. Toisena esimerkkinä käytetään ohjesivuston toteuttamiseen valittua WordPressiä ja toisena WordPressin kilpailijaa Joomlaa. Sisällönhallintajärjestelmiin liittyen käydään läpi avoimen lähdekoodin merkitystä verkkosivujen toteuttamisen kannalta.

Työstä kertovassa osuudessa käydään läpi verkkosivuston toteuttamisen vaiheita. Läpi käytävät vaiheet ovat suunnittelu, toteutus ja asiakkaan perehdyttäminen. Vaiheiden osalta käydään läpi, miten asiakas huomioitiin prosessin edetessä ja millaisia ratkaisuja sivuston toteuttamiseen päätettiin käyttää.

2 KÄYTTÄJÄLÄHTÖINEN SUUNNITTELU

2.1 Käyttäjäkokemus

Termin käyttäjäkokemus määritelmä ei ole täysin vakiintunut, mutta sillä viitataan yleensä tuotteen tai palvelun loppukäyttäjän subjektiiviseen kokemukseen koko vuorovaikutuskaaresta. Käyttäjäkokemuksella viitataan siis käytännössä tunteeseen, joka käyttäjälle syntyy käytön aikana ja sen jälkeen. Tunteet voivat olla hyviä tai huonoja ja ne vaikuttavat käyttäjän mielialaan ja haluun palata palvelun tai tuotteen pariin. (All About UX.)

Käyttäjäkokemus ei ole teknologialähtöistä, vaan keskittyy ihmisiin (Roto ym. 2011, kirjoittajan suomennos).

Käyttäjäkokemus voidaan jaotella kahteen eri tekijään: käytännöllinen ja nautinnollinen. Käytännöllisiä tekijöitä ovat konkreettisemmat asiat kuten tuotteen tehokkuus ja käytettävyyttä. Nautinnollisiin tekijöihin voidaan lukea abstraktimmat ominaisuudet kuten miellyttävä design tai viihdearvo (Ux Guider, 2020). Hyvä käyttäjäkokemus muodostuu näiden tekijöiden yhteisvaikutuksesta ja puutteet kummassakin tahansa voivat vaikuttaa käyttäjän lopulliseen kokemukseen negatiivisesti. Harvoin kuitenkaan visuaalisesti kömpelö sivusto loistavalla käytettävyydellä jää käyttäjän mieleen erityisen hyvänä kokemuksena, vaikka siinä sivu tarkoituksensa täyttäisikin. Toisaalta pienet puutteet käytettävyydessä ja tehokkuudessa on joskus helpompaa antaa anteeksi, kun sivusto on ulkoasultaan käyttäjää jollain tavalla puhutteleva.

Käyttäjäkokemuksen arvioinnin apuna voidaan käyttää joitakin mittareita: onnistuiko käyttäjä suorittamaan haluamansa tehtävän ja miten kauan tehtävän suorittamiseen meni aikaa? Objektiiivisen arvion tekemiseksi käyttäjäkokemukselle täytyy siis määrittää mittareita, joille voidaan antaa jokin arvo, kuten aika minuutteina tai yksinkertainen kyllä tai ei-vastaus. Tällä tavalla käyttäjäkokemuksia voidaan mitata ja vertailla. Toisaalta loppukäyttäjän tyytyväisyys on tärkeimpiä mittareita, kun arvioidaan käyttäjäkokemuksen onnistuneisuutta. Pelkkä numeerinen arviointi voi olla haastavaa, eikä pelkästään siihen tukeutuminen välttämättä ole myöskään mielekästä, koska kokemuksen laatu on aina subjektiivinen asia. Kokemuksen subjektiivisuus taas tarkoittaa sitä, että käyttäjäkokemukset vaihtelevat yksilökohtaisesti eikä yhden yksilön kokemuksen perusteella siis varsinaisesti vielä voida määritellä tuotteen käytettävyyttä tai sen laatua. (Goodman ym. 2012, 16)

2.2 Käyttöliittymä

Käyttöliittymä ja käyttäjäkokemus ovat molemmat tärkeitä osia tuotetta tai palvelua suunniteltaessa. Käyttöliittymä on teknisempi ja objektiivisempi osa suunnittelua, kun taas

käyttäjäkokemus on abstraktimpi ja hyvin subjektiivinen. (Lamprecht 2020.) Käyttöliittymällä viitataan laitteen käyttöön tarkoitettuun tekniikkaan. Käyttöliittymän kautta käyttäjä voi esimerkiksi kirjoittaa tekstiviestejä, valita pyykinpesukoneen ohjelman tai maksaa verkko-ostoksensa. (Salakari 2016.) Käyttöliittymä on osatekijä käyttäjäkokemuksessa, mutta ei määrittele yksinään kokemuksen laatua. Käyttöliittymän tehtävänä on tehdä hahmottujen päämäärien saavuttaminen mahdolliseksi. Hyvä käyttöliittymä on yhdenmukainen, selkeä, helposti omaksuttavissa ja visuaalisesti miellyttävä.

Selkeys ja logiikka ovat tärkeitä ominaisuuksia käyttöliittymässä. Käyttäjän toimintaa ohjaavat aina aikaisemmat kokemukset ja tottumukset. Sen vuoksi käyttöliittymää suunniteltaessa ei ole välttämätöntä keksiä erikoisia tai ainutlaatuisia ratkaisuja – päinvastoin ne saattavat jopa tehdä käyttökokemuksesta hämmentävän. Kun käyttöliittymässä suositaan jo aiemmin toimiviksi havaittuja elementtejä, on käyttäjän helpompi ja nopeampi omaksua se. (Toivanen 2015.)

Käyttöliittymän yhdenmukaisuus auttaa käyttäjää käyttämään tuotetta tehokkaasti. Tehokkuus on merkittävä tekijä käyttäjäkokemuksen onnistumisessa. Yhdenmukaisuudella tarkoitetaan sitä, että kaikkialla käyttöliittymässä noudatetaan samoja periaatteita: visuaalinen muotoilu pysyy yhtenäisenä ja painikkeiden toiminta on johdonmukaista kaikkialla käyttöliittymässä (Toivanen 2015). Yhdenmukaisen käyttöliittymän avulla käyttäjä pystyy paremmin ennakoimaan käyttöliittymän toimintaa ja näin ollen käyttö on sujuvampaa ja miellyttävämpää verrattuna tuotteeseen, jonka käyttöliittymä ei ole yhdenmukainen.

2.3 Käytettävyys

Tuotteen tai palvelun käytettävyydellä viitataan siihen, miten hyvin loppukäyttäjä pystyy hyödyntämään tuotetta. Käytettävyyttä arvioidaan suhteuttamalla käytön onnistuminen resursseihin, joita tuotteen käyttö vaatii, kuten aikaan ja rahaan. (Ux Guider.) Käytettävyydelle ei kuitenkaan ole olemassa mitään absoluuttista määritelmää, jonka avulla olisi mahdollista aukottomasti määritellä, onko tuotteen käytettävyys hyvä vai huono (Jokela 2010, 19).

Käytettävyys on hyvän käyttäjäkokemuksen kulmakivi ja sen merkitys korostuu erityisesti tarkasteltaessa tuotteita tai palveluita, joiden käytettävyys on huono. Puutteellinen käytettävyys aiheuttaa käyttäjälle negatiivisia kokemuksia: halutun asian tekeminen ei onnistu, ei onnistu riittävän nopeasti tai aiheuttaa epämiellyttäviä tunteita. (Goodman ym. 2012, 26) Negatiivisia kokemuksia käytettävyydestä voi syntyä esimerkiksi käytettäessä järjestelmää, joka on vanhentunut ja rakennettu aikanaan kattamaan mahdollisimman laajasti toiminnallisuuksia. Tällaiset järjestelmät ovat usein kankeita ja moniportaisia, joka

hankaloittaa käytettävyyttä varsinkin moderneihin järjestelmiin tottuneille käyttäjille. Esimerkiksi taloushallinnon ja liiketoiminnan ohjaamiseen käytettävät toiminnanohjausjärjestelmät saattavat olla rakenteensa ja monimutkaisuutensa vuoksi haastavia käyttöönotettavia. (Topi ym. 2005.)

Käytettävyyttä arvioitaessa tulee aina peilata tuotteen suhdetta sen käyttäjään, käyttäjän tavoitteisiin ja käyttöympäristöön. Lisäksi arvioinnissa on mahdollista käyttää ISO 9241-11 standardissa määriteltyjä attribuutteja: tuloksellisuus, tehokkuus ja mielekkyys. (Kuutti 2003, 13.) Ohjelmistokehittäjä Wille Kuutin mukaan käytettävyys voidaan lisäksi jakaa viiteen yksityiskohtaisempaan osa-alueeseen, jotka pohjaavat samoihin arvoihin kuin standardissa:

- Opittavuus
- Muistettavuus
- Tehokkuus
- Pieni virhealttius
- Miellyttävyys

Opittavuudella viitataan käyttöliittymän helppouteen: miten nopeasti ja helposti käyttäjä sisäistää keskeiset toiminnot ja oppii käyttämään tuotetta sujuvasti. Opittavuuden merkitys korostuu, kun tuotetta ei käytetä säännöllisesti tai sen parissa ei ole tarkoitus viettää kerralla paljoa aikaa. Mikäli kyseessä on tuote, jota yksittäinen käyttäjä ei joudu käyttämään kuin kerran, tai vain osana isompaa prosessia, on erityisen tärkeää, että opittavuus on riittävällä tasolla. Käyttäjä ei halua esimerkiksi työhakemusta tehdessään joutua taistelemaan rekrytointiportaalin kiemuroiden kanssa, vaan prosessin tulisi olla selkeä ja suoraviivainen, jotta päästään helposti haluttuun lopputulokseen.

Osa-alueet sivuavat toisiaan ja erityisesti opittavuus ja muistettavuus ovat lähellä toisiaan. Muistettavuuden merkitys korostuu kuitenkin enemmän, kun tarkastellaan sellaisen tuotteen käytettävyyttä, joka on monimutkaisempi ja jota käytetään useammin. Sananmukaisesti muistettavuutta arvioidessa kiinnitetään huomiota siihen, miten hyvin tuotteen käyttö on omaksuttavissa. Muistettavien asioiden minimoimisella ja käyttöliittymän loogisuudella parannetaan prosessin sujuvuutta. Erityisesti järjestelmissä, joita käytetään usein, mutta jotka ovat monimutkaisia tai niillä tehdään pitkiä monivaiheisia prosesseja, on muistettavuus tärkeää.

Tehokkuudella tarkoitetaan haluttuun lopputulokseen pääsemiseen vaadittavien resursien määrää (Jokela 2010, 18.) Käytännössä tehokkuuden määrittelee siis aika, joka

prosessin suorittamiseen kuluu. Tehokkuuden tärkeys korostuu, kun käytettävyyks on muuten kunnossa, mutta joka kerta halutun asian tekeminen kestää suhteettoman kauan.

Pieni virhealttius sivuaa hieman ISO-standardissa määriteltyä tuloksellisuutta: mitä vähemmän virheitä käyttäjä voi tehdä, sitä todennäköisemmin hän pääsee haluttuun lopputulokseen. Virhealttiutta voidaan vähentää minimoimalla ylimääräiset ärsykkeet käyttöliittymässä ja tekemällä tuotteesta mahdollisimman helposti sisäistettävän. Usein jo vaihtoehtojen vähentäminen ja toimintojen automatisointi auttavat vähentämään virhealttiutta. Tuloksellisuutta tarkasteltaessa virheiden määrä ja laatu ovat olennaisia asioita. Mitä enemmän virheitä käyttäjä voi tehdä, sitä todennäköisemmin käyttäjä kokee olevansa umpiku-jassa ja saattaa jopa jättää prosessin kesken. Mikäli käyttäjä ajautuu umpikujaan virheiden takia, saattaa hän turhautua, eikä lopulta pääse lainkaan päämääräänsä. (Jokela 2010, 18.) Virhealttiutta voidaan vähentää, ja näin tuloksellisuutta parantaa, ohjeistamalla käyttäjää tuotteen tai palvelun käytössä. Hyvä käyttöohje paikkaa käyttöliittymän epäloogisuuksia ja auttaa omaksumaan myös monimutkaisemmat rakenteet nopeammin. Kaikissa tapauksissa käyttäjän ohjeistaminen ei välttämättä ole mielekäästä (kuten vaikkapa tavallisen verkkosivun kohdalla), mutta varsinkin esimerkiksi huonekaluun kokoamiseen ohje on miltei ehdoton ja toisaalta myös monimutkaisemman järjestelmän käyttöönotto voi sujua huomattavasti helpommin, jos järjestelmän käyttöön on olemassa kattavat ohjeet. (Tullis & Albert 2013, 6.)

Viidestä osa-alueesta miellyttävyyks on vaikein arvioida aukottomasti. Miellyttävyyks muodostuu aina käyttäjän omakohtaisesta kokemuksesta, eikä sitä sen vuoksi voi mitata objektiivisesti. Miellyttävyydellä kuvataan käytännössä usein käyttäjäkokemusta eli millaisia tunteita tuotteen käyttämisestä syntyy kokonaisvaltaisesti. (Jokela 2010, 18.). Miellyttävyyteen käyttäjästä riippuen liittyy erilaisia asioita, kuten värimaailma, muodot, latausnopeus ja käyttöympäristö. Värimaailmalla voidaan luoda eri käyttäjille erilaisia tunteita: henkilön kokemus jostakin väristä tai värien yhdistelmästä on subjektiivinen ja siihen vaikuttaa sekä henkilökohtaiset mieltymykset, että mahdolliset fyysiset tai psyykkiset rajoitteet. Fyysinen rajoite voi olla esimerkiksi värisokeus ja psyykkinen rajoite ahdistus- tai tarkkaavaisuus-häiriö, jotka kaikki liittyvät myös saavutettavuuteen. (Kuntoutussäätiö 2020) Väreillä voidaan kuitenkin pyrkiä luomaan haluttuja mielikuvia ja helpottamaan sivun tai palvelun käytettävyyttä ja saavutettavuutta. Värien avulla sisällön elävöittäminen on kohtuullisen vaivaton ja niiden avulla voidaan sekä korostaa, että häivyttää elementtejä. Käyttöliittymäsuunnittelussa väreillä voidaan korostaa käyttöliittymän logiikkaa ja ohjata käyttäjän toimintaa, mutta liikaa tai turhaa värien käyttöä tulee vältellä. (Mankki 2004.)

2.4 Käyttäjät

Käyttäjillä voi sivuston tarkoituksesta riippuen olla useita erilaisia rooleja, mutta lähtökohteisesti jokaisella sivulla on ainakin ylläpitäjiä ja loppukäyttäjiä. Tässä yhteydessä ylläpitäjillä viitataan lähinnä sivuston sisällöntuottajiin, sekä visuaalisesta ilmeen ja toiminnallisuuksien rakentamisesta vastaaviin henkilöihin.

Eri käyttäjillä on erilaisia tarpeita. Verkkosivua suunniteltaessa tulee miettiä, miten otetaan mahdollisimman suuren yleisön tarpeet huomioon, ilman että sivustosta tulee liian raskas tai monimutkainen. Sivuston tarkoituksen perusteella voidaan muodostaa käsitys siitä, millainen keskimääräinen loppukäyttäjä on: millaiset atk-aidot käyttäjällä on, miksi ja mihin käyttäjä tarvitsee sivustoa, sekä millainen kokemuspohja erilaisista käyttöliittymistä käyttäjällä on.

Sivuston luonteesta riippuen ylläpitopuolta käyttävien henkilöiden taustat voivat olla hyvin erilaiset. Useimmiten organisaatioissa sisältöä tuottavat henkilöt eivät ole välttämättä itse erityisen kokeneita verkkosivujen ylläpitäjiä ja monimutkaiset prosessit saattavat aiheuttaa hämmennystä. Loogisella käyttöliittymällä ja henkilöiden ohjeistamisella voidaan välttää useimmat ongelmatilanteet.

Eri sisällönhallintajärjestelmillä on erilaisia käyttäjäroolien hierarkioita ja käytössä olevat roolit riippuvat myös järjestelmän luonteesta. Kaikilla järjestelmillä on aina kuitenkin vähintään yksi käyttäjä, jolla on täysin rajattomat oikeudet myös palvelinpuolella. Mitä alemmas käyttäjäroolien hierarkiaa mennään, sitä riisutummat oikeudet rooliin kuuluu. Taulukossa 1 avattu sisällönhallintajärjestelmä WordPressin käyttäjäroolien hierarkia on kuvaava esimerkki tyypillisestä roolien portaisuudesta.

Rooli	Oikeudet
Super admin <i>Super-ylläpitäjä</i>	Käyttäjällä on rajaton pääsy tietoverkkopuolen ylläpito-ominaisuuksiin, sekä kaikkiin muihinkin toimintoihin
Administrator <i>Ylläpitäjä</i>	Käyttäjällä on rajaton pääsy kaikkiin ylläpito-toimintoihin
Editor <i>Toimittaja</i>	Käyttäjä voi julkaista ja hallita omia ja muiden käyttäjien julkaisuja
Author <i>Sisällöntuottaja</i>	Käyttäjä voi julkaista ja hallita omia julkaisujaan

Contributor <i>Kirjoittaja</i>	Käyttäjä voi kirjoittaa ja hallita omia julkaisujaan, mutta ei julkaista niitä
Subscriber <i>Tilaaaja</i>	Käyttäjä voi hallita ainoastaan omaa käyttäjäprofiiliaan

Taulukko 1. WordPressin käyttäjäroolien oikeudet ja hierarkia (WordPress)

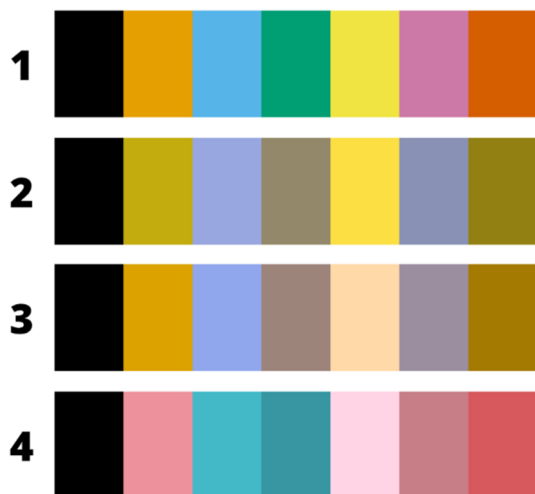
Myös käyttäjärooleilla on merkitys käytettävyydestä ja käyttäjäkokemuksesta puhuttaessa. Käyttäjälle voi olla hedelmällisempää, ettei hänellä ole järjestelmässä käytettävissä kuin juuri tarvittavat ominaisuudet – kun pelkästään artikkeleita sivulle tuottava henkilö ei edes pysty tekemään muuta, kuin mitä hänen on tarkoitus tehdä, eivät ylimääräiset toiminnallisuudet häiritse hänen työskentelyään, eikä hän voi tahallisesti tai tahattomasti tehdä vahinkoa esimerkiksi sivuston koodille tai ulkoasulle. Logiikan voi kääntää myös toisinpäin: kun käyttäjälle annetaan epähuomiossa liian vähän oikeuksia, eikä käyttäjällä ole ennestään riittävää tietotaitoa siitä, mitä hänen kuuluisi voida järjestelmässä tehdä, voi hän kokea varsin voimakasta turhautumista siitä, että halutun asian tekeminen ei onnistukaan.

2.5 Saavutettavuus

Ihmisten erilaiset rajoitteet voivat olla merkittävä haaste verkkosivua suunniteltaessa: kun sivuston käyttäjien lähtökohdat ovat erilaisia, on myös käyttökokemus erilainen. Esteettömyys ja saavutettavuus tähtäävät molemmat samaan tavoitteeseen. Paikkojen ja palvelujen tulisi olla kaikkien käytettävissä huolimatta siitä, millaisia rajoitteita henkilöllä mahdollisesti on. Esteettömyys viittaa konkreettisempiin asioihin kuten rakennuksiin ja liikennevälineisiin ja saavutettavuus erityisesti verkkosisältöön (Invalidiliitto). Saavutettavuudella tarkoitetaan erityisesti aistillisten rajoitteiden huomioimista ja se on tärkeä osa laajalle yleisölle tarkoitettua sivua. Saavutettavuuden huomioimisella sivun suunnittelussa ja toteutuksessa pidetään huoli siitä, ettei mitään käyttäjäryhmää rajata pois (Lawton Henry 2005).

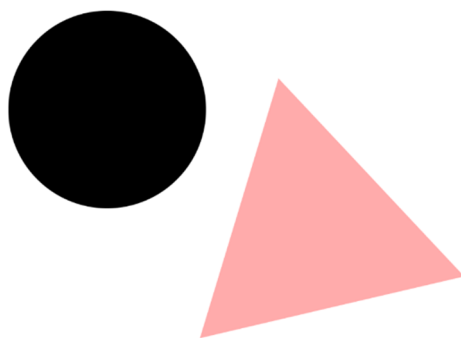
Saavutettavuutta voidaan parantaa erilaisilla keinoilla. Yleinen tapa on esimerkiksi huolehtia siitä, että kaikille visuaalisille elementeille (kuten kuville) on määritelty niin sanottu vaihtoehtoinen teksti, jossa kuvaillaan elementin sisältö tekstinä. Tämä auttaa käyttäjiä, joille kuvien hahmottaminen on haastavaa tai mahdotonta ja myös saavutettavuusdirektiivi edellyttää vaihtoehtoisten tekstien käyttämistä. Kuvat tunnistetaan tekstin seasta ruudunlukuohjelmalla. (Celia.) Saavutettavuuden huomioimisella voidaan tuoda internetsisältö paremmin käden ulottuville henkilöille, joilla on esimerkiksi puutteita kuulo- tai

näköaistissa, fyysisiä rajoitteita tai kognitiivisia haasteita. Haasteet voivat olla pysyviä tai hetkellisiä (Lawton Henry 2005).



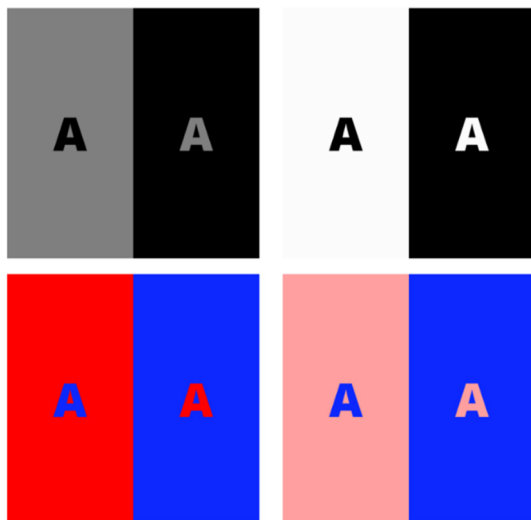
Kuva 1. Sama väripaletti eri värisokeuden muotojen näkökulmasta: 1. normaali, 2. trotonopia, 3. protonopia, 4. deutopia (Nichols)

Aktiivisuus- ja tarkkaavaisuushäiriö (ADHD) on melko yleinen kognitiivinen haaste, joka voi vaikuttaa selvästi yksilön kykyyn vastaanottaa myös sähköisen palvelun viestejä. Eri asteisista aktiivisuus- ja tarkkaavaisuushäiriöistä kärsii 4–7 % lapsista ja aikuisilla vastaava prosenttiluku on noin puolet. (Celia.) Erityisesti tarkkaavaisuushäiriön vaikutukset voivat olla käyttäjän kokemuksen kannalta ratkaisevia: mikäli sisältö on ylitsevuotavan visuaalista, värikästä ja runsasta, voi käyttäjän olla vaikeaa ohjata oma keskittymisensä oikeaan asiaan ja näin ollen saada jokin tavoiteltava asia tehtyä sivuston avulla. Samat haasteet voivat vaikuttaa myös ahdistuneisuudesta kärsivään käyttäjään: liika informaatio ja monimutkainen sisältö kuormittaa henkisesti ja saa aikaan epämiellyttäviä tunteita. Sivun sisältöä ja toiminnallisuuksia suunniteltaessa tulisi siis pitää mielessä, että siinä ei saisi olla mitään turhaa, vaan kaiken sisällön tulisi auttaa ja tukea käyttäjää saavuttamaan tavoitteensa. (Kuntoutussäätiö 2020.)



Kuva 2. Elementtien muotoeroja

Aistillisista haasteista puhuttaessa viitataan eri asteisiin näkö- kuulo- ja tuntoaistien rajoitteisiin. Tällaisia rajoitteita ovat muun muassa kuurous, värisokeus ja hermoston ongelmista johtuvat tuntohäiriöt. Toisin kuin näkörajoitteiselle, ei tavallisen verkkosivun, jolla ei juuri käytetä audiomateriaalia, käyttäminen yleensä ole hankalaa henkilölle, jonka kuuloaistissa on jonkinasteinen heikentyminen. Kuuloaistin rajoitteetkin pystytään kuitenkin huomioidmaan esimerkiksi videosisällössä tekstityksillä ja huolehtimalla siitä, että sivulla ei ole sellaista äänimateriaalia, jota ei sieltä löydy myös tekstimuodossa. Näköaistin rajoitteet asettavat enemmän vaatimuksia: sivuston rakenteen ja kuvien sisällön tulee olla luettavissa ruudunlukuohjelmassa, jonka avulla näkörajoitteinen henkilö pystyy käsittelemään verkkosivun tai -palvelun tarjoamaa informaatiota.



Kuva 3. Kontrastin vaikutus luettavuuteen, 1. rivi: keskiharmaa / musta, valkoinen / musta, 2. rivi: yhtä kirkkaat punainen ja sininen, haalistettu punainen / kirkas sininen

Toiminnallisuuksien ja sisällön saavutettavuuden lisäksi myös visuaalinen saavutettavuus on tärkeää. Visuaalinen saavutettavuus koskee erityisesti tekstin luettavuutta, johon voi vaikuttaa kirjasinvalinnoilla, kirjasimen värillä, koolla ja kontrastilla. Kontrastilla tarkoitetaan useimmiten värien kirkkauseroa (Kuva 3), mutta sillä voidaan viitata myös kokoeroon (Kuva 4) tai muotoeroon (Kuva 2). Riittävät kontrastit auttavat hahmottamaan kokonaisuuden ja erottamaan tekstin osat toisistaan (Rogriquez 2018). Kontrastien lisäksi saavutettavuutta, ja sen myötä käytettävyyttä, voidaan merkittävästi parantaa valitsemalla värejä, joiden yhdistelmät sopivat myös värisokeille (kuva 1). (Celia).

Tämä

On hyvin kontrastinen

Tässä

On heikompi kontrasti

Kuva 4. Kirjasimen koon ja leikkauksen vaikutus kontrastiin

Erilaisten käyttäjien huomioiminen ulkoasusuunnittelussa edellyttää erilaisia toimia ja ratkaisuja. Esimerkiksi käyttäjien, joilla on autismin kirjoon kuuluva Aspergerin oireyhtymä, huomioiminen edellyttää räikeiden värien välttelyä käyttöliittymäsuunnittelussa (Aluehallintovirasto). Jotkin käyttöliittymät tarjoavatkin nykyään käyttäjälle mahdollisuuden valita itse käyttöliittymän värimaailma, jolloin käyttäjä itse voi valita itselleen sopivimman värimaailman ja näin ollen parantaa järjestelmän käytettävyyttä itselleen. Tällä hetkellä usein valinta tehdään vaalean matalakontrastisemman ja tumman voimakaskontrastisemman värimaailman kesken (esimerkiksi Applen iOS-käyttöjärjestelmän normaali- ja tummatila), mutta jotkut järjestelmät, WordPress mukaan lukien, tarjoavat laajemman valikoiman (Apple; WordPress). Ääriesimerkki laajasta käyttöliittymän muokattavuudesta on Windowsin käyttöjärjestelmä, jossa käyttöliittymää voi mukauttaa aina kirjasimen ja valikoiden koosta värimaailmaan. (Heikkinen 2018)

3 SISÄLLÖNHALLINTAJÄRJESTELMÄT

3.1 Toiminta ja ominaisuudet

Sisällönhallintajärjestelmä (CMS, content management system) on järjestelmä, jonka avulla voidaan sananmukaisesti hallita ja julkaista sisältöä. Terminä sisällönhallintajärjestelmä tarkoittaa järjestelmää, jonka avulla voidaan hallita ja käsitellä useita erilaisia sisältötyyppejä aina tekstisisällöstä mediasisältöön. Termiä käytetään kuitenkin varsin laajasti ja sen merkityksen rajaaminen tiukasti koskemaan vain tietyt kriteerit täyttäviä järjestelmiä on hankalaa (Toivanen 2009). Erilaisia sisällönhallintajärjestelmätyyppejä ovat esimerkiksi dokumenttienhallintajärjestelmät, www-sisällönhallintajärjestelmät ja wikijärjestelmät. (Wikipedia).

Useimmiten termillä sisällönhallintajärjestelmä tai CMS kuitenkin viitataan julkaisujärjestelmään, jolla hallitaan ja julkaistaan sisältöä verkkoympäristöön, kuten WordPress tai Drupal (Tolvanen 2009). Kaikista verkkosivuista jopa 61 prosenttia käyttää jotakin sisällönhallintajärjestelmää, mikä kertoo järjestelmien käytettävyydestä suhteessa perinteiseen alusta loppuun manuaalisesti toteutettavaan (w3Techs). Sisällönhallintajärjestelmien suosio perustuukin sen helppouteen: käyttäjiltä ei vaadita yhtä suurta ymmärrystä eri ohjelmointikielistä ja kuvantamiskielistä, mutta lopputulos voi silti olla yhtä hyvä. Lisäksi sisällönhallintajärjestelmät mahdollistavat useamman käyttäjän työskentelyn ketterämmin kuin perinteisillä tavoilla toteutetut sivuprojektit.

Sisällönhallintajärjestelmien ydinominaisuuksiin kuuluu pelkän sisällön hallinnan ja julkaisemisen lisäksi näihin liittyviä muita toiminnallisuuksia, kuten metadatan hallinta, jolla voidaan vaikuttaa hakukonenäkyvyyteen ja osin myös saavutettavuuteen, ulkoasun luominen ja hallinta, versiointi, sekä käyttäjien ja käyttöoikeuksien hallinta. Sisällönhallintajärjestelmä kokoaa siis liki kaikki verkkosivuston ja sen sisällön ylläpitoon liittyvät toiminnallisuudet yhden järjestelmän alle, mikä parantaa tehokkuutta ja tuo ylläpito-ominaisuudet paremmin saataville. (Shivakumar 2016, 112.) Yrityksille sisällönhallintajärjestelmien edut voivat olla varsin merkittäviä: sisällön käsittely on tehokasta ja uuden sisällön julkaiseminen nopeaa ja joustavaa. Isojenkin sisältömassojen hallinta on mahdollista käyttövalmiiden kategoriointi- ja hallintatyökalujen avulla, eikä uuden sisällön julkaiseminen vaadi aina uuden sivupohjan luomista, vaan vanhojen pohjien hyödyntäminen on tehty helpoksi (Shivakumar 2016, 113).

Hyvistä puolista huolimatta sisällönhallintajärjestelmä aiheuttaa myös joitakin rajoitteita: ilman lähdekoodiin kajoamista käytössä on jonkun muun luoma käyttöliittymä, joka ei välttämättä suoraan toimi kaikille käyttäjille, vaikka se olisikin mahdollisimman suurelle

yleisölle suunniteltu. Tämän merkitys voi käyttäjästä riippuen olla joko marginaalinen tai merkittävä, mutta jonkun toisen suunnittelema järjestelmä harvoin täyttää käyttäjän kaikkia vaatimuksia, vaikka se sinänsä sopisikin tarkoitukseensa loistavasti. Valmis järjestelmä asettaa aina omat rajoituksensa ja valmiit raamit toiminnallisuuksille, mutta useat sisällönhallintajärjestelmät ovat onneksi melko laajasti kustomoitavissa kuhunkin projektiin sopivaksi. Kokonaiskuvaa tarkasteltaessa sisällönhallintajärjestelmien hyödyt ovat siis selvästi haittoja merkittävämpiä ja yleensä sisällönhallintajärjestelmän käyttö onkin varsin perusteltua, riippuen tietysti myös projektin luonteesta ja tavoitteista. (Shivakumar 2016, 115.)

3.2 Avoin lähdekoodi

Tavallisesti erilaiset ohjelmistot ja järjestelmät ovat käytettävissä tai niiden käyttöoikeus on ostettavissa, mutta käyttäjä ei voi tehdä niihin muutoksia kuin järjestelmän sallimissa rajoissa. Avoimen lähdekoodin ohjelmat ovat tässä suhteessa päinvastaisia: kuka tahansa voi muokata ohjelman koodia parantaakseen sitä tai tehdäkseen siitä itselleen sopivamman ja jopa levittää tätä paranneltua versiota eteenpäin. Avoimen lähdekoodilla viitataan siis sananmukaisesti ohjelmaan, jonka lähdekoodi on vapaasti saatavilla ja muokattavissa. (Weber 2004, 5.)

Ohjelmoijat selittävät asian usein yksinkertaisesti: kun kuulet vapaasti saatavilla olevasta ohjelmistosta, ajattele sananvapautta, älä ilmaista kaljaa (Weber 2004, 5, kirjoittajan suomennos).

Avoimen lähdekoodin tärkeimpinä elementteinä voidaan pitää seuraavia:

- Lähdekoodi täytyy jakaa ohjelmiston mukana tai sen täytyy olla jollain muulla tavalla saatavilla
- Kenen tahansa tulee voida jakaa ohjelmistoa eteenpäin ilman kustannuksia tai lisenssimaksuja
- Kenen tahansa tulee voida tehdä muutoksia ohjelmistoon ja jakaa samoin ehdoin siitä tehtyä versiota eteenpäin

Mikäli yksikin näistä kolmesta jää toteutumatta, ei voida varsinaisesti enää puhua avoimen lähdekoodin ohjelmistosta, vaan silloin on kyse joko tavallisesta kaupallisesta ohjelmistosta tai jostain näiden kahden väliltä olevasta ratkaisusta (Weber 2004, 5).

Sisällönhallintajärjestelmien lisäksi avoin lähdekoodi on varsin laajasti käytössä muissakin tietoteknisissä sovelluksissa: esimerkiksi tietokantajärjestelmä MySQL ja Apache, joka on käytetyin verkkopalvelin, perustuvat molemmat avoimeen lähdekoodiin. Voidaan siis puhua yksittäisten ohjelmien sijaan varsin laajasta ilmiöstä, jonka tuotteita käyttävät myös

suuret yhtiöt, kuten Disney tai Reuters. Vaikka yksityisten kuluttajien tietokoneella käyttöjärjestelmä on useimmiten Microsoftin Windows (markkinaosuus työpöytäkäytössä 2019 78 %) tai Applen macOS (markkinaosuus 16 %), löytyy yhä useamman ohjelman taustalta avoimen lähdekoodin järjestelmä. (Wikipedia; Weber 2004, 6.)

Avoimen lähdekoodin yksi lippulaivoista on suomalaisen Linus Torvaldsin Unix-käyttöjärjestelmästä jalostama Linux. Linux sai alkunsa 90-luvulla Torvaldsin turhaututtua Microsoftin DOSiin ja siirryttyä Unixin yksinkertaistettuun versioon Minixiin, hän alkoi kehittää sen pohjalta omaa käyttöjärjestelmäänsä. Torvaldsin Linux otti nopeasti tuulta alleen, mutta kohtasi myös kritiikkiä. Ensimmäistä Linuxin versiota julkaistessaan Torvalds kuvaili sen käytettävyyden olevan kohtuullisella tasolla riippuen siitä, millaisia vaatimuksia käyttäjillä on. Tämä on olennaista myös nykypäivänä käytettävyyden ja käyttökokemuksen kannalta: lopullinen käytettävyyden taso riippuu aina myös yksittäisen käyttäjän omista vaatimuksista ja odotuksista. Vuosikymmenen lopulla Linux oli saavuttanut jo odotettua suuremman suosion ja avoimen lähdekoodinsa ansiosta myös muut ohjelmoijat pääsivät osallistumaan sen kehittämiseen. (Weber 2004, 55.)

3.3 Arkkitehtuurityypit

Sisällönhallintajärjestelmien järjestelmäarkkitehtuuri voidaan jakaa useampaan tyyppiin:

- Coupled (kytketty)
- Decoupled (irtikytkenä)
- Headless (päätteetön)
- Hybridi

Coupled on perinteisesti sisällönhallintajärjestelmissä käytetty arkkitehtuuri. Se sananmukaisesti kytkee backendin ja frontendin toisiinsa. Käytännössä tämä tarkoittaa sitä, että sekä järjestelmän toiminnallisuudet, tietokanta ja muokkaustyökalut, että itse sisällön esittäminen laitteilla käyttäjille, on koottu samaan järjestelmään. Tämä mahdollistaa sen, että kaiken voi tehdä saman järjestelmän sisällä, eikä backendin ja frontendin välille tarvitse erikseen rakentaa yhteyttä. Toisaalta tämä rajoittaa käytettävissä olevia sisältötyyppejä, eikä tarjoa yhtä laajoja ohjelmointimahdollisuuksia kuin muut arkkitehtuurityypit. Esimerkiksi WordPress ja Joomla! käyttävät perinteistä kytkettyä arkkitehtuurityyppiä. (Brightspot.)

Decoupled-arkkitehtuuri erottelee backendin ja frontendin toisistaan ja yhteys näiden kahden välillä luodaan erilaisten API:n avulla. Decoupledissa verkkosivun hallinta siis hajautetaan kahteen järjestelmään, joista toinen vastaa tietokannoista ja sisällönhallinnasta ja toinen sisällön esittämisestä käyttäjälle. Frontendin hoitava järjestelmä hakee API:n ja

erilaisten rajapintaratkaisujen avulla sisältöä backendistä. Vaikka järjestelmät ovatkin itsenäisiä, ne ovat silti osa yhtä kokonaisuutta. Decoupled-arkkitehtuurin käyttökohteita voivat olla esimerkiksi mobiilisovellukset ja puettava teknologia, kuten älykellot. (Brightspot.)

Headless-arkkitehtuuria käyttävän sisällönhallintajärjestelmän käyttökohteet ovat samantaisia, kuin decoupled-arkkitehtuurin ja sitä voidaanakin pitää irtikytkennän alatyypinä. Molemmat käyttävät APIa tai erillistä verkkopalvelua tiedon välittämiseen, mutta päätteettömässä arkkitehtuurissa perinteinen frontend puuttuu. Backendin tieto ei kuitenkaan samalla tavalla mene määrättyyn käyttöliittymäkerrokseen, vaan sitä voidaan välittää useampaan eri paikkaan. (Brightspot.)

Hybridi sananmukaisesti viittaa järjestelmään, jossa on yhdistelty eri arkkitehtuurityyppejä. Lopullisen sivuston osia voidaan hakea eri järjestelmistä eri tavoilla. Esimerkiksi backendissä pyörii itse sisällönhallinta ja sieltä sieltä haetaan APIlla ja eri rajapintaratkaisuilla sisältöä. Samaan aikaan ulkoasun design voi tulla suoraan sisällönhallintajärjestelmästä frontendiin. Hybridi on samantyyppinen, kuin päätteeton, mutta hybridi mahdollistaa backendin ja frontendin välisen tiedonkulun laajemman muokattavuuden, sekä parantaa jo julkaistun sisällön käsittelymahdollisuuksia. (Winkels 2019.)

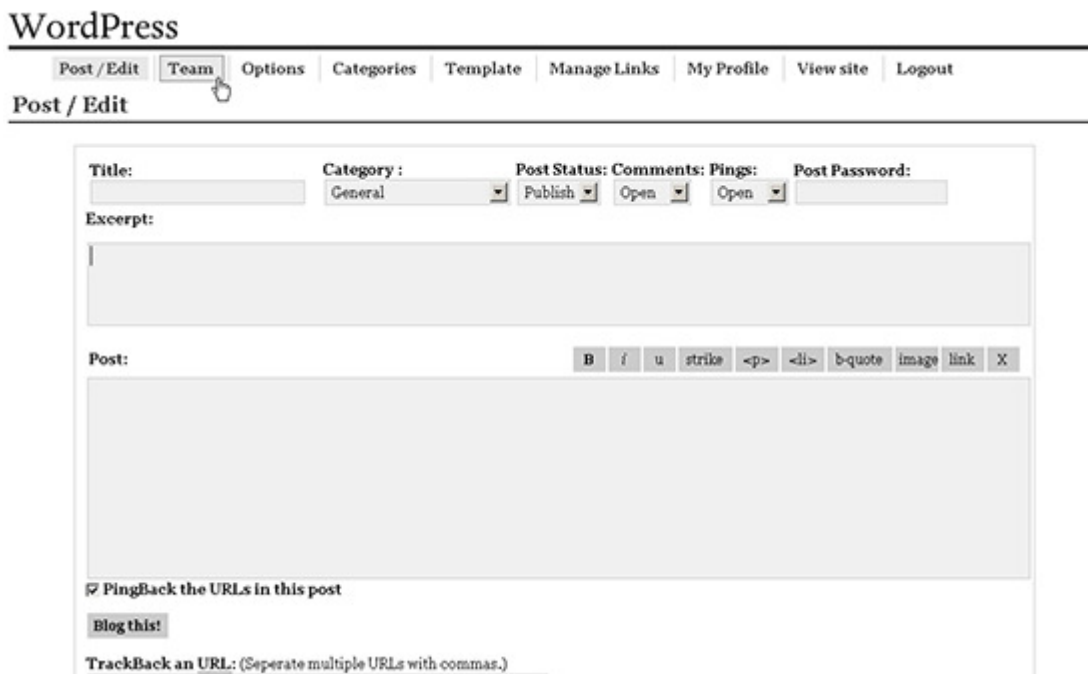
Eri arkkitehtuurityypeillä on erilaiset vahvuudet ja käyttötarkoituksensa. Perinteinen kytetty arkkitehtuuri tuo kaikki sisällönhallintajärjestelmän puolet samaan osoitteeseen, mikä voi olla käytettävyyden kannalta hyvä, mutta toisaalta se samaan aikaan rajoittaa mahdollisuuksia ja tekee kokonaisuudesta haavoittumaisemman. Irtikytkentäarkkitehtuurin, tai sen päätteettömän alatyypin, mahdollisuuden ovat ovat laajemmat, mutta niiden käyttäminen edellyttää enemmän ohjelmistopuolen osaamista ja valmiita ratkaisuja on tarjolla vähemmän. Kaikille löytyy kuitenkin omat käyttökohteensa ja on projektin laajuudesta ja tavoitteista kiinni, mikä arkkitehtuurityyppi on sille sopivin. (Brightspot; Winkels 2019.)

3.4 Wordpress

WordPress on PHP- ja MySQL-pohjainen sisällönhallintajärjestelmä, jonka kehitystyö on alkanut vuonna 2001. (WordPress). Se sai alkunsa, kun Matt Mullenweg alkoi kehittämään omaa versiotaan Cafelog blogialustasta yhdessä Mike Littlen kanssa. Muutama vuosi myöhemmin WordPressin ensimmäinen versio julkaistiin ensisijaisena käyttötarkoituksenaan blogien julkaiseminen. (Kinsta.) Avoimeen lähdekoodiin pohjautuvalla WordPressillä voi nykyään tehdä huomattavasti toiminnallisempia ja monipuolisempia verkkosivustoja kuin sen alkuvuosina (Namecheap).

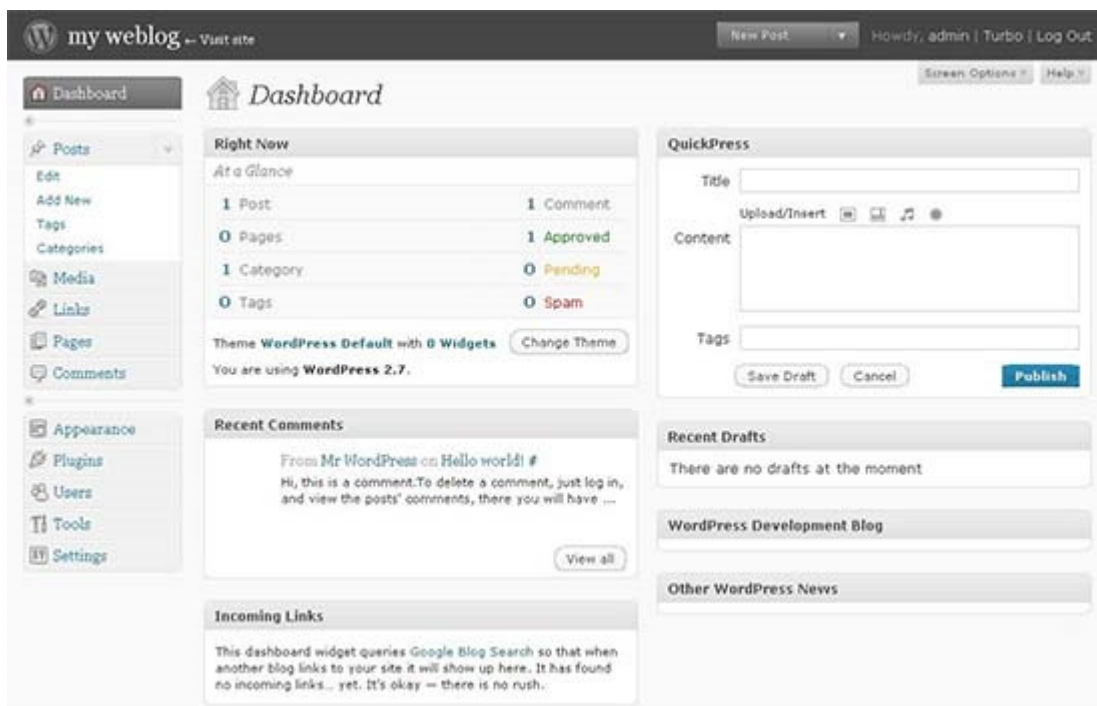
Verkkosivuille ja palveluille tyypillisesti WordPressin käyttöliittymä on muuttunut uusien versioiden myötä. Ensimmäiset käyttöliittymät olivat riisuttuja, eikä myöskään

ominaisuuksia ollut niin paljoa, kuin tuoreemmissa versioissa. Esimerkiksi kuvassa 5 näkyvä versio 0.71 vuodelta 2003 on koruton, mustavalkoinen ja toiminnot ovat selvästi rajoitumpia kuin nykyisessä versiossa. Varhaisimmissa versioissa ominaisuudet ja toiminnot oli tarkoitettu käytännössä ainoastaan blogien julkaisemiseen.



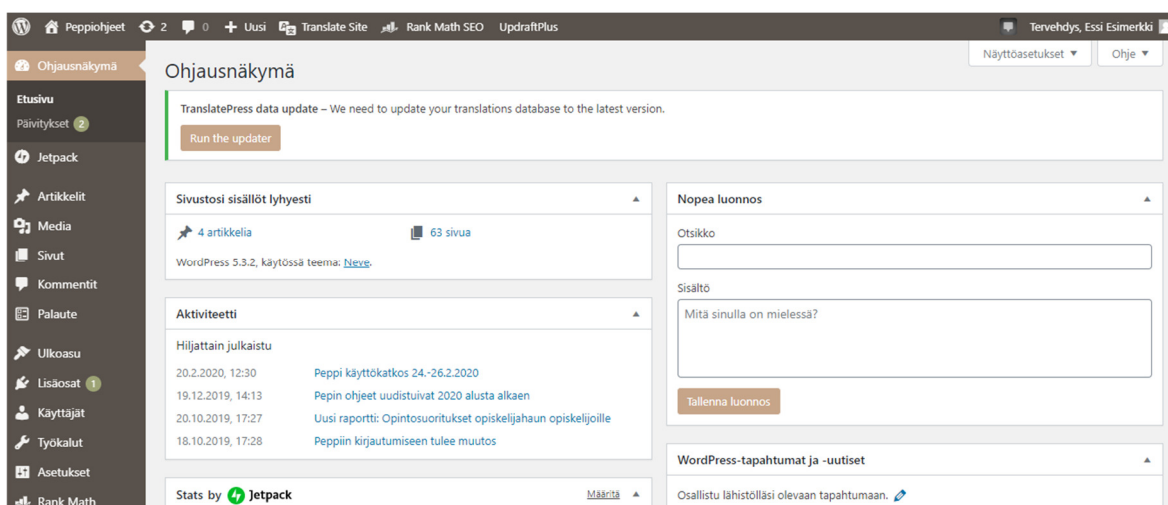
Kuva 5. Versio 0.71, 2003 (WPBeginner)

WordPressille ominaiset lisäosat esiteltiin ensimmäisen kerran vuoden 2004 versiossa 1.2 "Mingus". Samassa versiossa lisättiin myös muita ominaisuuksia, kuten julkaisun esikatselu. Käyttöliittymä oli moderniin versioon verrattuna yksinkertainen vuoteen 2008 asti, jolloin julkaistiin versio 2.7 "Coltrane". Kuvassa 6 olevan Coltranen käyttöliittymä muistutti jo enemmän 2010-luvun versioita: ohjauspaneelin alitusnäkyvässä on yhteenveto uusimmista kommentteista ja pieni tilasto artikkeleiden vaiheista, myös sivuvalikon rakenne on käytännössä sama. Uusien versioiden mukana myös ominaisuudet päivittyivät. (WPBeginner.) Minguksen ja Coltranen jälkeen Wordpressin kehitys on jatkunut, mutta peruselementit ovat pysyneet samankaltaisina.



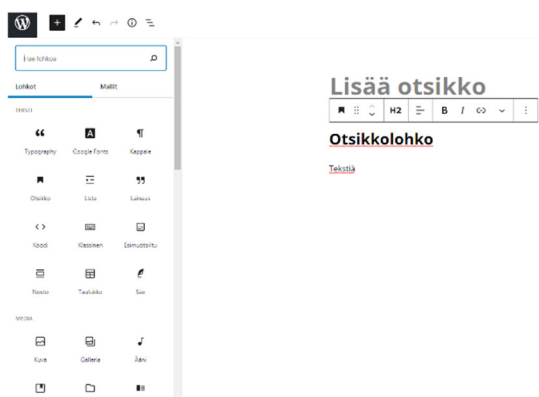
Kuva 6. Versio 2.7 Coltrane (WPBeginner)

Myös kuvan 7 tuoreen Wordpress version käyttöliittymän pohjana on ohjausnäkö, jonka kautta sivuston ulkoasua, toimintaa, käyttäjiä ja sisältöä hallitaan. Ohjausnäkömää noudattaa melko tavanomaista käyttöliittymän asettelua: päävalikko vasemmalla reunassa, ylävalikossa koottuna muutama toiminnallisuus, sekä pääsy oman käyttäjän hallintaan, keskelle aukeaa sivujen sisällöt ja muokkausnäkömät. Tällaisen asetteluun hyviä puolia ovat sen loogisuus ja selkeys. Asettelu on myös tuttu monille käyttäjille muiden järjestelmien käyttöliittymistä, joten useimmille Wordpressin käyttöliittymän omaksuminen on kohtuullisen helppoa.



Kuva 7. Versio 5.3.2 ohjausnäkö (WordPress)

WordPressissä on oletuksena sisäänrakennettu kuvan 8 Gutenberg-lohkoeditori, jolla sivujen ja artikkelien sisältö rakennetaan. WordPressin sivut ja artikkelit muodostuvat html muotoillusta tekstistä ja tästä johtuen sisältö on varsin vapaasti hyödynnettävissä. Gutenberg luo lohkojen avulla strukturoituja html-elementtejä, pitäen samalla huolen siitä, ettei editorin toimintaperiaate suoraan näy itse sivulla. Lohkojen avulla myös käyttäjäkokeumus säilyy johdonmukaisena, sillä eri lohkot käyttäytyvät keskenään samalla tavalla, vaikka niiden luomat elementit ja toiminnot ovatkin keskenään erilaisia. (WordPress.)



Kuva 8. Gutenberg lohkoeditori (WordPress)

Olennainen osa WordPressiä ovat sen käytettävissä olevat lisäosat, joita käyttäjät ja yritykset tuottavat ja tarjoavat WordPress käyttäjien käyttöön usein ilmaiseksi. Lisäosilla WordPressin ominaisuuksia ja toiminnallisuuksia voidaan lisätä ilman, että käyttäjän itse täytyy ohjelmoida niitä. Esimerkkejä yleisesti käytetyistä lisäosista ovat erilaiset varmuuskopiointilisäosat ja lomakkeiden luomiseen käytettävät lisäosat. Lisäksi lisäosien avulla WordPressiin voidaan esimerkiksi asentaa uusi sivunrakentaja (page builder), jolla voidaan korvata WordPressin oma lohkoihin perustuva sivunrakentaja. (WordPress.)

Blogialustana tunnetun Wordpressin suosio on ollut tasaista, mutta merkittävää kasvua ei ole enää viime vuosina tapahtunut, toisin kuin WordPressin kilpailijoilla, joiden kasvu on ollut selkeästi nousujohteista. Vuonna 2020 W3 Techs:in toteuttaman verkkoteknologiakyselyn perusteella WordPress kipusi kattamaan jo 39 prosenttia kaikista verkkosivuista ja oli edelleen käytetyin sisällönhallintajärjestelmä kattaen sisällönhallintajärjestelmää käyttävistä verkkosivuista 64 prosenttia. WordPress jätti kilpailijansa pitkälle taakseen: seuraavaksi suosituimpien Shopify:n 5 ja Joomla:n 4 prosenttia ovat vain murto-osa WordPressin osuuteen verrattuna. (w3techs.)

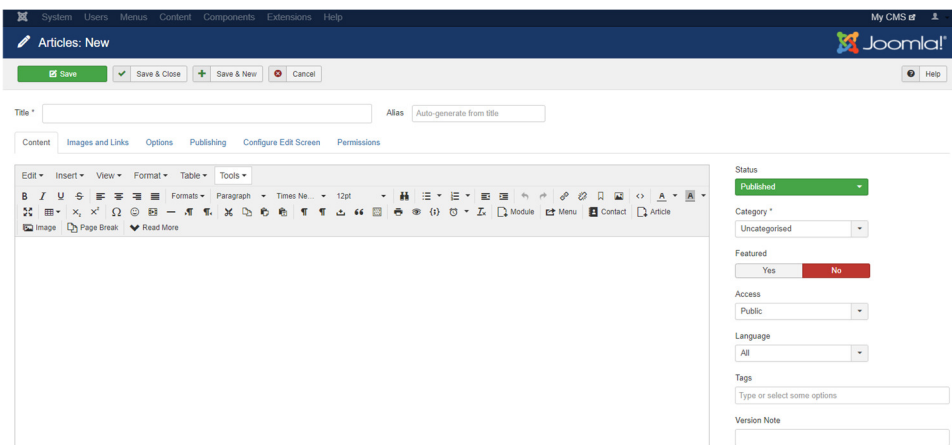
3.5 Joomla

Joomla on Open Source Matters-tiimin luoma Wordpressin tavoin avoimeen lähdekoodiin perustuva sisällönhallintajärjestelmä (Open Source Matters). W3 Techsin kyselytulosten perusteella Joomla on kolmanneksi käytetyin sisällönhallintajärjestelmä ja sen ominaisuudet ja toimintaperiaatteet ovat melko samankaltaiset kuin kilpailijallaan. WordPress on pitkään jatkuneen suosionsa myötä laajemmin muokattavissa erilaisilla lisäosilla kuin Joomla, mutta suosion vaikutus näkyy myös Wordpressin haavoittuvuudessa ja näin ollen vähemmän käytetty Joomla! peittoaa tällä hetkellä Wordpressin tietoturvan osalta. (Prohaska 2020.)

Joomla on saanut alkunsa 2000-luvun alussa, kun australialainen Miro kehitti Mambo-nimellä kulkeneen sisällönhallintajärjestelmän. Myöhemmin Mambo julkaistiin maksuttomana avoimen lähdekoodin lisenssillä ja sen myötä kerrytti myös suosiotaan. Mambon kehitys ei ehtinyt jatkua pitkään ennenkuin tekijänoikeusselkkausten jälkeen iso osa Mambon kehittäjistä jätti sen ja he lähtivät kehittämään Mamboa omaan suuntaansa nimellä Joomla. Ensimmäinen versio Joomlaista julkaistiin vuonna 2005. (Javatpoint.)

Seuraava merkittävä versio 1.5 julkaistiin kolme vuotta ensimmäisen version jälkeen. Tämän version myötä Joomlaan ominaisuudet paranivat selvästi: se tarjosi paremmat mahdollisuudet ulkoasun räätälöimiseen ja mahdollisti myös kansainvälisemmän käyttäjäkunnan oikealta-vasemmalle luettavien kielten tuen avulla. Lisäksi järjestelmän käytettävyys parani merkittävästi. (Pignataro.)

Joomla on ominaisuuksiltaan samankaltainen kuin WordPress, se mahdollistaa sivujen rakentamisen ja julkaisemisen ja kuten Wordpressiin, myös Joomlaan voidaan asentaa erilaisia teemoja ja lisäosia (Prohaska 2020). Joomlaan heikkoudet johtuvat lähinnä sen vähäisemmästä suosiosta Wordpressiin verrattuna: Wordpressin käytön osaavia verkkosivujen suunnittelijoita ja toteuttajia löytyy enemmän ja myös tukea tarjoava yhteisö on suurempi. Lisäksi Joomlaan hyödyntäminen blogikäytössä ei ole yhtä suoraviivaista kuin Wordpressissä, sillä Joomlaan ei suoraan ole käytössä tavallisia staattisia sivuja ja artikkelisivuja, kuten Wordpressissä, vaan ne täytyy luoda kategorioiden avulla. Toisaalta Joomlaan käyttöliittymä kuvassa 9 on itsessään Wordpressiä suoraviivaisempi ja se antaa vielä enemmän räätälöintimahdollisuuksia niin käyttäjäroolien, kuin teemojenkin suhteen. (Jackson 2020.)



Kuva 9. Joomlaan tekstieditori (Joomla)

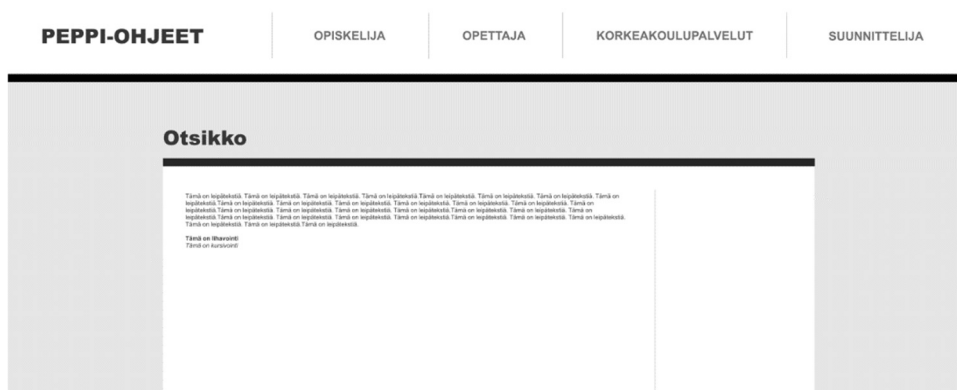
4 OHJESIVUSTON TOTEUTTAMINEN

4.1 Suunnittelu

Työn tavoitteena oli luoda uudelle LAB-ammattikorkeakoululle sivusto, jossa voitaisiin jakaa Peppi-opiskelutietojärjestelmän käyttöohjeita sekä henkilöstölle, että opiskelijoille. Peppi on ammattikorkeakoululla käytössä oleva toiminnanohjausjärjestelmä, jolla hallinnoidaan opiskelijoiden tietoja, opintototeutuksia ja muita opetukseen ja opiskeluun liittyviä kokonaisuuksia. Työ toteutettiin tiiviissä yhteistyössä opintohallinnon edustajan kanssa. Sivun tavoitteeksi asetettiin selkeä, uuden ammattikorkeakoulun ilmeen mukainen sivusto, josta löytyisi helposti tarvittava tieto, ilman turhia toiminnallisuuksia. Sisällönhallintajärjestelmästä oli tehty ennakkopäätös jo ennen projektin varsinaista aloitusta, mutta projektin alussa WordPressin valinta varmistui lopullisesti. WordPress valikoitui alustaksi erityisesti kahdesta syystä: asiakkaalla oli hyviä kokemuksia WordPressin käytöstä jo ennestään ja WordPressin ominaisuudet olivat sopivia tällaisen verkkosivuston toteuttamiseen.

Sivun ulkoasua ja sisältöä suunniteltiin tavoitteiden perusteella. Ulkoasun värimaailma päätettiin ottaa miltei suoraan Peppi-järjestelmän omasta värimaailmasta ja korostusväreiksi valittiin LAB-korkeakoulun graafisen ohjeistuksen mukaisia värejä. Värimaailma muodostui siis mustan ja valkoisen lisäksi vaaleasta harmaasta. Linkkien ja muiden nostojen korostamiseen käytettiin mintunvihreää korkeakoulun graafisen ohjeistuksen mukaisesti. Muita värejä käytettiin ainoastaan eri kuvakkeissa Pepin värimaailman mukaisesti.

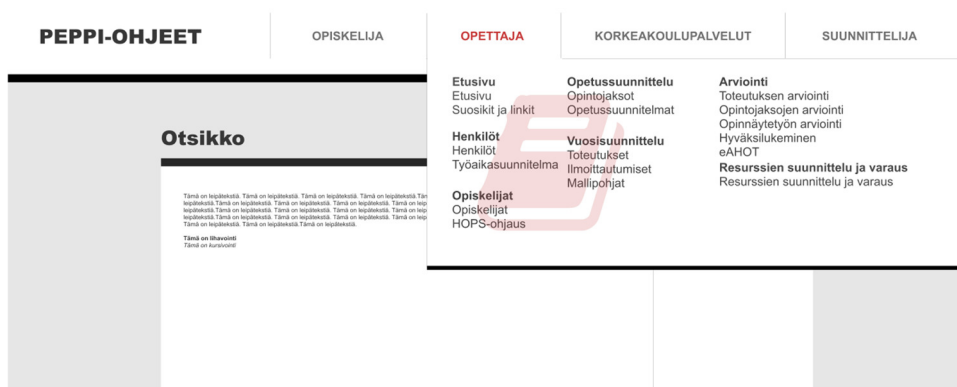
Ulkoasusta toteutettiin prototyyppi Adobe XD-ohjelmalla. XD:llä toteutettiin näkymä perussivusta, johon sijoitettiin täytetekstiä, sekä sivulle suunnitellut muut kiinteät elementit. XD:n käytön etuja olivat erityisesti valikkorakenteen ja valikon logiikan ja toimintojen testaaminen etukäteen. XD:n avulla myös sivuston ulkoasun konseptia pystyttiin demonstroimaan asiakkaalle tehokkaasti.



Kuva 10. Ulkoasun prototyyppi

Sivurakennetta suunniteltiin sisällyluetteloiden avulla. Asiakkaan kanssa selvitettiin, mitkä sivut vanhasta wikipohjaisesta sivustosta säilytettäisiin ja miten sivujen rakenne muuttuisi uudelle sivulle siirrettäessä. Sivusta koottiin aihekokonaisuuksia, joiden avulla rakenne hahmottui paremmin. Aihekokonaisuudet määrittivät myös sivujen hierarkkisuuden ja ne otettiin myös valikkorakenteeseen mukaan.

Valikon rakenne suunniteltiin ensin paperille ja kun sen ajatus ja ryhmittely oli selkeä, sijoitettiin valikko myös prototyypiin. Prototyypin toteutettiin valikon toiminnallisuudet, joiden avulla sen logiikka pystyttiin tarkastelemaan ja tarvittavat muutokset rakenteeseen voitiin tehdä jo ennen itse sivustolle vientiä. Prototyypissä demonstroitiin, miten valikko käyttäytyy, kun valikon osaa klikataan sen ollessa auki ja miten se käyttäytyy, kun toisen alavalikon ollessa auki, klikataankin toista ylävalikon osaa.



Kuva 11. Valikon toiminnan demonstroiminen prototyypillä

4.2 Toteutus

Sivusto toteutettiin asiakkaan palvelimella sijaitsevaan WordPress-asennukseen syksyllä 2019. Suunnitelman pohjalta sivulle valittiin sopiva teema, joka antoi puitteet sivuston rakentamiselle. Toiminnallisuuksien toteuttamiseen vaadittavista lisäosista tehtiin vasta tässä vaiheessa päätöksiä, kun niiden soveltuvuutta pystyttiin ensin testaamaan reaaliaikaisesti sivustolla. Esimerkiksi valikon rakentamiseen käytetty lisäosa valikoitui muutaman vaihtoehdon joukosta testien perusteella, kun se osoittautui vaihtoehtoista sopivimmaksi ja mahdollisesti valikon toiminnan ja ulkonäön muokkaamisen riittävällä tasolla.

Käyttöoikeudet jaettiin projektin alussa vain henkilöille, joiden oli tarkoituksenmukaista joko osallistua sivuston tekemiseen tai tarkkailla sen edistymistä. Käytännössä tämä tarkoitti sitä, että käyttöoikeus oli aluksi pääkäyttäjällä eli asiakkaan tietohallinnon edustajalla, sivuston toteuttajalla ja projektiin osallistuvilla asiakkaan edustajilla. Myöhemmin käyttöoikeuksia voitiin laajentaa kaikille, joiden toivottiin osallistuvan sivun sisällöntuotantoon. Täysiä ylläpito-oikeuksia jaettiin edelleen vain välttämättömille käyttäjille virheiden

välttämiseksi. Lopuille jaettiin yhtä tasoa alemmat oikeudet, jotka mahdollistivat pääsyn sisältöön, mutta ei esimerkiksi ulkoasun tai sivuston koodin muuttamista.

Käyttöön otetut käyttäjäroolit:

- Admin
- Editor

Asiakas tuotti kaiken sisällön sivustolle itse ja osittain käytettiin vanhaa olemassa olevaa sisältöä. Sisällöt käytiin projektin aikana yhdessä läpi ja niiden kirjoitusasuja ja rakennetta yhtenäistettiin, jotta kokonaisuudesta saatiin mahdollisimman yhdenmukainen ja looginen. Viimeistely sisällölle tehtiin, kun se oli siirretty WordPressiin ja nähtiin, miten se asettuu sivupohjille. Tekstille määriteltiin tietyt perustyyli, joita eri sivuilla käytettiin johdonmukaisesti. Esimerkiksi otsikkotyyli määriteltiin kolmeen eri tasoon, jotta sisällöntuottajien olisi helppo valita tarkoituksenmukainen tyyli. Kolme eri tasoa olivat pääotsikko H1, alemman tason otsikko H3 ja alaotsikko H4. Lisäksi asiakasta ohjeistettiin käyttämään otsikkotyyliä H5, kun otsikkoa ei haluttu nostaa sisällysluetteloon.

Sivusto toteutettiin osittain kaksikielisenä ja kääntämiseen käytettiin TranslatePress-lisäosaa, jolla oli toteutettu jo aiemmin käännöksiä LABin edeltäjän, Lahden Ammattikorkeakoulun, Wordpress-pohjaisella IT-sivustolla. Käännöslisäosan käytettävyyden ei kuitenkaan ollut tämän verkkosivuprojektin kannalta kaikkein paras, vaan sen ominaisuudet pikemminkin rajoittivat käännösten toteuttamista. Lisäosaa ei kuitenkaan vaihdettu, vaan käännösten toteuttamisen osalta tyydyttiin niihin ominaisuuksiin, joita lisäosa tarjosi.

Mobiiliversion toimintaa tarkkailtiin koko toteutuksen ajan ja ratkaisusta pyrittiin tekemään mahdollisimman responsiivisia, jotta ne toimisivat oikein myös mobiililaitteissa. Erityisesti työpöytäversiossa monisarakkeisena toteutettu valikko aiheutti alkuun haasteita, kun valikon rakenne ei kestänyt suoraan mobiiliversion kääntämistä. Merkittävimpiä haasteita sivun rakentamisessa olivat sisällön yhtenäistäminen, kaksikielisyyden toteuttaminen, sekä sivurakenteen ja sen myötä valikkorakenteen kokoaminen loogiseksi kokonaisuudeksi. Sisällön käsitteleminen oli melko hidasta ja raskasta, koska suuri osa vanhasta sisällöstä oli vanhentunutta ja niiden muotoilut ja tyylit vaihtelivat. Sivuston julkaiseminen onnistui haasteista huolimatta asiakkaan antaman aikataulun puitteissa.

4.3 Asiakkaan ohjeistaminen

Asiakkaalle jäävässä dokumentaatioissa käytiin läpi sivuston toiminnan ja päivittämisen kannalta olennaiset asiat. Erityisesti keskityttiin kertaamaan sisällöntuotantoon liittyviä asioita ja ydintoimintojen ylläpitämisen pääkohtia. Projektin päätyttyä asiakkaalle toimitetun

ohjeistuksen tarkoitus oli taata perustoimintojen, kuten sivujen ja artikkeleiden lisäämisen, helppous. Lisäksi asiakkaalle luvattiin antaa tarvittaessa etätukea sivustoon liittyvien ongelmien ratkaisemiseen muutama kuukausi projektin päättymisen jälkeen.

Päivitysohjeessa pyrittiin avaamaan sivuston pääperiaatteet riittävällä, mutta tehokkaalla tavalla. Selkeän ohjeistuksen merkitys asiakkaan kannalta voi olla kriittisen merkittävä, mikäli asiakkaalla itsellään ei ole riittävää kokemusta kyseisen järjestelmän käyttämisestä ja ylläpitämisestä. Lisäksi sisällön ylläpitämisen ja lisäämisen ohjeistusten avulla asiakas pystyy halutessaan pitämään sisällön yhtenäisenä ja johdonmukaisena, mikä taas lisää sivuston laadukkuutta ja näyttäytyy myös loppukäyttäjälle viimeistellympänä kokonaisuutena.

Sivun tai tiedotteen lisääminen ja muokkaaminen

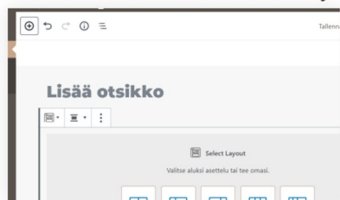
Ylävalikon painikkeen **+ Uusi** alta voit lisätä uuden sivun tai tiedotteen (artikkelin). Klikkaamalla siirryt suoraan tekstieditoriin.

Lisää sivulle otsikko.

Voit valita valmiin asettelun, mikäli tiedät jo etukäteen tarvitsevasi monipalstaisen sivun. Mikäli yksi palsta riittää, valitse **Ohita**.

Sisältö muotoillaan blokeilla ja yleensä pelkästään otsikko- ja kappaleblokit riittävät. Lohkon lisäät editorin **+** painikkeella.

Valitse tiedotteelle oikean reunan valikosta **kategoria**: Käyttökatos, uutta tai yleinen. Voit lisätä myös tiedotteen sisältöä kuvaavia avainsanoja.



Blokkityypit

Sivu muodostuu esimuotoilluista blokeista. Yleensä sisällön saa järjestykseen viidellä perusblokillä:

Otsikko

Otsikot **H3**

Alaotsikot **H4**

Mikäli et halua otsikkoa sisällysluetteloon, käytä otsikkotyylillä H5 tai Kappale-blokkia, jossa teksti on lihavoitu.

Kappale

Perustekstiä, voit muotoilla yksittäisiä osia teksteistä lihavoimalla, kursivoimalla ja tehdä linkkejä.

Taulukko

Taulukoita käytetään vain, mikäli sisältö sitä vaatii.

Lista

Lähtökohtaisesti listoissa käytetään bulletteja. Mikäli tarkoitus on kuvata kronologisia vaiheita, voit käyttää myös numeroitua listaa.

Kuva

Kuva-blokillä voit lisätä sivulle kuvan joko kuvakirjastosta, lataamalla omalta tietokoneelta tai URL-osoitteesta.

Kuvat keskitetään **vasempaan** reunaan

Kuva 12. Asiakkaalle tehdystä ohjeistuksesta esiteltiin esimerkiksi sivun muotoiluun käytettävät lohkot (blokit)

Ohjeessa käytiin kuvan 12 mukaisesti läpi, millaisista elementeistä sivut on rakennettu, jotta arvailulta ja yksittäisen käyttäjän mielivaltaisilta valinnoilta voitaisiin välttyä. Tavoitteena oli antaa asiakkaan sisällöntuottajille riittävä pohjatieto, jonka avulla uusien sivujen luominen sujusi jouhevasti ja tehokkaasti. Erityisesti sisältösivujen luomiseen liittyvissä ohjeissa pyrittiin huomioimaan käyttäjät, joilla ei ole ennestään tavanomaisia toimisto-ohjelmia lukuun ottamatta kokemusta sisällönhallintaohjelmista tai tekstin ja kuvien asettelusta. Typografialtaan sivusto noudattelee LAB-ammattikorkeakoulun omaa graafista ohjeistoa ja eri tekstityylit vakioitiin suoraan WordPressissä. Tämän ja valmiiden lohkojen käytön tarkoituksena oli vähentää käyttäjävirheitä.

Lisäksi asiakasta ohjeistettiin siitä, millaista sävyä (kuva 13) ohjeteksteissä tulisi tavoitella. Ohjeiden sävyt käytiin läpi hyvän ja huonon esimerkin avulla. Ohjeistamalla asiakasta esimerkkien avulla toivottiin, että kahden erilaisen tekstin sävyero tulisi mahdollisimman selkeästi esille. Tavoitteena oli tarjota asiakkaalle jatkossakin konkreettinen ohjenuora sisällön tyylistä ja välttää vanhan wikisivuston virheet tältä osin.

Sisällön tyyli

- » Käytä käskymuotoa
- » Pyri tekemään ohjeista napakoita ja selkeitä
- » Kerro vain olennaiset asiat, kaikkia asiaan liittyviä toiminnallisuuksia ei tarvitse esitellä itse ohjeessa, jos ne eivät ole halutun asian suorittamisen kannalta olennaisia
- » Tarvittaessa havainnollista ruutukaappauksilla
- » Selkeät vaiheet voi kirjoittaa listaksi
- » Ilmaise selkeästi toiminnot, jotka käyttäjän tulee tehdä päästäkseen haluttuun lopputulokseen
- » Pidä sisältö yhteydenmukaisena ja tarvittaessa kirjoita termejä auki napakan ohjeen alapuolelle

Oikealla suuntaa-antavat esimerkit selkeästä, napakammasta ohjeesta ja tarinnallisemmasta tyylistä, jota tulee välttää.

» Mene osoitteeseen peppi.lab.fi

» Kirjaudu sisään koulun AD-tunnuksilla

Onnistuneen sisäänkirjautumisen jälkeen näet etusivun, jonka sisältö vaihtelee työpöydän mukaan.

Opiskelijoilla etusivulla näkyy esimerkiksi yhteenveto opintojen etenemisestä, pikalinkkejä ja uusimmat tiedotteet.

Kirjoita osoiteriville peppi.lab.fi ja paina enteriä, jonka jälkeen avautuu kirjautumissivu. Kirjautuminen tehdään koulun tunnuksilla.

Kun kirjautuminen on onnistunut aukeaa Pepin etusivu. Opiskelijana näet sinisellä teemalla olevan työpöydän, jossa on yhteenveto opinnoista ja uutisia.

Kuva 13. Sisällön tyyli ohjeistettiin sekä listamuotoisina ohjeina, että esimerkkien avulla

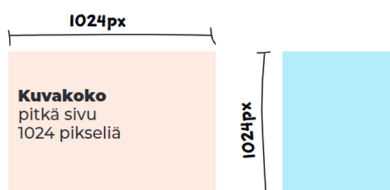
Tekstisisällön lisäksi asiakasta ohjeistettiin kuvasisällön osalta (kuva 14). Kuvien osilta käytiin läpi mihin tarkoitukseen niitä voitaisiin sivustolla käyttää ja ohjeistettiin sivuston luonteen vuoksi, ettei kuvia käytettäisi pelkästään koristeellisina elementteinä. Lisäksi kuvan lisääminen WordPressin mediakirjastoon käytiin läpi. Asiakasta ohjeistettiin lisäämään kaikille kuville myös selittävä vaihtoehtoiset tekstit ja informatiiviset tiedostonimet. Näiden tarkoituksena oli sekä parantaa sivuston saavutettavuutta, että auttaa mediakirjaston hallinnoimista. Sivuston latausnopeuden ylläpitämiseksi asiakkaalle ohjeistettiin käyttämään kuvia, joiden pitkän sivuun pituus on enimmillään 1024 pikseliä. Tähän kuvakokoon päädyttiin, koska kuvien osalta tavoiteltiin mahdollisimman hyvää tiedostokoon ja kuvanlaadun suhdetta. Lisäksi valittu kuvakoko on linjassa yleisesti verkkoympäristössä suosittujen kuvakokojen kanssa. Kuvien lisääminen käytiin läpi vaiheina ja kuvakokoa havainnollistamassa käytettiin kahta suorakaidetta.

Kuvat

Kuvia voi käyttää esimerkiksi havainnollistamassa ohjeita. Älä käytä kuvia pelkästään koristeena.

Kuvan lisääminen mediakirjastoon

- » Valitse vasemman reunan valikosta **Media** -> **Lisää uusi**
- » Pudota ladattavat tiedostot latausnäkyymään tai paina **Valitse tiedostot**
- » Varmista, että kuvalla on informatiivinen nimi
- » Mikäli haluat muuttaa kuvan tietoja, paina latausnäkyymässä kuvan vieressä olevaa **Muokkaa**-linkkiä



Kuvan tietojen lisääminen jälkikäteen

- » Avaa **Mediakirjasto** ja haluamasi kuva
- » Täytä kaikki kentät

Kuvalle lisättävät tiedot

Vaihtoehtoinen teksti

Kuvaile kuvan tarkoitusta. Vaihtoehtoinen teksti parantaa sivun saavutettavuutta näkörajoitteisille henkilöille. Se auttaa myös parantamaan hakukonenäkyvyyttä.

Otsikko

Otsikko on lyhyt teksti, josta selviää kuvan sisältö ja se erottuu muista kuvista. Otsikko voi olla esimerkiksi muotoa *LAB_logo* tai *etusivu_opiskelija*.

Kuvateksti

Kuvan sisällön tarkempi, lausemuotoinen kuvailu. Sisältö riippuu kuvan käyttötarkoituksesta.

Kuvaus

Kuvaus tulee näkyviin kun hiiri on kuvan päällä. Kuvaus voi olla yhyempi versio kuvatekstistä, esimerkiksi *Opettajan arviointinäkymä*.

Kuva 14. Kuvasisällölle ohjeistettiin suositeltu kuvakoko, sekä avattiin metatekstien käyttöä

Käyttöoikeuksien hierarkia avattiin asiakkaalle toimitetussa dokumentaatiossa kuvassa 15 näkyvän taulukon avulla, jotta asiakkaalla olisi työkalut arvioida itse, minkä tason oikeudet olisivat tarkoituksenmukaisia kullekin käyttäjälle. Koska sivulle ei tehty kirjautumisen takana olevaa sisältöä, eikä näin ollut tarkoitus tehdä myöskään tulevaisuudessa, päädyttiin käyttäjäoikeuksien osalta suosittelemaan neljää eri roolia: admin (pääkäyttävä), editor (toimittaja), author (sisällöntuottaja) ja contributor (kirjoittaja).

Uuden käyttäjän lisääminen

Lisää uusi käyttäjä vasemman valikon **Käyttäjät**-kohdasta. Valitse käyttäjänimi ja salasana, sekä tarvittava rooli. Lähetä käyttäjälle tunnus ja salasana sähköpostitse.

Käyttäjien roolit

Toiminto	Admin	Editor	Author	Contributor
Sisällön julkaiseminen	X	X	X*	
Sisällön luominen	X	X	X	X
Sisällön muokkaaminen	X	X	X*	X*
Sisällön poistaminen	X	X	X*	X*
Tiedostojen lataaminen	X	X	X	
Uusien käyttäjien lisääminen	X			
Ulkoasun muokkaaminen	X			
Lisäosien asentaminen	X			
Käyttäjien tietojen muokkaaminen	X			
Muut ylläpitotoiminnot	X			

* Ei voi hallita muiden käyttäjien luomaa sisältöä, vain omaansa

Kuva 15. Käyttäjän lisääminen ja käyttäjäroolien esittely taulukkomuodossa

Vaikka WordPressin käyttöliittymää voidaan pitää yleisesti melko loogisena, ohjeistettiin asiakkaalle silti myös perustoimintojen, kuten sivun ja kuvien lisäämisen, tekeminen. Tällä pyrittiin varmistamaan se, ettei tulevien käyttäjien tarvitse opetella sivun käyttöä yrityksen ja erehdyksen tai itseopiskelun kautta, vaan heillä on aina käytössään valmis dokumentaatio, josta perustoimintojen tehtäväpolut voi tarvittaessa tarkistaa. Erillinen valmis ohjeistus myös helpottaa tilanteita, joissa asiakkaan täytyy nopealla aikataululla saada uusi henkilö perehdytettyä sivuston päivittämiseen.

5 YHTEENVETO

Teoriaosuudessa käydään läpi yleisesti verkkosivujen käyttäjälähtöiseen suunnitteluun liittyviä asioita. Käyttäjien huomioiminen parantaa sivuston käytettävyyttä, jonka avulla taas käyttäjäkokemus paranee. Käytettävyyteen liittyvistä asioista etenkin saavutettavuuden huomioiminen voi olla ratkaisevassa asemassa yksittäisen käyttäjän kannalta. Mikäli sivuston suunnittelussa ei ole otettu huomioon saavutettavuuteen liittyviä asioita, kuten luetavuutta ja värejä, voi verkkosivun käytettävyys olla erittäin puutteellinen käyttäjälle, jolla on jonkinasteisia aistillisia tai kognitiivisia rajoitteita. Käyttäjäkokemuksen vaivattomuus pyrittiin huomioimaan myös työosuuden verkkosivun toteutuksessa, erityisesti sisällöntuottajat huomioiden.

Työosuudessa asiakkaalle toteutettiin WordPress-pohjainen verkkosivusto, jolla julkaistiin ammattikorkeakoululla käytössä olevan Peppi-opintotietojärjestelmään liittyviä tiedotteita ja ohjeita järjestelmän käyttöön. Verkkosivusto toteutettiin syksyllä 2019. Verkkosivustolle asetettiin aikataulun lisäksi joitakin tavoitteita: selkeä käyttöliittymä, ammattikorkeakoulun ilmeen mukainen ulkoasu ja hyvä käytettävyys. Käytettävyys huomioitiin sekä ylläpitäjien, että sivun loppukäyttäjien kannalta. Ylläpitöpuolen käytettävyys otettiin huomioon valitsemalla sivun rakentamiseen selkeitä ja helposti sisäistettäviä elementtejä, joiden avulla erityisesti sisällöntuottajien työtä pyrittiin helpottamaan. Loppukäyttäjien huomioiminen näkyi lähinnä ulkoasusuunnittelussa ja sisällön rakenteessa. Ulkoasun värimaailma koostui rauhallisista väreistä ja sivuston rakenne noudatteli perinteistä käyttöliittymärakennetta. Sivuston julkaisu tapahtui aikataulussa ja julkaisun jälkeen asiakkaalle tarjottiin sivun ylläpitoon liittyvää tukea vielä muutaman kuukauden ajan.

Sisällönhallintajärjestelmä itsessään luo hyvät puitteet sellaisten verkkosivujen rakentamiseen, jotka ovat saavutettavia ja joiden käytettävyys sekä loppukäyttäjille, että sisällöntuottajille ja ylläpitäjille on kunnossa. Tästä huolimatta verkkosivun suunnittelussa tulee ottaa tavoitteiden mukaisesti huomioon käytettävyyteen liittyvät asiat, eikä vain luottaa järjestelmän valmiisiin ominaisuuksiin. Verkkosivujen rakenne, sisältö ja toiminnallisuus voidaan optimoida käyttäjälähtöisesti. Toimintoja voidaan suoraviivaistaa ja sivun eri osia korostaa visuaalisesti niin, että ne erottuvat vaivattomasti toisistaan. Esimerkiksi ruudunluo-kuohjelmat pystyvät kyllä tulkitsemaan suoraan HTML-merkkauskieltä, mutta kuvien merkitys näkörajoitteiselle jää epäselväksi, jos verkkosivun sisällön suunnittelussa ei tätä olla huomioitu vaihtoehtoisten tekstien avulla. Käytettävyyden ja erityisesti saavutettavuuden huomioiminen varmistaa sen, että verkkosivu ei suoraan sulje mitään käyttäjäryhmää pois.

LÄHTEET

Painetut lähteet

Goodman, E., Kuniavsky, M. & Moed, A., 2012, Observing the user experience – a practitioner's guide to user research, Waltham, Elsevier

Jokela, T., 2010, Navigoi oikein käytettävyyden vesillä, Väylä-Yhtiöt Oy

Kuutti, W., 2003, Käytettävyys, suunnittelu ja arviointi, Saarijärvi, Talentum Media Oy/paino Gummerus Kirjapaino Oy

Shivakumar, SK. 2016, Enterprise Content and Search Management for Building Digital platforms, Hoboken, John Wiley & Sons Inc.

Tullis, T., Albert, B. 2013. Measuring the User Experience, Waltham, Elsevier

Weber, S., 2004, The Success of Open Source, Lontoo, Harvard University Press

Elektroniset lähteet

All About UX. User experience definitions. Viitattu 14.5.2020. Saatavissa: <http://www.allaboutux.org/ux-definitions>

Aluehallintovirasto. Ohjeita suunnittelun tueksi. Viitattu 23.9.2020. Saavutettavuusvaatimukset. Saatavissa: <https://www.saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/ohjeita-suunnittelun-tueksi/#teksti>

Apple. Dark Mode. Viitattu 11.9.2020. Human Interface Guidelines. Saatavissa: <https://developer.apple.com/design/human-interface-guidelines/ios/visual-design/dark-mode>

Brightspot. The pros and cons of coupled, decoupled and headless CMS architecture. Viitattu 19.11.2020. Saatavissa: <https://www.brightspot.com/solutions/decoupled-cms-and-headless-cms-platforms>

Celia. Kuvien vaihtoehtoiset tekstit. Viitattu 1.4.2020. Saavutettavasti.fi. Saatavissa: <https://www.saavutettavasti.fi/kuva-ja-aani/kuvat/>

Celia. Värit ja kontrastit. Viitattu 15.7.2020. Saavutettavasti.fi. Saatavissa: <https://www.saavutettavasti.fi/kuva-ja-aani/varit-ja-kontrastit/>

Heikkinen, S. 2018. Mukauta Windows 10:n ulkonäkö mieleiseksesi. Viitattu 15.11.2020. Yle Digitreenit. Saatavissa: <https://yle.fi/aihe/artikkeli/2018/03/21/digitreenit-mukauta-windows-10n-ulkonako-mieleiseksesi>

Huttunen, M., Socada, L. ADHD (aktiivisuuden ja tarkkaavuuden häiriö). Viitattu 18.10.2020. Terveyskirjasto Duodecim. Saatavissa: https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=dlk00353

Invalidiliitto. Saavutettavuus. Viitattu 13.6.2020. Saatavissa: <https://www.invalidiliitto.fi/es-teettomyys/saavutettavuus>

Jackson, B. 2020. Joomla vs WordPress. Viitattu 20.11.2020. Kinsta. Saatavissa: <https://kinsta.com/blog/joomla-vs-wordpress/>

Javatpoint. History of Joomla. Viitattu 19.11.2020. Saatavissa: <https://www.javatpoint.com/history-of-joomla>

Kinsta. The History of WordPress, its Ecosystem and Community. Viitattu 18.6.2020. Saatavissa: <https://kinsta.com/learn/wordpress-history/>

Kuntoutussäätiö. 2020. Psykkisen hyvinvoinnin vaikutus oppimiseen. Viitattu 17.9.2020. Oppimisvaikeus.fi. Saatavissa: <https://oppimisvaikeus.fi/materiaalit/oppimisvaikeuksien-tunnistaminen-monikulttuurisilla-oppijoilla/psykkinen-hyvinvointi-ja-sen-vaikutus-oppimiseen/>

Lamprecht, E. 2020. The Difference Between UX And UI Design. Viitattu 3.11.2020. Careerfoundry. Saatavissa: <https://careerfoundry.com/en/blog/ux-design/the-difference-between-ux-and-ui-design-a-laymans-guide/>

Lawton Henry, S. 2005. Introduction to Web Accesibility. Viitattu 12.4.2020. W3C Web Accessibility Initiative. Saatavissa: <https://www.w3.org/WAI/fundamentals/accessibility-intro/#context>

Mankki, M. 2004. Värit ja niiden merkitys käyttöliittymän visuaalisessa suunnittelussa. Viitattu 2.5.2020. Käyttöliittymäpsykologia. Saatavissa: <https://docplayer.fi/3073889-Varit-ja-niiden-merkitys-kayttoliittymän-visuaalisessa-suunnittelussa.html>

Namecheap. WordPress History from Creation to Today. Viitattu 2.7.2020. Saatavissa: <https://www.namecheap.com/wordpress/history-of-wordpress/>

Nichols, D. We see colors differently. Viitattu 18.9.2020. Coloring for Colorblindness. Saatavissa: <https://davidmathlogic.com/colorblind/>

Open Source Matters inc. About Open Source Matters. Viitattu 24.10.2020. Open Source Matters. Saatavissa: <https://www.opensourcematters.org/organisation.html>

Prohaska S. 2020. WordPress vs. Joomla 2020. Viitattu 26.10.2020. Ithelps. Saatavissa: <https://www.ithelps-digital.com/en/blog-en/websites/wordpress-joomla>

Pignataro, M. History of Joomla. Viitattu 18.11.2020. CorePHP. Saatavissa:

<https://www.corephp.com/blog/history-joomla-essay/>

Rodriquez, J. 2018. The easiest way to keep your web apps accessible: Just use text . Viitattu 15.10.2020. LogRocket. Saatavissa:

<https://blog.logrocket.com/the-easiest-way-to-keep-your-web-apps-accessible-c2b57506cc2a/>

Roto, V., Law, E., Vermeeren, A., Hoonhout, J. 2011. User Experience White Paper. Viitattu 12.5.2020. All About UX, Saatavissa:

<http://www.allaboutux.org/files/UX-WhitePaper.pdf>

Salakari, K. 2016. Mikä on käyttöliittymä? Viitattu 2.6.2020. Saatavissa:

<https://www.salakari.net/dwkb/01-mika-on-kayttoliittyma/>

Toivanen, A. 2015. Hyvä käyttöliittymä - 10 muistisääntöä. Viitattu 23.6.2020. Graafinen.com. Saatavissa:

<https://www.graafinen.com/suunnittelu/digi/hyva-kayttoliittyma-10-muistisaantoa/>

Tolvanen, P. 2009. Käsitesekamelskaa: julkaisujärjestelmä, CMS, portaali, sisällönhallintajärjestelmä. Viitattu 4.11.2020. Vierityspalkki. Saatavissa:

<https://vierityspalkki.fi/2009/11/03/kasitesekamelskaa-julkaisujarjestelma-cms-portaali-sisallonhallintajarjestelma/>

Topi, H., Lucas, W. & Babaian, T. 2005. Identifying Usability Issues with an ERP Implementation. Viitattu 16.11.2020. Saatavissa:

https://www.researchgate.net/publication/220710435_Identifying_Usability_Issues_with_an_ERP_Implementation

UX Guider. Käytettävyydestä yleisesti. Viitattu 18.7.2020. Ux Guider. Saatavilla:

<https://uxguider.fi/kaytettavyydesta.html>

Ventura, M. 2017. Editor Technical Overview. Viitattu 2.11.2020. Make WordPress Core. Saatavissa:

<https://make.wordpress.org/core/2017/01/17/editor-technical-overview/>

W3Techs. Usage statistics of content management systems. Viitattu 12.10.2020.

W3Techs - World Wide Web Technology Surveys. Saatavissa: https://w3techs.com/technologies/overview/content_management

Wikipedia. MacOS. Viitattu 15.10.2020. Saatavissa: <https://fi.wikipedia.org/wiki/MacOS>

Winkels, N. 2019. The definitive guide to CMS architecture. Viitattu 19.11.2020. Bloomreach. Saatavissa:

<https://developers.bloomreach.com/blog/2019/cms-architecture.html#decoupled>

Wordpress. History. Viitattu 12.6.2020. Saatavissa: <https://wordpress.org/support/article/history/>

WPBeginner. Evolution of WordPress User Interface. Viitattu 14.6.2020. Saatavissa: <https://www.wpbeginner.com/showcase/evolution-of-wordpress-user-interface-2003-2009/>