

KARELIA-AMMATTIKORKEAKOULU
Viestinnän koulutusohjelma

Pieta Honkakoski

ELEMENTTIKIRJASTON KONSEPTI JA PROTOTYYPPI

Opinnäytetyö
Toukokuu 2017



OPINNÄYTETYÖ
Toukokuu 2017
Viestinnän koulutusohjelma

Länsikatu 15
80110 JOENSUU
013 260 600

Tekijä(t)
Pieta Honkakoski

Nimeke
Elementtikirjaston konsepti ja prototyyppi

Toimeksiantaja
atFlow Oy

Tiivistelmä

Opinnäytetyössä tutkittiin miten verkkosivujen tuotantoprosessi voidaan muuttaa modulaariseksi. Konseptissa määritellään elementtikirjaston ominaisuudet ja tarkoitus. Elementtikirjasto on verkkosuunnittelijoille ja front-end –toteuttajille tarkoitettu työkalu verkkosivuteemojen rakentamiseen. Se on päivitettävä kirjasto, joka sisältää testattuja, laadukkaita ja uudelleenkäytettäviä käyttöliittymäelementtejä.

Työn ja toimeksiantajan tavoitteena on tuottaa responsiivisia verkkosivuteemoja kustannustehokkaasti ja nopeasti. Kehitysideoiden pohjalta yritys voi kehittää elementtikirjastoa eteenpäin. Prototyyppi on yrityksen sisäisille sivuille toteutettu elementtikirjaston responsiivinen etusivu. Se on toimiva kokonaisuus, jossa voi lisätä, ladata ja poistaa elementtejä käyttämällä erillistä FTP-ohjelmaa.

Elementtikirjaston avulla verkkosivujen tuotantoprosessista voidaan tehdä modulaarinen. Sinne ladattavien elementtien laatu määrittelee tuotettujen verkkosivuteemojen laadun. Elementtien toteuttamisen avuksi ehdotetaan atomisen suunnittelujärjestelmän ja 8-point gridin käyttöä.

Kieli
suomi

Sivuja 48
Liitteet 3
Liitesivumäärä 3

Asiasanat
responsiivinen verkkosuunnittelu, konseptisuunnittelu, prototyypittäminen, elementtikirjasto, modulaarisuus



THESIS
May 2017
Degree Programme in Communication

Länsikatu 15
80110 JOENSUU
FINLAND
013 260 600

Author (s)
Pieta Honkakoski

Title
Concept and Prototype of the Element Library

Commissioned by
atFlow Oy

Abstract

The purpose of this thesis was to study how the process of creating websites can be made modular. The concept defines the purpose and features of the element library. The element library is a tool for building website themes and it is used by web designers and front-end coders. The library can be updated and contains reusable, tested and high-quality interface elements.

The goal of the commissioner and this work is to produce responsive website themes fast and cost-effectively. Based on the development ideas the commissioner can improve the element library. The prototype is a responsive front page of the element library that is build on the commissioners private site. The element library is a working whole where you can add, download and remove elements using an FTP-program.

It is possible to make the process of creating websites modular by using the element library. The quality of the final product is defined by the quality of the elements added into the element library. To ease the process of creating elements it is suggested to use design systems like atomic design and 8-point grid.

Language
Finnish

Pages 48
Appendices 3
Pages of Appendices 3

Keywords
responsive web design, web design, concept design, prototyping, element library, modularity

Sisältö

Lyhenteet	5
1 Johdanto	7
2 Verkkosuunnittelu.....	8
2.1 Verkkosuunnittelun perusteet.....	8
2.2 Responsiivinen verkkosuunnittelu	9
2.3 Mobile First.....	12
2.4 Vesiputousmalli	13
2.5 Staattisen sommitelman ongelmia.....	15
3 Prototyypittäminen	16
3.1 Prototyypittämisen perusteet.....	16
3.2 Prototyypittäminen responsiivisessa verkkosuunnittelussa.....	17
4 Modulaarisuus.....	18
4.1 Modulaarisuuden perusteet.....	19
4.2 Modulaarisuus verkkosuunnittelussa.....	20
4.3 Modulaarisuus visuaalisessa suunnittelussa.....	22
4.4 Suunnittelujärjestelmät	24
5 Atomisen suunnittelu.....	27
5.1 Atomisen suunnittelun perusteet	27
5.2 Atomisen suunnittelujärjestelmän rakenne.....	27
6 Elementtikirjaston konsepti ja prototyyppi.....	30
6.1 Opinnäytetyön tavoite.....	30
6.2 Elementtikirjaston tavoite.....	30
6.3 Elementtikirjaston konsepti.....	32
6.4 Elementtikirjaston konseptin ensimmäinen kehitysversio.....	33
6.5 Elementtikirjaston kehitysehdotukset	37
6.6 Elementtikirjaston prototyyppi.....	39
6.7 Prototyypin toteutus.....	40
7 Tulokset ja pohdinta	43
7.1 Toiminnallisen opinnäytetyön tulokset.....	43
7.2 Opinnäytteen tulosten tarkastelu	44
7.3 Oppimisprosessi ja ammatillinen kasvu.....	45
Lähteet	47

Liitteet

Liite 1	Elementtikirjaston käyttöliittymän työpöytäversio
Liite 2	Elementtikirjaston käyttöliittymän tablettiversio
Liite 3	Elementtikirjaston käyttöliittymän mobiiliversio

Lyhenteet

HTML	HTML on verkkosivujen rakentamisessa käytetty kieli, joka kuvailee verkkosivuston rakennetta (Duckett 2014, 20).
CSS	CSS on verkkosivujen tyylien rakentamisessa käytetty kieli, joka määrittelee miltä sivuston elementit näyttävät (Duckett 2014, 227-229).
FLUID GRID	Fluid grid on relatiivisiin määreisiin perustuva joustava asettelu ja yksi responsiivisen verkkosuunnittelun tärkeimmistä osista (Marcotte 2015, 10, 44).
FIXED WIDTH	Fluid gridin vastakohta. Relatiivisten määreiden sijasta asettelu on rakennettu staattisilla pikseleillä (Duckett 2014, 381-382).
LIQUID LAYOUT	Relatiivisen ja responsiivisen asettelun yksi muoto, joka perustuu prosenteilla määriteltyihin leveyksiin (Duckett 2014, 381-382).
MEDIA QUERY	CSS –ehtolause, joka rajoittaa sivustolla käytettyjen tyylien laajuutta annettujen määritysten mukaan (Carver 2015, 7).
MOBILE FIRST	Luke Wroblewskin kehittämä suunnittelutapa, joka keskittyy mobiilikokemukseen ennen muita näyttökokoja (Wroblewski 2009).
OOCSS	Nicole Sullivanin kehittämä modulaarinen tapa tuottaa nopeaa, päivitettävää ja standardoitua koodia (Sullivan, 2013).

SMACC	Jonathan Snookin luoma tyyliohje modulaariseen verkkosuunnitteluun (Snook, 2017).
BEM	Yandexin kehittämä CSS-tyyliin nimeämiskäytäntö (Starkov & Stepanova 2017).
FRAMEWORK	Valmiita suunnittelujärjestelmiä, koodikirjastoja ja perustoja rakennettaville verkkosivuille (Duckett 2014, 391).
8-POINT GRID	Käyttöliittymien visuaaliseen yhtenäisyyteen keskittyvä suunnittelujärjestelmä (Dahl 2016).
FTP	File transfer protocol eli FTP mahdollistaa tiedostojen siirron paikallisen koneen ja verkkosivuston palvelimen välillä (Duckett 2014, 489).

1 Johdanto

Opinnäytetyön toimeksiantajana toimii atFlow Oy, joka on vuonna 2008 perustettu joensuulainen verkkokauppoja ja verkkosivuja tuottava monipuolinen digitoimisto. Yrityksenä atFlow keskittyy aktiivisten verkkosivu- ja verkkokauppaprojektien lisäksi Dataflow Platformin kehitystyöhön, johon toiminnallinen opinnäytetyöni liittyy. Dataflow Platform on alusta, jossa jälleenmyyjät ja asiakkaat voivat hallita omia verkkosivu- ja verkkokauppaprojektejaan. Platformiin integroitu Marketplace on paikka, josta käyttäjät voivat ostaa valmiita verkkosivuteemoja käytettäväksi projekteissaan.

Elementtikirjaston konsepti ja prototyyppi pyrkivät antamaan uusia näkökulmia näiden verkkosivuteemojen tuotantoprosessiin. Jotta Dataflow Platformin sisällä toimivan Marketplacen konsepti toimisi tarvitaan kymmeniä testattuja, laadukkaita ja responsiivisia verkkosivuteemoja. Vanhat tuotantotavat eivät enää pysty vastaamaan tähän kasvavaan kysyntään. Elementtikirjaston on tarkoitus olla työkalu, joka muuttaa verkkosivustojen tuotantoprosessin modulaariseksi. Kirjasto ratkaisee koodin uusiokäyttöön liittyviä ongelmia. Modulaarinen lähestymistapa takaa, että jokaista uutta sivustoa lähestyttäessä voidaan luoda vahva, testattu perusta jonka päälle rakentaa sen sijaan, että verkkosivun rakentaminen aloitettaisiin aivan alusta.

Dataflow Platformiin toteutettavien verkkosivuteemojen ulkopuolella elementtikirjastoa voidaan käyttää myös yksittäisten, personoitujen verkkosivuprojektien tuotantoon. Verkkosivujen suunnittelijat ja tuottajat voivat toteuttaa julkaistavien verkkosivujen lisäksi visuaalisia ja interaktiivisia prototyyppejä näytettäväksi asiakkaille nopeasti ja kustannustehokkaasti.

Elementtikirjaston tavoitteena on siirtää projektien pääpaino front-end toteutuksesta tuotteiden ja palveluiden suunnitteluprosessiin, jossa asiakas ja projektin vaatimukset ovat pääosassa. Se helpottaa asiakkaiden kanssa kommunikointia, jolloin sivuston ulkoasun ja toteutuksen sijasta voidaan keskittyä tärkeimpiin toiminnallisuuksiin ja ominaisuuksiin sekä tuotteiden laatuun. Tuotantoprosessi

automatisoituu ja käytetty aika jakautuu tasaisemmin suunnittelu- ja tuotantovaiheiden kesken. Tuloksena on tehokkaammin rakennettuja ja laadukkaampia tuotteita.

Toiminnallisen opinnäytetyön tuloksena syntyy elementtikirjaston konsepti ja prototyyppi. Konseptissa määritellään, miten elementtikirjaston on tarkoitus toimia ja mihin sitä tullaan käyttämään. Elementtikirjaston etusivusta luodaan responsiivinen prototyyppi, jonka pohjalta konsepti voidaan viedä kehitykseen ja lopuksi tekniseen tuotantoon. Sen tarkoitus on avata keskustelua uusista työtyövoista ja kehittää konseptia eteenpäin.

2 Verkkosuunnittelu

2.1 Verkkosuunnittelun perusteet

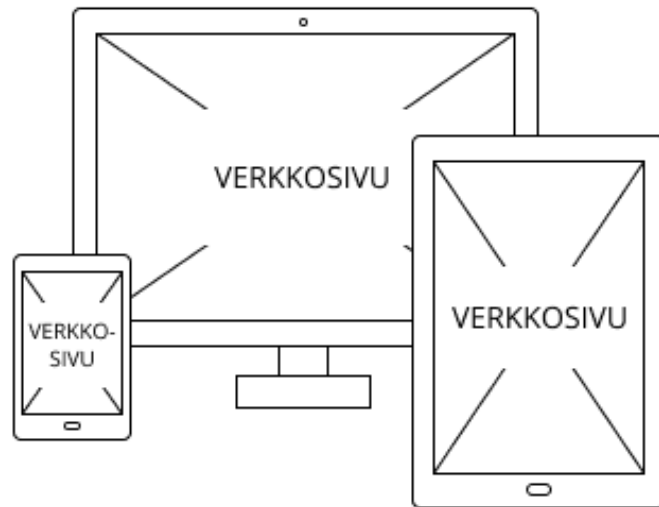
Verkkosuunnittelu on saanut alkunsa graafisten käyttöliittymien luomisesta ihmisen ja teknologian välille (Macdonald 2003, 20). Se on suhteellisen uusi suunnittelun osa-alue, jonka olemassaolon on mahdollistanut tietokoneiden leviäminen koko kansan käyttöön. Ensimmäinen verkkoselain jolla pystyi sekä katsomaan, että kehittämään verkkosivuja oli Marc Anderseenin johdolla luotu Mosaic. Se rakennettiin Steve Jobsin kehittämälle NeXT –tietokoneelle. Nykyinen verkko sai alkunsa kun Anderseen ja Jim Clark perustivat Netscape Communications Corporationin vuonna 1995 ja loivat sen ajan suosituimman nettiselaimen, Netscapen. (Macdonald 2003, 18.)

Verkkosuunnittelu ottaa myös vaikutteita perinteisestä interaktio- ja graafisesta suunnittelusta. Verkkosivuja käytetään palveluiden ja tuotteiden alustoina, sekä informaation ja toiminnallisuuksien esittämisessä ja visualisoinnissa verkossa. Muihin perinteisiin suunnittelun aloihin verrattuna verkkosuunnittelun tuloksena ei synny yhtä fyysistä tuotetta, vaan verkkosivu muokkautuu alati erilaisten näytöpöytänteiden ja kehittyvien teknologioiden myötä digitaalisessa ympäristössä. (Macdonald 2003, 6.)

2.2 Responsiivinen verkkosuunnittelu

Graafisten verkkoselainten kehittyminen ja erilaisten näyttöpäätelaitteiden määrän nousujohteinen kasvu koitui ongelmalliseksi verkkosuunnittelijoille. Suunnittelija ei voinut ennalta tietää, millä laitteella ja missä olosuhteissa käyttäjä tulisi näkemään tuotetun sivuston. Mobiililaitteille erikseen tuotettujen sivujen määrä oli kasvussa, mutta tapa jolla ne rakennettiin ei kyennyt vastaamaan kehittyvän teknologian haasteisiin. Sen sijaan, että suunnittelijat olisivat tyytyneet tuottamaan samaa sivustoa uudestaan jokaiselle eri laitteelle, apuun tuli responsiivinen verkkosuunnittelu. (Carver 2015, 4.)

Responsiivinen verkkosuunnittelu on verkkosivujen suunnittelua skaalautuviksi erilaisilla selaimilla ja laitteilla, kuten älypuhelimilla ja tablet- ja pöytätietokoneilla (kuvio 1). Suunnittelutyölin tavoitteena on tuottaa verkkosivusto vain kerran yhteen URL-osoitteeseen. Sen tarkoitus on auttaa sivustoja sopeutumaan tulevaisuuden teknologian asettamiin haasteisiin ja pyrkiä antamaan käyttäjälle yhtenäinen kokemus kaikkien olemassaolevien laitteiden ja selainten kesken. (Carver 2015, xv, 5-6.)



Kuvio 1. Responsiivinen verkkosivusto skaalautuu eri laitteissa ja selaimissa.

Vuonna 2010 Ethan Marcotte kirjoitti artikkelin responsiivisesta verkkosuunnittelusta, joka levisi nopeasti suunnittelijoiden kesken ja auttoi suunnittelutapaa nousemaan alan standardiksi. Artikkelissa Marcotte esittelee responsiivisen verkkosuunnittelun olevan ratkaisu nopeasti muuttuvan alan, kehittyvien laitteiden ja alustojen synnyttämiin haasteisiin. Sen sijaan, että jokaiselle uudelle laitteelle ja selaimelle rakennettaisiin oma alkuperäisestä verkkosivusta muokattu erillinen kokonaisuus Marcotte ehdottaa alkuperäisen sivuston suunnittelua sopeutuvaksi jokaiselle eri laitteelle ja selaimelle mukaan lukien niille, joita ei ole vielä julkaistu. (Marcotte 2010.) Marcotten esittämät ennustukset responsiivisen verkkosuunnittelun tarpeesta olivat oikeassa, sillä jo vuoden 2014 tammikuussa ensimmäistä kertaa Yhdysvaltojen historiassa mobiililaitteita käytettiin Internetin selaamiseen enemmän kuin pöytätietokoneita (O'Toole 2014).

Responsiivisen verkkosuunnittelu koostuu kolmesta tärkeimmästä pääelementistä, jotka ovat joustavaan ruudukkoon ja relatiivisiin määreisiin perustuva fluid grid, tähän rakenteeseen mukautuva responsiivinen media sekä media queryt (Marcotte 2015, 11). Fluid grid eli joustava asettelu perustuu staattisten pikselien sijasta relatiivisiin määreisiin. Sivuston leveyden kasvaessa joustava

asettelu sopeutuu uusiin mittasuhteisiin automaattisesti ja poistaa tarpeen luoda erillisiä sivustoja eri laitteille ja selaimille. Joustava asettelu on responsiivisen verkkosivun perusta ja tärkein osa. (Marcotte 2015, 10, 44.)

Fixed width –asettelu on joustavan asettelun vastakohta. Asettelen leveys on mitattu pikseleissä, ja se ei muutu vaikka käyttäjä suurentaisi tai pienentäisi selainikkunaa. Liquid layout, jota voidaan ajatella joustavan asettelun yhtenä muotona, on relatiivisiin määreisiin kuten prosentteihin perustuva asettelu, joka mahdollistaa sivuston sopeutumisen selainikkunan koon vaihteluun. (Duckett 2014, 381-382.)

Aloittamalla joustavan asettelun rakentamisella kirjoitettavan koodin määrä vähenee huomattavasti. Relatiivisilla määreillä rakennettu asettelu on automaattisesti jo responsiivinen ja media queryilla tehtävät hienosäädöt ovat silloin paljon yksinkertaisempia. (Marcotte 2015, 103.) Median responsiivisuus perustuu joustavan asettelun skaalautuvuuteen. Käyttämällä kuvissa ja eri mediatyypeissä relatiivisia määreitä ne muuttuvat sopeutuviksi ja ovat joustavasta asettelusta riippuvaisia. Asettamalla medialle relatiivisen maksimileveyden pikseleiden sijasta kuvat ja media säilyttävät alkuperäisen kuvasuhteensa kaikissa leveyksissä. (Marcotte 2015, 44.)

Media queryt kuuluvat W3C:n luomaan määrittelyyn CSS3:sta, joka esittelee ne seuraavana loogisena askeleena HTML4:n tukemista mediatyypeistä, joita käytetään liittämään sivustoon erilaisia tyylietodostoja riippuen selaimen tai tulostimen asetuksista (W3C 2012). Ne ovat CSS –ehtolauseita, jotka rajoittavat verkkosivulla käytettyjen tyylien laajuutta annettujen määritteiden mukaisesti (Carver 2015, 7).

Media queryilla määritellään katkeamiskohdat verkkosivuston leveydessä, joissa sivuston rakenne ja tyylit muuttuvat selaimen tai laitteen leveydelle sopivaksi. Tämä ominaisuus tekee suunnittelutavasta tehokkaan ja sivustoista sopeutuvia ja skaalautuvia. (Carver 2015, 9.) Media queryilla voidaan korjata sivuston ulkoasun visuaalisia virheitä sitä mukaa, kun selainikkunan koko muuttuu. Niillä voi-

daan myös optimoida sisältö näyttämään hyvältä jokaisella eri laitteella. (Marcotte 2015, 82.)

Responsiivinen verkkosuunnittelu on jatkuvuuden tarjoamista eri kontekstien välille. Yksinkertaisimmillaan se on yhden tuotteen tarjoamista lukemattomille laitteille ja selaimille. Joustava asettelu, responsiivinen media ja media queryt tekevät sen mahdolliseksi. (Marcotte 2015, 109.) Suunnittelutapaa tukee myös määritelmä käytettävyydestä, jonka mukaan objektit ja ympäristöt tulisi suunnitella mahdollisimman monelle ihmiselle käytettäväksi ilman muutoksia tai muutoksia (Butler, Holden & Lidwell 2010, 16). Myös yhtenäisyyden määritelmää jonka mukaan tuotteen käytettävyys paranee jos samanlaiset osat ovat visuaalisesti, toiminnallisesti, sisäisesti ja ulkoisesti samantyyppisiä (Butler ym. 2010, 56) voidaan ajatella yhdeksi responsiivisen verkkosuunnittelun tavoitteista.

2.3 Mobile First

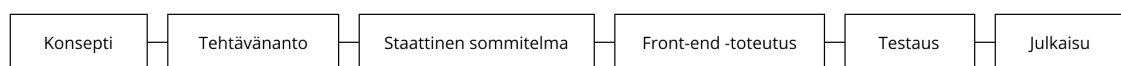
Kasvaessaan ja kehittyessään responsiivisten verkkosivujen tuottamiseen on ehdotettu erilaisia lähestymistapoja. Yksi niistä on Mobile First, joka ilmestyi Luke Wroblewskin blogikirjoituksessa (Wroblewski 2009). Tekstissään Wroblewski koittaa vakuuttaa ihmisiä suunnittelemaan ja tuottamaan verkkosivuja keskittymällä ensin mobiilikokemukseen. Keskittymällä kaikkein tärkeimpiin ominaisuuksiin ja lisäämällä elementtejä asteittain tilan kasvaessa suunnittelija saa keskittyä olennaiseen sen sijaan, että sivuston sisältöä joudutaan karsimaan ja rajoittamaan mitä pienemmäksi selainikkuna muuttuu. (Carver 2015, 32-34.) Tämän suunnittelutavan tehokkuutta tukee myös 80/20 –sääntö, jonka mukaan suuri prosenttiosuus tuotteen tekemistä vaikutuksista on pienen prosenttiosuuden tuottamaa. Tämä tarkoittaa sitä, että jos tuotteen olennaisimmat toiminnallisuudet ja käyttöliittymä ovat hyvin suunniteltuja, tuote on todennäköisesti parempi ja käytettävämpi. (Butler ym. 2010, 14.)

Mobile Firstin vastakohta on arvokas hajoittaminen, jossa monimutkainen kokonaisuus muutetaan yksinkertaisemmaksi rakentamalla turvaverkkoja rikkinaisten ominaisuuksien ympärille (Carver 2015, 34). Hyvä esimerkki arvokkaasta

hajoittamisesta löytyy Ethan Marcotten kirjasta *Responsive Web Design* (Marcotte 2015), joka syntyi Marcotten vuonna 2010 kirjoittaman artikkelin pohjalta. Marcotte esittelee responsiivista verkkosuunnittelua rakentamalla esimerkiksi sivuston, jossa hän toteuttaa suunnittelutavan kolmea tärkeintä piirrettä: joustavaa asettelua, responsiivista mediaa ja media queryita (Marcotte 2015, 11). Marcotte aloittaa sivuston rakentamisen kuitenkin pöytäkoneen leveydestä noudattaen visuaalisen suunnittelijan tuottamaa staattista sommitelmaa ja korjaa tämän lähestymistavan aiheuttamia ongelmia vasta jälkeinpäin sen sijaan, että lisäisi elementtejä yksi kerrallaan aloittaen mobiililaitteiden leveyksistä. (Marcotte 2015, 68-74.)

2.4 Vesiputousmalli

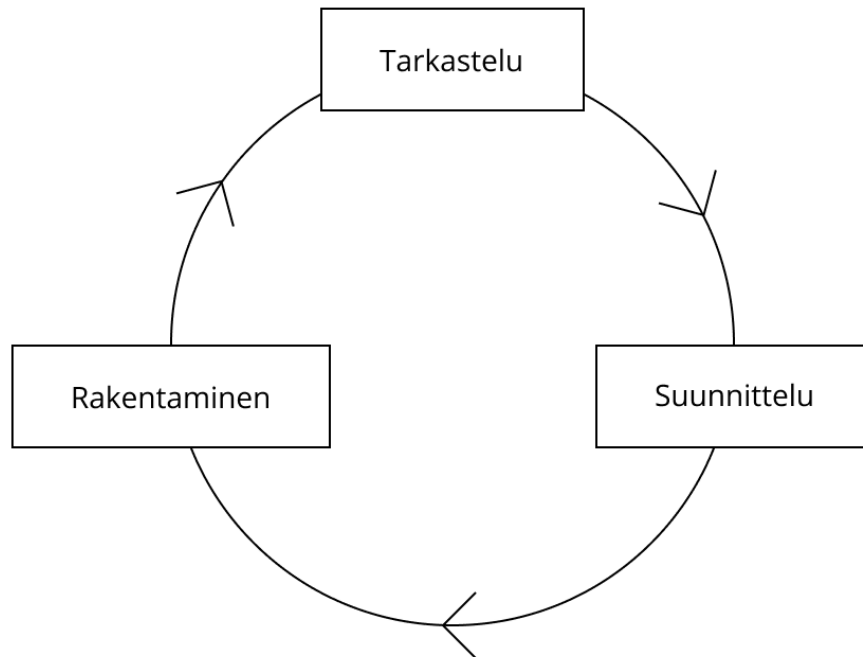
Perinteinen verkkosivun suunnittelu- ja tuotantoprosessi noudattaa vesiputousmallia (kuvio 2), joka jakaa projektin tehtäväkeskeisiin vaiheisiin (Marcotte 2015, 115). Prosessi alkaa konseptin kehittämisestä alkukartoituksessa tai palaverissa asiakkaan kanssa ja siitä seuraavasta tehtävänannosta projektitiimille. Tuotantotiimin visuaalinen suunnittelija tuottaa staattisen sommitelman tulevan verkkosivuston etusivun ulkoasusta graafisella ohjelmalla, kuten Adoben Illustrator tai Photoshop, mikä sitten hyväksytetään asiakkaalla. Näitä visuaalisia sommitelmia voi olla useita, ja ne edustavat erilaisia tyyllillisiä lähestymistapoja. Asiakas valitsee yhden ja antaa siitä palautetta. Tarvittavien muutosten jälkeen hyväksytty sommitelma annetaan eteenpäin front-end –toteuttajalle, jonka tehtävänä on rakentaa koodilla valittua sommitelmaa vastaava lopputulos selaimen. Vasta tämän vaiheen jälkeen verkkosivusto ja konsepti testataan. Projektin hyväksytetään asiakkaalla ja sivusto julkaistaan. (Carver 2015, 6-7.)



Kuvio 2. Vesiputousmalli.

Vesiputousmalli on epäkäytännöllinen, hidas ja kallis prosessi, jos projektin vaatimukset tai konsepti muuttuvat kesken kaiken. Asiakkaan omat tavoitteet voivat jäädä helposti taka-alalle, jos prosessin pääpaino on tuotannossa eikä itse konseptin suunnittelussa ja asiakkaan kanssa kommunikoinnissa. (Carver 2015, 6-7.) Tuotannon tavoitteena on toteuttaa relatiivisia määreitä hyväksikäyttävä responsiivinen kokonaisuus, joka esittää asiakkaan määrittelemän konseptin parhaalla mahdollisella tavalla. Siksi suunnitteluvaiheessa pikselimääreillä luotu sommitelma ei riitä, ja suunnittelun on tapahduttava selaimessa, joka on verkkosivuston luonnollinen ympäristö. (Marcotte 2015, 116-117.) Ongelmat tuotantomallissa tulevat esiin myös silloin, kun projektitiimin vastuulla on suunnitella useita sivuja samanaikaisesti. Visuaalisen suunnittelijan on mahdotonta tuottaa jokaisesta eri sivusta useita erilaisia näkymiä, kerätä niistä palautetta ja korjata niitä samalla, kun tuotantotiimi odottaa omaa vuoroansa. (Marcotte 2015, 116.)

Iteratiivinen ja yhteistyöhön perustuva suunnittelu (kuvio 3) on Marcotten suosittelema vaihtoehto vesiputousmallille. Siinä koko suunnittelutiimille esitetään visuaalinen sommitelma ja luonnos sivuston konseptista. Sen tarkoitus on toimia keskustelun aloittajana, eikä lopullisena ulkoasuna. Vapaamuotoisten kysymysten ja ideoiden jakamisen jälkeen suurimmat keskustelussa nousseet ongelmat korjataan ja kehittäjät aloittavat responsiivisen prototyypittämisen selaimessa. Suunnittelijat ja kehittäjät tekevät yhteistyötä prototyypin luomisessa ja valmis tuotos esitetään yhteisessä tapaamisessa, jossa se testataan erilaisilla laitteilla ja selaimilla. Prototyyppiä kehitetään eteenpäin esittelyssä nousseiden ajatusten, ideoiden ja ongelmien perusteella. (Marcotte 2015, 115-120.)



Kuvio 3. Iteratiivinen ja yhteistyöhön perustuva suunnitteluprosessi.

2.5 Staattisen sommitelman ongelmia

Staattinen sommitelma tulee terminä printtimainonnasta ja graafisesta suunnittelusta. Verkko-suunnittelussa se on graafisella ohjelmalla tuotettu visuaalinen kokonaisuus, joka esittää sivuston lopullista ulkoasua kauan ennen kuin itse sivusto aletaan rakentamaan selaimen. Front-end -toteuttajat rakentavat sivuston tämän sommitelman pohjalta. Se ei kuitenkaan toimi responsiivisessa verkko-suunnittelussa, koska sivuston lopullinen ulkoasu riippuu selaimesta, selaimen määrittelyistä asetuksista ja käyttäjän käyttämän laitteen koosta. (Carver 2015, 17.)

Ongelma täysien staattisten sommitelmien suunnittelemisessa on, että ne olettavat liikaa ja epäonnistuvat kommunikoimaan sivuston skaalautuvuutta. Ne esittävät vain yhtä staattista tilaa sivusta. Ne ovat myös kalliita ja hitaita tuottaa ja toteuttaa responsiivisesti, koska jokainen määre on mitattu pikseleinä kehi-

tysvaiheessa käytettyjen relatiivisten määreiden sijasta. Sommitelmat myös harvoin esittävät lopullista sivuston ulkoasua. Jos asiakkaalle tuotetaan useita sommitelmia, vain yksi niistä valitaan ja jäljelle jäävät heitetään pois. Sommitelman sisältö on suunnittelijan kontrolloimaa, mutta asiakkaan syöttäessä omia sisältöjään valmiille verkkosivulle sivuston ulkoasu voi muuttua dramaattisesti. Sen sijaan, että sivusto suunniteltaisiin noudattamaan tarkkoja suunnittelijan määrittelemiä visuaalisia sääntöjä olisi suunniteltava sivusto niin, että asiakas saa syötettyä sisältöä sivuille tahtonsa mukaan. (Carver 2015, 48-49.)

Pelkkään ulkoasuun keskittyminen suunnitteluvaiheessa riitelee myös muoto seuraa funktiota –ajattelun kanssa. Määritelmän mukaan suunnittelun tuloksen kauneus on puhtaan ja selkeän toiminnallisuuden tuotosta. Visuaaliset päätökset tulisi olla toissijaisia toiminnallisuuden suunnitteluun verrattuna. Tärkein kysymys, mitä suunnitteluvaiheessa tulisi kysyä visuaalisten valintojen sijasta olisi mitkä ominaisuudet ja toiminnallisuudet sivustossa tekevät siitä onnistuneen. (Butler ym. 2010, 106.)

Joissakin tapauksissa visuaalinen suunnittelija saattaa tuottaa rautalankamallin sivustosta ennen staattisen sommitelman toteuttamista. Rautalankamallit ovat yksinkertaisia hahmotelmia sivuston tärkeimmistä ominaisuuksista ja informaatiosta. Ne esittävät tulevan sivuston hierarkian, eikä niissä ole visuaalisia elementtejä, kuten värejä, typografiaa tai kuvia. Ne ovat tehokas keino keskittyä suunnitteluvaiheessa sivuston tarkoitukseen ja sisältöön visuaalisuuden sijaan. (Duckett 2014, 463.)

3 Prototyypittäminen

3.1 Prototyypittämisen perusteet

Prototyypittäminen on ideoiden, toiminnallisuuksien, monimuotoisten vaatimusten ja määritteiden testaamista, tutkimista ja esittämistä yksinkertaisten ja keskeneräisten suunnitelman osien avulla. Se antaa suunnittelijalle tärkeää tietoa

suunniteltavan tuotteen tai palvelun toiminnallisuudesta ja mahdollisuuden visualisoida, arvioida, oppia ja parantaa sitä eteenpäin ennen sen valmistamista oikeaksi, valmiiksi tuotteeksi. Prototyypittäminen voidaan jakaa kolmeen peruskategoriaan: konsepti-, kehittyvään ja poisheitettävään prototyypittämiseen. (Butler ym. 2010, 194.)

Konseptiprototyypittäminen on ensimmäisten alustavien ideoiden tutkimista nopeasti ja kustannustehokkaasti. Sen avulla suunnittelijan on helppo kommunikoida ideoitaan toisille, määritellä niiden vaatimuksia ja selvittää mahdollisia ongelmia. Sen tuotoksia voidaan myös arvioida. Esimerkkinä konseptiprototyypittämisestä on elokuvan kuvakäsikirjoitukset, luonnokset ja verkkosuunnittelussa rautalankamallit. (Butler ym. 2010, 194.) Poisheitettävät prototyypit ovat hyödyllisiä kun halutaan testata jotain yksittäistä osaa isommassa kokonaisuudessa. Kun haluttu informaatio on saatu selville, prototyypit heitetään pois ja suunnittelija keskittyy saadun tiedon perusteella isomman kokonaisuuden parantamiseen. (Butler ym. 2010, 194.)

Kehittyvää prototyyppiä käytetään, kun useat suunniteltavan tuotteen tai palvelun määritteet ja ominaisuudet ovat epävarmoja tai muuttuvia. Alkuperäistä prototyyppiä kehitetään, arvioidaan ja muokataan eteenpäin toistuvasti, kunnes siitä syntyy valmis tuote. Luotavat vaatimukset ja määritteet eivät koske tuotteen lopputulosta, vaan aina prototyypissä kehitettävää seuraavaa vaihetta. Tätä prototyypittämisen tapaa käytetään usein ohjelmistojen kehittämisessä. (Butler ym. 2010, 194.)

3.2 Prototyypittäminen responsiivisessa verkkosuunnittelussa

Responsiivisessa verkkosuunnittelussa prototyypittäminen toimii parhaiten itse selaimessa. Karujen prototyyppien rakentaminen kannattaa aloittaa heti, vaikka visuaaliset suunnitelmat eivät olisikaan valmiita tai hyväksytyjä. Tämä auttaa rakentamaan siltaa staattisia sommitelmia tuottavien visuaalisten suunnittelijoiden ja itse responsiivisia toteutuksia rakentavien front-end –toteuttajien välille. Prototyypivaiheessa toteuttajat toimivat myös suunnittelijoina. He testaavat

prototyypeillä alkuperäistä staattista konseptia ja tekevät ehdotuksia siitä, miltä sivusto voisi näyttää eri leveyksillä responsiivisesti. Tyylihdotukset esitellään suoraan selaimesta ja suunnittelijat voivat jatkaa työtänsä oikeaan suuntaan prototyyppien pohjalta. Tuotantoprosessi nopeutuu ja suunnittelijat ja toteuttajat saavat enemmän aikaa erilaisten tyyliuuntien ja toiminnallisuuksien suunnitteluun ja kehittämiseen. (Marcotte 2015, 117-119.)

Brad Frost painottaa front-end –toteuttajien olevan myös suunnittelijoita. HTML ja CSS ovat käyttöliittymien rakennuskieliä ja niillä määritellään sivuston lopullinen ulkoasu. Mitä nopeammin suunnittelu- ja prototyyppivaihe saadaan siirrettyä selaimen, sitä paremmat mahdollisuudet projektilla on onnistua. Jos asiakas näkee pelkän staattisen sommitelman tulevasta sivusta, kaikki sivuston todelliset ominaisuudet ja käyttäytymistavat responsiivisuudesta sivuston todelliseen suorituskäyttöön ja käyttöliittymän interaktioihin jäävät huomiotta. (Frost 2016, 119-120.)

Dan Mallin ja Jamie Kosuyn tapa prototyypittää selaimessa on nopea ja kustannustehokas. Jokaisen prototyypin on synnyttävä tunnin sisällä. Jos sen tekemiseen menee kauemmin, se ei ole enää prototyyppi vaan lähempänä valmista tuotetta. Ensimmäisen prototyypin pitää myös olla niin yksinkertainen, että kuka tahansa voi tehdä sen. Tämän prototyypin pohjalta rakennetaan seuraavia prototyyppijä, jotka ratkaisevat aina yhden yksittäisen ongelman konseptissa ja muuttuvat siten kehitysversioiden myötä monimutkaisemmiksi. (Mall 2015, 38.)

Prototyyppijä rakentaessa ei pidä keskittyä ulkoasuun tai niiden tarkkaan visuaalisuuteen. Tämä varmistaa ettei rakennettava prototyyppi ole vahingossakaan liian lähellä valmista tuotetta. Pahimmassa tapauksessa prototyyppi siirretään keskeneräisenä tuotantoon. Rakentamalla visuaalisesti keskeneräisiä prototyyppijä varmistetaan, että työtä pitää jatkaa. (Mall 2015, 39.) Prototyyppien hiominen täydellisyyteen ei ole kustannustehokasta (Friedman 2015, 51).

4 Modulaarisuus

4.1 Modulaarisuuden perusteet

Modulaarisuus on keino hallita monimutkaisia kokonaisuuksia ja järjestelmiä. Toiminnallisuuksiltaan samanlaiset ominaisuudet ryhmitellään yhteen ja niistä luodaan järjestelmän sisällä toimivia itsenäisiä kokonaisuuksia, moduuleita. Järjestelmä pysyy ehjänä, vaikka moduuleita vaihdettaisiin, päivitetäisiin tai lisättäisiin. (Butler ym. 2010, 160.)

Modulaarisuus on ollut olemassa kauan ennen tietokoneita ja ensimmäisiä tuotantoprosessien modularisointumisia voidaan havaita autoteollisuudessa. Teollistumisen myötä järjestelmien modulaarisuus ja Henry Fordin kehittämä liukuhihnana muuttivat autojen tuotantoprosessin pysyvästi. Autot olivat alun perin uniikkeja kokonaisuuksia, jotka tehtiin käsityönä ja olivat sen takia hyvin vaikeita korjata tai päivittää. Ford muutti tuotantoprosessia jakamalla auton erillisiin komponentteihin ja kokonaisuuksiin joita voitiin vaihtaa ilman, että se vaikutti muihin osiin. Tämän modulaarisen tuotantoprosessin ansiosta autoista tuli yhtenäisempiä, turvallisempia ja nopeampia tuottaa. (Frost 2016, 11-12.)

Modulaarisuus levisi pian muihin tuotannon aloihin. WWW:n kehittäjä Tim Berners-Lee kirjoittaa artikkelissaan Principles of Design (2013) modulaaristen järjestelmien suunnittelusta. Suunniteltaessa järjestelmää on sidottava yhteen samantapaiset ominaisuudet. Jos järjestelmää on päivitettävä on paljon todennäköisempää, että päivitystä tarvitsevia osia on vain yksi. Kokonaisen järjestelmän vaihtaminen ei silloin ole järkevää. Modulaarisuus mahdollistaa myös sen, että muut ihmiset voivat vaihtaa ja päivittää järjestelmän eri osia samaan aikaan. (Berners-Lee 2013.)

Järjestelmän yksittäiset moduulit tulisi suunnitella niin, että ne kommunikoivat muiden moduulien kanssa mahdollisimman yksinkertaisella tavalla. Niiden sisäinen monimutkaisuus tulisi piilottaa niin, että moduulien muodostaman järjestelmän hahmottamisesta ja kokoamisesta tulisi helpompaa. Modulaarinen suunnittelu kannustaa luovuuden, uusien moduulien luomisen ja kehittämisen lisäksi myös laadulliseen kilpailuun. Isoon ja monimutkaiseen järjestelmään

keskittymisen sijasta suunnittelijoiden on helpompaa keskittyä pienempiin kokonaisuuksiin. (Butler ym. 2010, 160.)

Modulaarisuudessa on useita etuja. Yksittäiset moduuleja voidaan kehittää eteenpäin ja niitä voidaan vaihtaa vaikuttamatta järjestelmän muihin moduuleihin. Jos moduulien väliset käyttöliittymät ovat tarpeeksi selkeitä, kehittäjä voi suunnitella olemassaolevan moduulin uudelleen ilman syvempää ymmärrystä koko järjestelmästä. (Berners-Lee 2013.) Hyvin ja puhtaasti suunniteltua moduulia voidaan käyttää uudelleen toisessa järjestelmässä (Berners-Lee 2012).

Tavoitteena on rakentaa mahdollisimman puhtaita ja yksinkertaisia järjestelmiä valmiilla käyttöliittymillä, jotka pystyvät toimimaan uusien järjestelmien kanssa joita ei välttämättä ole vielä edes kehitetty (Berners-Lee, 2012). Tämä ajattelu-tapa tukee responsiivista verkkosuunnittelua hyvin. Näyttöpäätelaitteiden kasvu ei ole hidastumassa, joten on osattava suunnitella verkkosivut toimimaan laitteilla joita ei ole vielä olemassakaan (Frost 2016, 12).

4.2 Modulaarisuus verkkosuunnittelussa

Internet ja verkkosuunnittelu ovat perineet sanastonsa printatun median maailmasta. Alun perin Internet oli akateemisissa piireissä tiedostojen jakamiseen tarkoitettu väline, ja sen takia verkkosuunnittelussa sivun konsepti on hyvin vahva. Nimeämiskäytännöillä on ollut suuri vaikutus siihen, miten käsitämme Internetin ja verkkosivut. Internetin alkuaikoina yritysten päätavoitteena oli siirtää printtimateriaali samassa formaatissa sähköiseksi verkkoon, sillä heidän oli helppo ajatella verkkosivuja printtimaailman jatkeena. Ajattelemalla verkkosivuja samanlaisina kuin kirjan tai lehden sivut oletamme, että ne ovat itsenäisiä, eristettyjä ja mitattavia kokonaisuuksia, vaikka oikeasti verkkosivusto on alati muuttuva, interaktiivinen ja toisista sivuista riippuvainen kokonaisuus, joka voi olla hyvinkin monimutkainen järjestelmä. (Frost 2016, 9-11.)

Verkkosivustojen toteuttaminen tulisi siis alkaa sivumäärän määrittämisen sijaan suunnittelemalla niiden vaatimukset, toiminnallisuus sekä käytetyt elemen-

tit, mikä voi olla haaste. Kaikkia onnistuneita verkkosuunnitteluprojekteja yhdistää niiden modulaarisuus. (Frost 2016, 11.) Yhtenäisten käyttöliittymien suunnitteluun ja tuotantoprosessien nopeuttamiseen on kehitetty useita modulaarisia ratkaisuja. Ne voivat olla elementtikirjastoja, viitekehysjä, nimeämiskäytäntöjä ja vaihtoehtoja staattisille sommitelmille. Rikkomalla suunnitteluprosessia pienempiin osiin säästetään aikaa ja vältetään esittämästä asiakkaalle epärealistisiä tavoitteita. (Frost 2016, 12-19.)

Modulaarisia lähestymistapoja kirjoittaa koodia ovat esimerkiksi OOCSS (Object Oriented CSS), joka on Nicole Sullivanin luoma tapa tuottaa nopeaa, päivitettävää ja standardoitua koodia. Se on keino rakentaa elementtejä, jotka ovat verkkosivustolla toistuvia visuaalisia kaavoja. Elementit voidaan erotella itsenäisiin HTML, CSS ja Javascript –tiedostoihin. OOCSS rakentuu kahden pääsäännön ympärille. Elementin rakenne ja ulkoasu on oltava erillään toisistaan, jolloin niitä voidaan vaihtaa ja päivittää nopeasti ja vaivattomasti. Elementin tulee myös toimia ja näyttää samalta riippumatta sen sijainnista sivustolla. (Sullivan 2013.)

Jonathan Snookin SMACC (Scalable and Modular Architecture for CSS) on enemmän tyyliohje kuin tiukka viitekehys modulaariseen verkkosuunnitteluun. Se jakaa kirjoitetun CSS:n kategorioihin, joiden avulla koodista voidaan löytää toistuvia kaavoja. SMACC:n tavoitteena on vähentää kirjoitetun koodin määrää, parantaa sen päivitettävyyttä ja ymmärrettävyyttä ja luoda samalla modulaarisuutta verkkosivujen tuotantoprosessiin. (Snook, 2017.)

Yandexin kehittämä BEM (block, element, modifier) on nimeämiskäytäntö, jonka avulla tyylien nimet kertovat enemmän niitä käyttävien elementtien sisällöstä ja toiminnallisuudesta. Sen etuja ovat uudelleenkäytettävyys, modulaarisuus ja ymmärrettävyys. (Starkov & Stepanova 2017.)

Frameworkit ovat valmiita suunnittelujärjestelmiä, koodikirjastoja ja perustoja verkkosivuille. Ne tarjoavat eri selaimilla ja laitteilla testatun koodin yleisiin verkkosivuston tarpeisiin, kuten asetteluun ja responsiivisuuteen. Usein ne tarjoavat myös valmiita käyttöliittymän elementtejä, kuten lomakkeita ja erilaisia painikkeita. Käyttämällä näitä perustoja toteuttajan ei tarvitse jokaisessa projektissa kir-

joittaa koodia alusta asti itse. (Duckett 2014, 391.) Suurin etu frameworkien käytössä on niiden nopeus. Ne mahdollistavat nopean prototyypittämisen ja niiden ympärillä on vahva yhteisö joka osaa auttaa ongelmatilanteissa. (Frost 2016, 20.)

Tällaiset koodikirjastot sisältävät kuitenkin usein paljon ylimääräistä koodia, jota kehittäjä ei tule koskaan käyttämään (Duckett 2014, 391). Maailman suosituin framework Bootstrap tarjoaa mahdollisuuden luoda kustomoituja koodipohjia, jolloin koodikirjastoon lisätään vain kehittäjän valitsemat ominaisuudet (Park 2013). Koodikirjastojen mukana tulevat valmiit nimeämiskäytännöt rikkovat sivun ja kirjoitetun koodin yhtenäisyyttä jolloin esimerkiksi BEM:in, OOCSS:n ja SMACSS:n käyttäminen vaikeutuu. Jos näitä koodikirjastoja käytetään sellaisenaan tuotetut verkkosivut alkavat nopeasti muistuttamaan toisiaan. Yrityksen yksilöllinen brändi voi menettää tehokkuutensa, jos sivustoja ei kustomoida. Koodikirjastot toimivat vahvana pohjana tarjoten valmiita ratkaisuja responsiivisuuteen ja muihin yleisiin toiminnallisiin ongelmiin, mutta niiden päälle on osattava rakentaa muistettavia käyttäjäkokemuksia. (Frost 2016, 22.)

4.3 Modulaarisuus visuaalisessa suunnittelussa

Modulaariset lähestymistavat ovat löytäneet tiensä myös visuaalisten suunnittelijoiden työskentelyprosesseihin. Suunnittelijat ovat huomanneet staattisten sommitelmien tuottamisen huonot puolet ja kehittäneet modulaarisempia ja tehokkaampia ratkaisuja niiden tilalle. Tarkkojen ulkoasujen sijasta suunnittelijat keskittyvät sivuston tunnelmaan. Tunnelman määrittely on tärkeää projektin alkuvaiheessa. (Frost 2016, 17.)

Dan Mall kertoo kokemuksistaan ja näkemyksistään nykyajan verkkosuunnittelijan työprosessista ja sen parantamisesta. Mallin mukaan staattisten sommitelmien luomisen sijasta voidaan luoda visuaalisia kirjastoja, joiden tarkoitus on auttaa löytämään oikea tyyli suunta projektille. Niiden luominen on paljon nopeampaa ja kustannustehokkaampaa kuin staattisten sommitelmien ja niillä voidaan esittää asiakkaalle paljon tarkempia kysymyksiä. Sivuston tyyliä etsies-

sään Mall luo visuaalisia kirjastoja ottamalla kuvia erilaisista olemassaolevista verkkosivuista, jotka edustavat erilaisia tyyliisuuntia. Hän editoi kuviin mukaan asiakkaan logon ja liittää viereen kysymyksen, johon hän toivoo saavansa vastauksen. Sen sijaan, että hän viettäisi päiviä erilaisten staattisten sommitelmien luomiseen hän voi saada tarvitsemansa vastaukset saman päivän aikana. (Mall 2015, 27-28.)

Tyylitiilet ovat Samantha Warrenin kehittämä vaihtoehto staattisille sommitelmille. Ne ovat sivuston tunnelmaan keskittyviä kollaaseja, joissa yhdistetään typografiaa, värejä ja käyttöliittymän elementtejä. Niitä käytetään, kun staattiset sommitelmat ovat liian laajoja ja perinteiset kuvakollaasit liian abstrakteja. Niiden pohjimmainen tarkoitus on varmistaa, että visuaalinen suunnittelija on ymmärtänyt asiakkaan tavoitteet ja toiveet. Tyylitiilen luominen aloitetaan suunnittelupalaverissa asiakkaan kanssa, jossa kerätään asiakkaan toistamat adjektiivit ja mielikuvat. Niistä koostetaan visuaalisia vastikkeita, ja niitä hiotaan, kunnes suunnittelija ja asiakas ovat samalla sivulla. (Warren 2017.)

Elementtikollaasit ovat lähestymistapa, jonka Mall kehitti Warrenin tyylitiilien pohjalta. Ne ovat hahmotelmia sivuston yleisestä tyylistä ja tunnelmasta asiakkaan toiveiden ja kommenttien pohjalta, mutta ne sisältävät enemmän verkkosivuelementtejä ja niiden interaktioita värien ja typografian lisäksi. Elementtikollaasien luominen ei vaadi koko sivuston ulkoasun suunnittelua, vaan suunnittelija voi keskittyä hahmottelemaan ja luonnostelevaan mielestään tärkeimpiä sivuston osia. (Mall 2015, 29-31.)

Jos staattiset sommitelmat ovat asiakkaalle tutumpia kuin elementtikollaasit, Mall suosittelee niiden kasaamista horisontaaliselle kanvakselle jolloin ne muistuttavat vähemmän perinteistä verkkosivua eivätkä hämää asiakasta luulemaan työtä keskeneräiseksi sommitelmaksi. Horisontaalisuus auttaa asiakasta ymmärtämään kollaasin perimmäisen tarkoituksen. (Mall 2015, 34.) Loppujen lopuksi responsiivinen verkkosuunnittelu on paljon enemmän kuin media queryita, sommitelmia ja rikkiäisten asettelujen korjaamista. Se on paljon monimuotoisempaa ja kattaa kaiken typografiasta sivuston latausnopeuteen. (Friedman 2015, 50.)

4.4 Suunnittelujärjestelmät

Suunnittelujärjestelmät ovat järjestelmiä, joiden avulla sivustojen tuottamisen prosessi modularisoituu, yhtenäistyy ja nopeutuu. Suunnittelujärjestelmä kokoaa erilaisia työtapoja yhteen ja siihen voi kuulua esimerkiksi nimeämiskäytäntö, koodikirjasto ja erilaisia visuaalisia lähestymistapoja. Suunnittelujärjestelmien kulmakivinä toimivat tyyliohjeet. Tyyliohjeet voivat olla dokumentaatioita yrityksen käyttämistä äänensävyistä erilaisissa viestintätilanteissa, brändin logosta ja muista määritteistä, käyttöliittymän elementeistä ja muista järjestelmän osaluista. (Frost 2016, 23.)

Brändin tyyliohjeeseen kuuluu yrityksen materiaalit ja ohjeet ja rajoitukset niiden käyttämisestä eri konteksteissa. Se voi sisältää erilaisia valmiiksi muotoiltuja tiedosto- ja esityspohjia, joita käytetään kaikissa yritykseen liittyvissä viestintätilanteissa. Brändiohjeita toteutetaan yhä useammin verkkoon, jossa ne ovat useampien käyttäjien saavutettavissa. (Frost 2016, 24.) Tyyli suunnan ohje on abstraktimpi kuin brändiohje. Se viestii konseptin takana olevaa filosofiaa, tunnelmaa ja tavoitteita. Tyyli suunnat voivat kehittyä ajan kuluessa, joten ohjeistusta päivitetään ja siitä kehitetään uusia versioita. (Frost 2016, 25.)

Viestinnän äänensävyä voidaan tehdä tyyliohje, joka määrittelee miten yrityksen useat viestijät voivat puhua eri alustoilla yhtenäisesti. Sen avulla yrityksen äänestä tulee tunnistettava ja viestijät osaavat tunnistaa tilanteet, jolloin äänensävyä on vaihdettava. Kirjoitetun sisällön tyyliohje on samantyyppinen kuin äänensävyä tuotettu tyyliohje, mutta sen avulla varmistetaan julkaistun tekstisisällön yhtenäisyys. (Frost 2016, 26-27.)

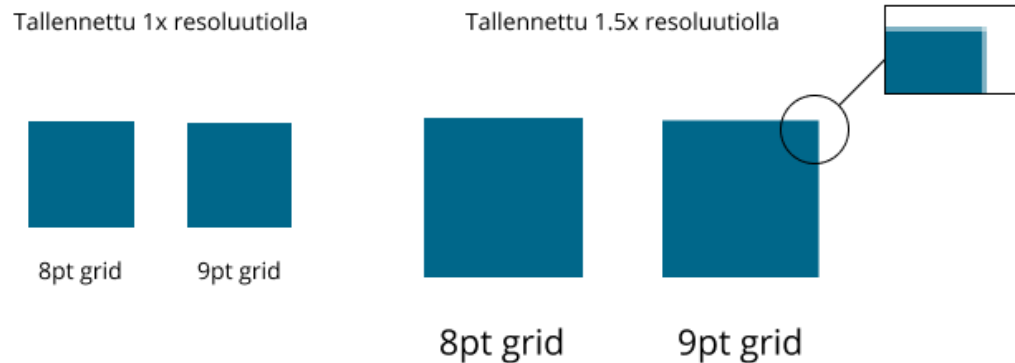
Koodin tyyliohjeet ovat ohjeita yhtenäisen ja laadukkaan koodin kirjoittamiseen. Ne auttavat toteuttajia kirjoittamaan skaalautuvaa ja modulaarisempaa koodia, jota on myöhemmin mahdollista uudelleenkäyttää ja joka on muille toteuttajille helposti ymmärrettävää ja luettavaa. (Frost 2016, 28.) Tyyliohjeet auttavat saavuttamaan yhtenäisyyttä eri alustoissa ja tuotteissa. Yhtenäisyys on tärkeää

käyttäjän luottamuksen ansaitsemiseksi. Tyyliohjeiden ansiosta prosessien osia ja sisältöä on helpompi modularisoida ja suunnittelijat voivat aina palata tyyliohjeeseen tarvittaessa. (Frost 2016, 30.)

Elementtikirjastot ovat tyyliohjeisiin verrattuna modulaarisempia. Kirjoitettujen ohjeiden sijasta ne tarjoavat valmiita kopioitavia elementtejä, joita voidaan käyttää erilaisissa projekteissa. Elementtikirjastoja kutsutaan erilaisilla nimillä, joita voi olla esimerkiksi front-end tyyliohje, käyttöliittymäkirjasto ja kaavakirjastot. (Frost 2016, 29.) Ne kommunikoivat sivuston ja yrityksen tyylikieltä hyvin konkreettisesti (Frost 2016, 31). Elementtikirjastojen käyttäminen tekee sivustoista yhtenäisiä, nopeuttavat tuotantoprosessia, mahdollistavat yhteisen kielen syntymisen suunnittelijoiden ja toteuttajien välille sekä ovat päivitettäviä ja muokattavia (Frost 2016, 65).

Käyttöliittymäkirjastossa kerätään kaikki verkkosivuston käyttöliittymän elementit ja esitellään ne yhdellä sivulla. Tämä dokumentaatio paljastaa ovatko sivustolla käytetyt elementit yhtenäisiä vai ei. Sen kokoaminen on myös hyvä opetus projektitiimin jäsenille. He oppivat yhtenäisyyden tärkeyden konkreettisella tavalla kerätessään yhteen eri käyttöliittymäelementtejä. Ne ovat erittäin tehokkaita esittämään, miksi yhtenäiset käyttöliittymät ovat tärkeitä. Kuka tahansa voi käsittää niiden tarkoituksen nopealla vilkaisulla. (Frost 2016, 99-106.)

Visuaalisiin suunnittelujärjestelmiin kuuluva 8-point grid on suunnittelujärjestelmä, joka keskittyy käyttöliittymän visuaaliseen yhtenäisyyteen eri alustoilla. Se perustuu elementtien mittojen ja välien määrittelyyn kahdeksalla jaollisilla tasaluvuilla. Luku kahdeksan on valittu muista tasaluvuista kahdesta syystä. Eri-laisten laitteiden koot ja resoluutiot vaikuttavat siihen, miten näytettävien pikselien määrä kerrotaan ja käyttämällä tasalukuja varmistetaan, että pikselit ovat aina reunoiltaan tarkkoja (kuvio 4) ja yleisimmät laitteiden näyttökoot ovat kahdeksalla jaollisia. (Dahl 2016.)



Kuvio 4. Tasaluvuilla rakennetut elementit ovat tarkkoja resoluutiosta riippumatta.

Käyttämällä 8-point gridiä verkkosivutuotannoissa suunnittelijat ja toteuttajat jatkavat yhtenäisen kielen, mikä helpottaa ja nopeuttaa tuotantoprosessia huomattavasti. Toteutetut sivustot näyttävät yhtenäisemmiltä laitteiden resoluutioista riippumatta, ja käyttäjät saavat paremman käyttökokemuksen. (Dahl 2016.) Sivuston visuaalinen yhtenäisyys lisää myös käyttäjien luottamusta sivustoa kohtaan (Butler ym. 2010, 56).

Suunnittelujärjestelmien suurimmat kompastuskivet ovat usein yrityksessä valitseva mentaliteetti siitä, ettei niiden rakentamiseen ole aikaa tai rahaa. Niiden ylläpitoon tarvitaan suunnittelijoiden ja toteuttajien työprosessiin tiiviisti integroitu strategia. Jos suunnittelujärjestelmät ja tuotetut tyyliohjeet jäävät projektien tuotantoprosessien ulkopuolelle, niistä ei ole hyötyä kenellekään. (Forst 2016, 35.) Hyvin järjestelty, suunniteltu ja informatiivinen suunnittelujärjestelmä tai elementtikirjasto tarjoaa sen käyttäjälle paljon enemmän kuin pelkkiä käyttöliittymän elementtejä. Se tarjoaa kontekstin ja syitä miksi elementtiä käytetään määritellyllä tavalla. Modulaarisuuden kasvaessa verkkosuunnittelussa parempia ja käytettävämpiä suunnittelujärjestelmiä tarvitaan enemmän kuin koskaan. (Frost 2016, 37.)

5 Atominen suunnittelu

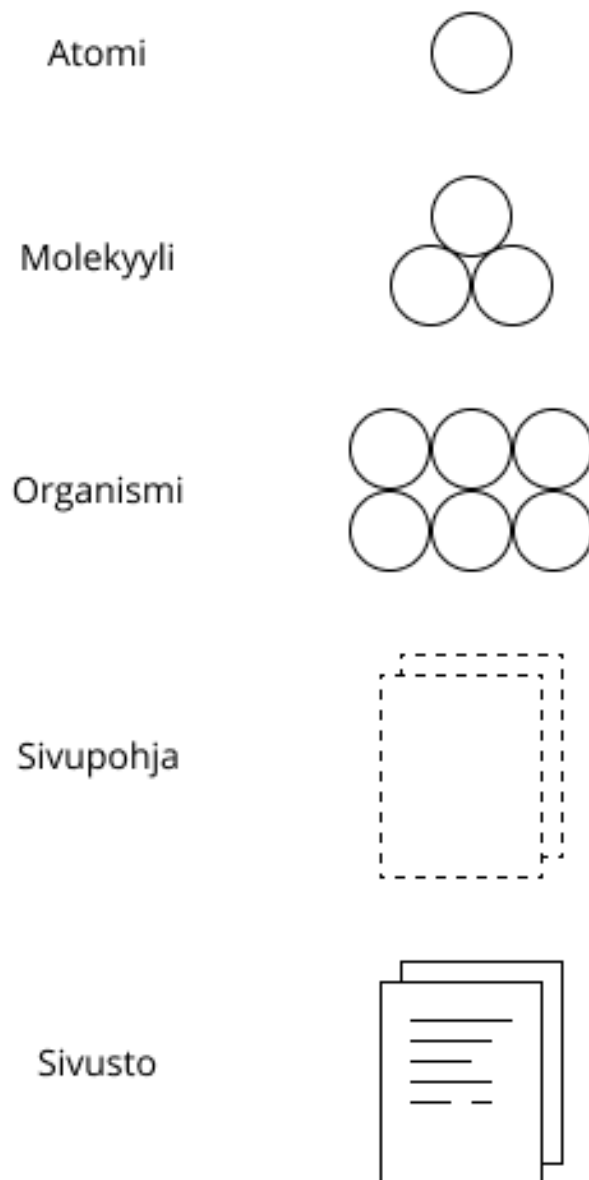
5.1 Atomisen suunnittelun perusteet

Atominen suunnittelu on Brad Frostin kehittämä modulaarinen suunnittelujärjestelmä käyttöliittymien rakentamiseen. Frost sai inspiraation atomiseen suunnitteluun luonnossa tapahtuvista kemiallisista reaktioista, joissa yksinkertaiset atomit yhdistyvät keskenään molekyyleiksi, joista monimutkaisemmat organismit koostuvat. Muuntamalla tämän luonnollisen prosessin koskemaan käyttöliittymien kehittämistä Frost päätyi suunnittelujärjestelmään, joka koostuu viidestä eri vaiheesta, joita yhdistelemällä lopputuloksena syntyy valmis ja modulaarinen verkkosivusto. (Frost 2016, 40-42.)

Atominen suunnittelu on modulaaristen suunnittelujärjestelmien rakentamista erillisten, yksittäisten sivujen sijaan. Suunnitteluprosessi alkaa komponenttien rakentamisella. Yhdistelemällä rakennettuja komponentteja keskenään saadaan aikaiseksi verkkosivuston eri elementtejä. Sivusto syntyy näiden elementtien rakentamisen ja yhdistelemisen tuloksena. (Mall 2015, 42.)

5.2 Atomisen suunnittelujärjestelmän rakenne

Viisi eri suunnittelun vaihetta atomisessa suunnittelussa ovat atomit, molekyylit, organismit, sivupohjat ja sivustot (kuvio 5). Atomit sisältävät käyttöliittymän peruselementit, jotka pysyvät toiminnallisuudeltaan ehjinä vaikka ne ottaisi pois isommasta kontekstista. Näitä elementtejä ovat esimerkiksi painikkeet ja lomakkeen osat. Atomeihin kuuluvat myös sivustolle määritellyt perustyylit, kuten fonttien koot ja rivinvälit. (Frost 2016, 42-43.)



Kuvio 5. Atominen suunnittelujärjestelmä.

Yksittäisillä atomeilla ei ole toimivaa tarkoitusta tai merkitystä, mutta yhdistelemällä niitä eri tavoin voidaan saada aikaiseksi eri konteksteissa toimivia modulaarisia molekyyliä. Molekyylit ovat yhteen ryhmiteltyjä käyttöliittymän elementtejä jotka toimivat omina kokonaisuuksinaan. Ne muodostuvat yksittäisistä atomeista ja esimerkiksi painike ja lomakkeen kenttä voivat muodostaa yhdessä toimivan hakukentän, joka voidaan lisätä sinne missä sitä tarvitaan. Yksinker-

taisten molekyylien luominen helpottaa niiden testaamista ja tekee niistä uudelleenkäytettäviä. Ne pysyvät myös helpommin yhtenäisinä. (Frost 2016, 44-45.)

Organismit yhdistävät molekyyliä, atomeita ja toisia organismeja monimutkaisemmiksi kokonaisuuksiksi. Sivustolla organismit ovat toisistaan helposti erottuvia ja niiden vaihtaminen, päivittäminen tai poistaminen ei vaikuta sivuston muihin organismeihin. Organismien sisällä voi olla rajaton määrä erilaisia atomeita, molekyyliä ja organismeja. Niitä voidaan toistaa useita kertoja, jolloin toteuttajan on mahdollista luoda esimerkiksi erilaisia listauksia. Nämä organismit toimivat missä vain ja niiden rakennetta voidaan päivittää tarpeen vaatiessa. (Frost 2016, 46-47.)

Sivupohjat ovat sivukohtaisia malleja tulevan sivuston asettelusta. Se antaa molekyylielle ja organismeille kontekstin, jossa niiden on toimittava. Sivupohjasta nähdään miten organismit toimivat yhdessä. Aivan kuten rautalankamallit sivupohjat keskittyvät rakenteeseen ja hierarkiaan valmiin sisällön sijasta. Rakenteen määrittelyllä voidaan varmistaa, että tuleva sivusto pitää sisällään asiakkaan tuottamaa sisältöä ilman ongelmia. Sivupohjassa esitetään rajoituksia ja sääntöjä oikeissa konteksteissa, joiden tarkoitus on auttaa asiakasta tuottamaan laadukasta sisältöä. Nämä säännöt voivat koskea kuvien kokoa, tekstien kirjainmääriä tai videoiden tiedostomuotoja ja resoluutiota. (Frost 2016, 49-51.)

Sivustot ovat valmiita kokonaisuuksia, jotka esittävät käyttöliittymän ulkoasun toteuttajan lisäämillä esimerkkisisällöillä. Ne ovat visuaalisesti lähimpänä staattisia sommitelmia, mutta toimivat selaimessa responsiivisesti ja ovat perusteellisesti testattuja ja valmiita tuotantoon. Sivuilla voidaan testata suunnittelujärjestelmän tehokkuutta ja toimivuutta. Jos sivustovaiheessa jokin organismi ei toimi tarkoitetulla tavalla, suunnittelija voi palata korjaamaan sitä vaikuttamatta muihin sivuston elementteihin. Sivustolla näkyvä sisältö voi olla huomattavan erilaista riippuen käyttäjän asetuksista, katseluoikeuksista ja sijainnista. Käyttäjä ei myöskään välttämättä käytä sivustoa suunnittelijan haluamalla tavalla. (Frost 2016, 52-55.) Jotta suunnittelujärjestelmät pystyvät vastaamaan näihin muutuksiin, on ne rakennettava ottamaan huomioon mahdollisimman monta erilaista käyttäjäskenaariota (Frost 2016, 79).

Atominen suunnittelujärjestelmä mahdollistaa sivustojen modulaarisen tuottamisen ja muokkaamisen. Tuotantoprosessi on erittäin tehokas, ja se keskittyy pienempien elementtien laatuun ja toiminnallisuuteen isommassa kokonaisuudessa. Atomisen suunnittelutapa ei ole lineaarinen, kuten tuotantoprosessin vesiputousmalli. Jokainen prosessin osa on yhteydessä toisiinsa ja suunnittelujärjestelmä antaa tuotantotiimille ajattelumallin, jonka avulla suunnittelijoiden ja toteuttajien tarvitsee keskittyä vain yhteen asiaan kerrallaan. (Frost 2016, 56-57.) Pattern Lab on Brad Frostin, Dave Olsenin ja Brian Muenzenmeyerin luoma atomisten suunnittelujärjestelmien luomiseen tarkoitettu työkalu. Se on sivugeneraattori, jonka avulla voidaan myös tallentaa ja dokumentoida sivuston elementtejä ja käyttää niitä elementtikirjaston lailla verkkosivuprojekteissa. (Frost 2016, 65-66.)

6 Elementtikirjaston konsepti ja prototyyppi

6.1 Opinnäytetyön tavoite

Toiminnallisen opinnäytetyön tavoitteena on tuottaa atFlow Oy:lle responsiivisten verkkosivuteemojen suunnitteluun ja rakentamiseen tarkoitettun elementtikirjaston konsepti ja prototyyppi. Prototyyppi käyttää hyväkseen Brad Frostin atomista suunnittelujärjestelmää ja 8-point gridiä. Kirjaston tarkoitus on olla aktiivisesti ja helposti päivitettävä kokonaisuus, joka ratkaisee verkkosivuteemojen tuotantoprosessin vesiputousmalliin liittyviä kustannustehokkuuden ongelmia. Tärkeimpänä tavoitteena on muuttaa verkkosivustojen tuotantoprosessi modulaariseksi elementtikirjaston avulla.

6.2 Elementtikirjaston tavoite

Elementtikirjaston tavoite on olla päivitettävä kirjasto, johon verkkosuunnittelijat voivat lisätä käyttövalmiita elementtejä. Sen tarkoitus on olla työkalu, joka aut-

taa tuotantoprosessia muuttamaan modulaariseksi ja kannustamaan toteuttajia rakentamaan laadukkaampia elementtejä. Elementti voi olla mikä tahansa itsenäisesti toimiva verkkosivulta löytyvä käyttöliittymän osa, kuten otsikko, artikkeli tai navigaatio. Elementtikirjastoon lisätään testattuja ja laadukkaasti tuotettuja käyttöliittymäelementtejä, joita voidaan yhdistellä ja käyttää erilaisissa verkkosivuprojekteissa. Elementtikirjaston käyttö tulee vähentämään nykyisen tuotantoprosessin ongelmia, joissa vanhoja yksittäisille asiakkaille tuotettuja ja personoituja sivupohjia kopioidaan ja käytetään uudelleen uusissa projekteissa. Vanhat pohjat saattavat sisältää responsiivisia ongelmia eikä niiden käyttämä koodi ole yhtenäistä. On kustannustehokkaampaa rakentaa verkkosivu uudelleenkäytettävistä ja modulaarisista elementeistä kuin korjata toistuvasti yhtä isoa kokonaisuutta.

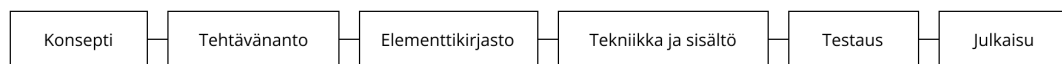
Elementtikirjaston prototyyppi toteutetaan yrityksen sisäisille verkkosivuille. Se käyttää samoja CSS –tiedostoja ja sääntöjä kuin sen avulla rakennettavat elementit ja teemat. Kirjaston tarkoituksena on, että verkkosivujen tuottajat ja suunnittelijat voivat rakentaa ja jakaa omaa tietopohjaansa eteenpäin luoden samalla laadukkaita tuotteita helpommin ja nopeammin. Laadukkaat verkkosivuteemat vähentävät myös asiakkailta tulevia korjauspyyntöjä.

Elementtikirjastoon tuotetut elementit kootaan ja lajitellaan omiin kategorioihin, josta verkkosivuteemojen suunnittelijat ja toteuttajat voivat hakusanoilla etsiä ja valita tarvitsemansa elementit. Uusien tekijöiden kouluttamiseen kuluu vähemmän aikaa ja heidän omatoimisuutensa kasvaa. Kirjaston ja elementtien päivitettävyys takaa yrityksen osaamisen ja hiljaisen tiedon dokumentoinnin ja sujuvan siirtämisen eteenpäin. Ideaalitulanteessa elementtikirjastoon luodut uudelleenkäytettävät elementit tarjoavat suunnittelijoille enemmän aikaa kehitystyöhön.

Elementtikirjasto on kehittyvä työkalu eikä sen tarkoitus ole olla valmis tuote. Sen kehittäminen on yrityksen vastuulla. Jotta elementtikirjastosta olisi hyötyä, se on liitettävä tiukasti mukaan projektitiimin tuotantoprosesseihin.

6.3 Elementtikirjaston konsepti

Käyttämällä elementtikirjastoa verkkosivustojen tuotantoprosessi muuttuu modulaariseksi. Tuotantoprosessi alkaa konseptoinnista ja tehtävänannosta. Seuraavassa vaiheessa ei kuitenkaan tuoteta staattista sommitelmaa kuten vesiputousmallissa, vaan sivuston rakentaminen aloitetaan suoraan selaimessa elementtikirjaston avulla (kuvio 6). Siinä missä vesiputousmallissa front-end –toteutus aloitettaisiin vasta staattisen sommitelman tuottamisen jälkeen, modulaarisessa tuotantoprosessissa suurin osa front-end –toteutuksesta on jo tehty. Aikaa jää erilaisten toiminnallisuuksien vaatimien tekniikoiden tekemiselle ja sisällön tuottamiselle. Testaukseen kuluu huomattavasti vähemmän aikaa, sillä sivustolla käytetyt elementit ovat jo testattuja ja responsiiviset ongelmat korjattuja.



Kuvio 6. Modulaarinen tuotantoprosessi.

Elementtikirjastolle voidaan määritellä kolme tärkeintä ominaisuutta. Kirjastoon ladataan testattuja ja toimivia elementtejä. Näitä elementtejä voidaan luokitella kategorioihin ja hakea hakusanoilla. Valitut elementit ladataan käytettäväksi erilaisissa projekteissa. Kirjastoon ladatut elementit pysyvät sisällöltään ja tyyleitään alkuperäisessä muodossa, eikä muissa projekteissa tapahtuneet muutokset vaikuta niihin. Elementtejä voidaan lisätä, päivittää ja poistaa kirjastosta.

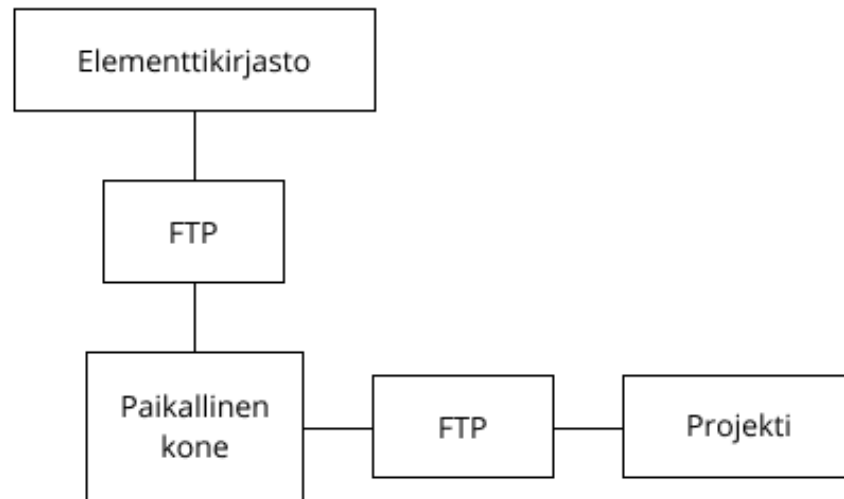
Elementtikirjaston ominaisuudet jaetaan eri kehitysversioihin ja niille annetaan tavoitteet. Näin varmistetaan, että elementtikirjastoa rakentaessa keskitytään olennaiseen ja ominaisuuksien toteuttaminen ei ole liian monimutkaista. Kehitysversioiden käyttö antaa konseptille rakennetta ja on paljon todennäköisempää, että asetetut tavoitteet toteutuvat.

Kehitysversioiden avulla voidaan keskittyä yhteen toiminnallisuuteen kerrallaan atomisen, modulaarisen ja iteratiivisen suunnittelutavan mukaisesti. Jokainen kehitysversio käydään läpi ottaen huomioon tuotantoprosessin aikana nousseet kommentit ja kehitysideat. Kehitysideoita ja versioita voidaan myös yhdistellä uusiksi kehitysversioiksi palautteen perusteella.

Elementtikirjaston käyttäminen ja kehittäminen edistää yrityksen brändin tunnettuutta ja erilaisten mielikuvien syntyä. Se auttaa tyylikielen syntymisessä ja sen avulla on yksinkertaista ja nopeaa toteuttaa erilaisia tyyliohjeita myös asiakkaille. Käyttämällä elementtikirjastoa yritys voi luoda tuotteilleen erilaisia laatuun liittyviä standardeja, joita voidaan käyttää markkinointitarkoituksissa.

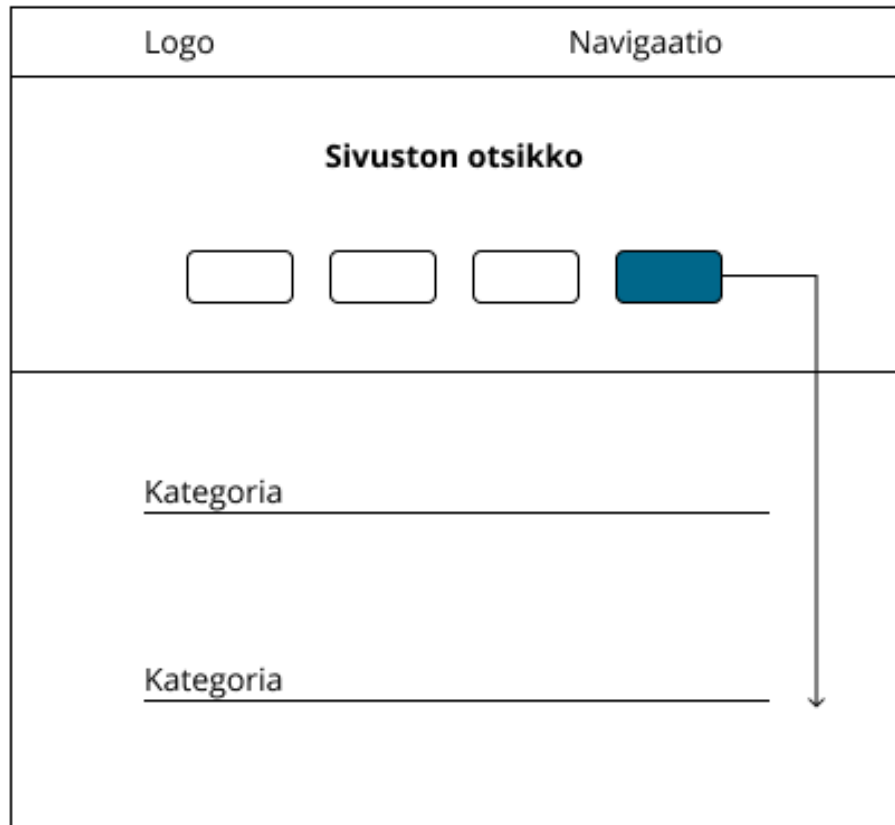
6.4 Elementtikirjaston konseptin ensimmäinen kehitysversio

Elementtikirjaston ensimmäinen kehitysversio sisältää visuaalisen ja responsiivisen käyttöliittymän toimivilla esimerkkisisällöillä. Se toteutetaan yrityksen sisäisille verkkosivuille ja toimii pohjana tulevaisuudessa toteutettavalle tekniikalle. Tämä kehitysversio kuuluu opinnäytetyön toiminnalliseen osuuteen. Elementtikirjaston ensimmäinen kehitysversio ei sisällä syvempää toiminnallisuutta. Elementtien lataaminen ja lisääminen tapahtuu erillisen tiedostojen siirtoon tarkoitetun FTP-ohjelman kautta (kuvio 7).

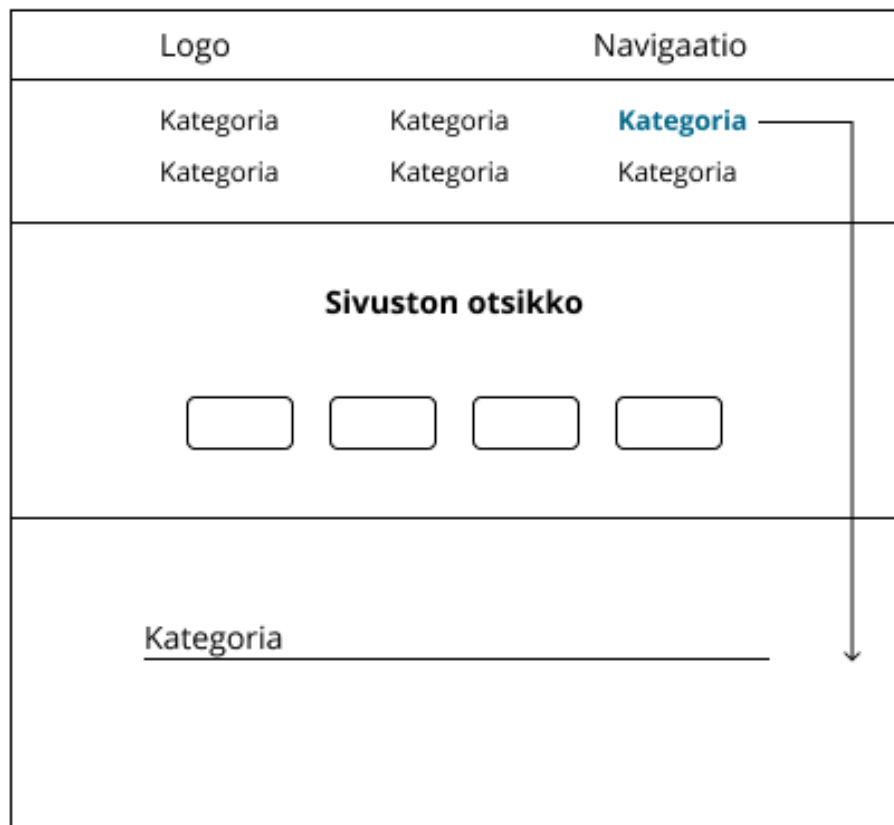


Kuvio 7. Elementtikirjaston toiminnallisuus ensimmäisessä kehitysversiossa.

Elementtien hakeminen elementtikirjastosta tapahtuu kategoriapainikkeiden ja navigaatiosta löytyvien kategorialinkkien avulla. Kaikki kategoriat ja elementit ovat listattu etusivulle. Kategoriapainikkeista (kuvio 8) ja navigaatiosta (kuvio 9) käyttäjä pääsee suoraan haluamansa kategorian kohdalle.



Kuvio 8. Kategoriapainike vie käyttäjän valitseman kategorian luo.



Kuvio 9. Navigaatiolla on sama toiminnallisuus kuin kategoriapainikkeilla.

Projektitiimille ehdotetaan elementtikirjastoon lisättävien elementtien tuotantotavaksi atomista suunnittelua ja 8-point gridin käyttöä. Näiden kahden suunnittelutavan käyttäminen lisää elementtien yhtenäisyyttä ja vähentää tulevaisuudessa verkkosivuteemojen kokoamiseen vaadittavaa työtä. Yrityksen tuotantotiimi päättää kuitenkin kuinka tiukat määritteet elementtien tuottamiseen luodaan ja mitä suunnittelujärjestelmiä otetaan käyttöön.

Elementtikirjastosta löytyvät elementit eivät ole ensimmäisessä kehitysversiossa pelkästään atomitason elementtejä, vaan molekyylejä ja organismeja. Nämä elementit esitetään ilman värejä tai fonttityylejä, jolloin suunnittelija voi keksittyä niiden toiminnallisuuden ja ominaisuuksien arviointiin visuaalisuuden sijasta.

Ensimmäisessä kehitysversiossa elementit näytetään visuaalisesti pelkistettyinä etusivulla omien kategorioidensa alle luokiteltuina. Elementeissä näytetään kansion nimi, josta ne löytyvät sekä mahdollinen esimerkkisisältö (kuvio 10). Elementtien hallittavaksi tehtyjä ominaisuuksia voidaan tarkastella yrityksen oman järjestelmän kautta, jolloin niitä ei tarvitse esittää erikseen elementtien käyttöliittymissä.



Kuvio 10. Jokaisessa elementissä näytetään nimi ja esimerkkisisältö.

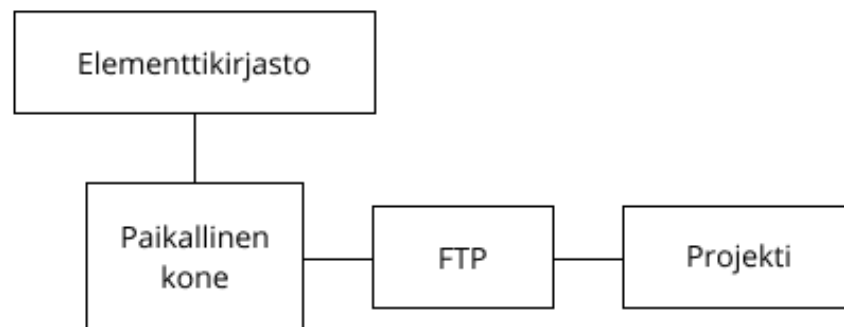
Ensimmäinen kehitysversio on jo toimiva elementtikirjasto. Seuraavissa kehitysversioissa elementtikirjaston toiminnallisuutta automatisoidaan ja kehitetään eteenpäin.

6.5 Elementtikirjaston kehitysehdotukset

Kehitysehdotukset ovat ehdotuksia siitä, mitä elementtikirjaston tulevat kehitysversiot voisivat olla ja miten konseptin ominaisuuksia voidaan kehittää eteenpäin. Ne ovat tarkoitettu viemään keskustelua eteenpäin ja antamaan yritykselle valmiita näkökulmia ja lähtökohtia tekniikan toteutukselle.

Elementtikirjaston tavoitteena on olla helppokäyttöinen ja tehokas työkalu suunnittelijoille ja toteuttajille. Tavoitteen saavuttamiseksi elementtien lataamisen ja

lisäämisen tulisi olla nopeaa ja yksinkertaista. Elementtejä on voitava ladata ja lisätä suoraan sivuston päältä ilman erillisiä FTP-ohjelmia (kuvio 11). Käyttöliittymän tulisi olla mahdollisimman yksinkertainen.



Kuvio 11. Elementtejä lisätään ja ladataan suoraan käyttöliittymän päältä.

Etusivulta käyttäjä näkee elementtikirjaston navigaation sekä hakualueen. Hakualue sisältää ennakoivan sanahaun, kategoriapainikkeet sekä elementtien lajitteluvaihtoehdot. Elementtikirjasto listaa käyttäjän määrittelemien hakutermien mukaiset elementit etusivulle hakualueen alapuolelle. Näitä elementtejä voidaan järjestää lisäyspäivämäärän, latauskertojen tai käyttäjien antamien arvosteluiden mukaan.

Elementtien määrän kasvaessa hakumääritteitä kehitetään eteenpäin. Kategorioiden ja kategoriapainikkeiden lisäksi elementtejä luokitellaan niiden toiminnallisuuden mukaan. Hakuosiosta voidaan hakusanojen lisäksi määrittellä hakuehdoja, joissa rajataan minkälaisia elementtejä näytetään. Hakuehdoissa voidaan esimerkiksi näyttää vain ne elementit, joihin voi lisätä kuvia tai joissa on vain

tekstiä. Sanallisen haun, toiminnallisuuteen keskittyvien määritteiden ja elementtien listausvaihtoehtojen yhdisteleminen varmistaa sen, että suunnittelija löytää elementtikirjastosta tarvitsemansa riippumatta elementtien suuresta määrästä.

Navigaatiossa sijaitsevista kategorialinkeistä päästään elementtikirjaston sisällä sijaitseville erillisille sivuille, jotka kokoavat valitun kategorian kaikki elementit yhteen. Tämä mahdollistaa elementtikirjaston monipuolisen käytön. Siinä missä etusivun hakualue tarjoaa nopean tavan etsiä ja ladata elementtejä, kategoriasivut ovat tarkoitettu syvempään suunnitteluun ja elementtien vertailuun keskenään. Useita eri kategoriasivuja voi olla auki selaimen välilehdillä samaan aikaan. Myös kategoriasivuilla on mahdollisuus lajitella elementtejä eri määritteiden mukaisesti.

Jokaisella yksittäisellä elementillä on oma uniikki sivunsa elementtikirjastossa, jossa elementti on koko selaimen levyinen. Tällä sivulla elementin responsiivisuutta voidaan testata pienentämällä ja suurentamalla selainikkunaa. Elementtikirjaston käyttäjät voivat antaa elementeille arvosteluita perustuen niiden käytettävyyteen ja responsiivisuuteen. Näin elementtejä voidaan järjestää kategorioiden sisällä myös arvosteluiden mukaisesti. Tämä helpottaa elementtikirjaston käyttöä ja elementtien päivittämistä, kun niiden määrä alkaa kasvaamaan.

Elementtikirjaston tavoitteena on kehittyä nopeasti ja tehokkaasti toimivaksi kokonaisuudeksi. Sen tarkoitus on palvella tuotantotiimiä. Kehitysversioita muokataan tuotantotiimin antaman palautteen mukaisesti ja usein suurimmat kehitystarpeet löytyvät vasta silloin, kun prototyyppiä päästään käyttämään projekteissa. Myös erilaiset kategoriat määrittyvät sen mukaan minkälaisia elementtejä projektitiimi sinne lataa.

6.6 Elementtikirjaston prototyyppi

Elementtikirjaston prototyyppi on kehittyvä prototyyppi. Konseptin määritteet ja ominaisuudet tulevat kehittymään ja mukautumaan työskentelyprosessien mukaan eikä elementtikirjasto tule ikinä olemaan valmis tuote. Kehittyvää prototyyppiä käytetään ohjelmistojen kehityksessä (Butler ym. 2010, 194), joten se sopii tähän tarkoitukseen hyvin.

Ensimmäinen kehitysversio sisältää elementtikirjaston etusivun responsiivisen käyttöliittymän, johon on aseteltu esimerkkinä valmiita elementtejä. Prototyypissä ei ole toiminnallisuutta, kuten haku-, lataus- tai lisäysmahdollisuuksia. Käyttöliittymä toteutetaan atomisen suunnittelutavan sekä 8-point gridin mukaisesti.

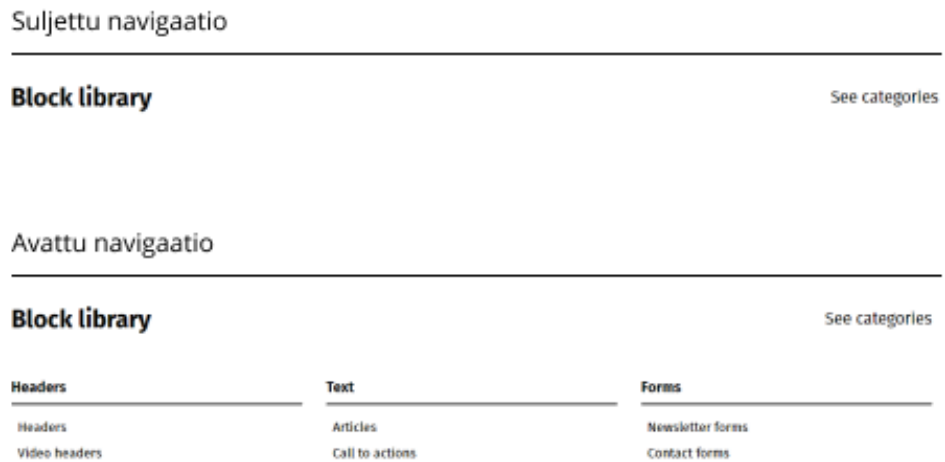
6.7 Prototyypin toteutus

Prototyypin rakentaminen aloitettiin luonnostelemalla ulkoasua tärkeimpien konseptissa määriteltyjen ominaisuuksien perusteella. Navigaation ja hakuosion on oltava luonnollisia osia elementtikirjastoa ja niiden on annettava elementeille tilaa. Etusivulla on kuitenkin tarjottava tietoa siitä, mikä elementtikirjasto on. Päädyin rakentamaan navigaation ja hakuosion etusivun yläosaan, jolloin ne erottuvat ladattavista elementeistä. Hakuosiossa on myös visuaalisuutta, joka auttaa erottamaan elementtejä ja hakuosiota toisistaan.

Seuraavassa vaiheessa prototyypin rakentaminen siirtyi selaimen. Kaikki elementtikirjaston prototyypissä käytetyt käyttöliittymän elementit ovat toteutettu käyttämällä atomista suunnittelua ja 8-point gridiä. Ne ovat myös ladattavissa elementtikirjaston elementtien hakemistosta ja niitä voi halutessaan käyttää projekteissa. Näin elementtikirjasto käyttää samoja CSS- ja HTML-tiedostoja kuin sieltä ladattavat elementit. Elementtikirjaston pohjalla toimii Bootstrap, jonka avulla kirjastosta saadaan nopeasti ja helposti responsiivinen.

Rakensin responsiivisen navigaation, joka sisältää elementtikirjaston logon sekä kategorialinkit. Kategorialinkit ovat asetettu avattavan ja suljettavan navigaation sisälle, jolloin niiden kasvava määrä ei häiritse käyttökokemusta (kuva 1). Elementtikirjaston englanninkieliseksi nimeksi määriteltiin 'Block library', joka tulee

yrittäjien tuotantotiimin sisäisestä puhekielestä. Suomeksi elementtikirjaston kutsumanimi on blokkikirjasto.



Kuva 1. Suljettu ja avattu navigaatio esimerkkisisällöillä.

Hakuosio on ainoa visuaalinen elementti sivustolla kategorioiden käytettyjen värien lisäksi. Sen taustakuvana toimii aiheeseen sopiva kuvituskuva. Tekstiosio tarjoaa informaatiota elementtikirjaston käyttötarkoituksesta (kuva 2).



Kuva 2. Elementtikirjaston hakuosio.

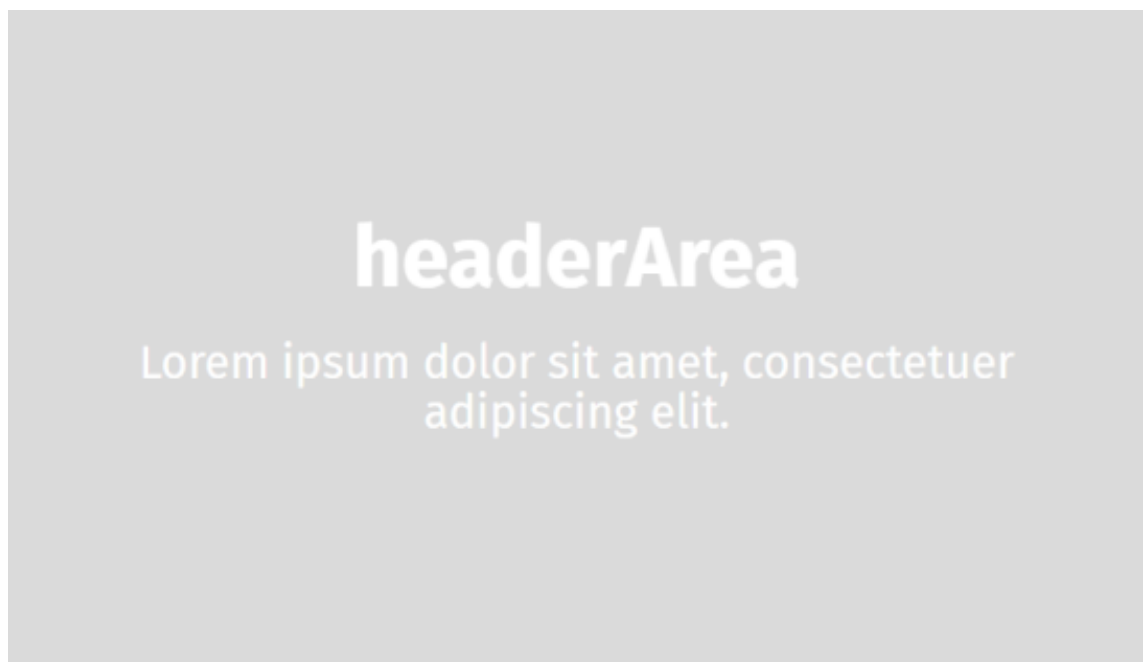
Navigaation ja hakuosion jälkeen alkaa kategorioiden sekä elementtien listaus. Jaoin elementit kategorioihin käyttämällä ensimmäisen otsikkotason otsikkoa sekä paksua linjaa, joka erottaa elementit toisistaan selvästi. Käyttämällä korosteväriä varmistin, etteivät kategorioiden otsikot sekoitu elementteihin (kuva 3).

Headers

Kuva 3. Elementit erotellaan toisistaan kategoriolla.

Elementtikirjaston elementit sijoitetaan näiden kategorioiden alle. Niihin ei lisätä visuaalisia elementtejä, kuten kuvia tai ikoneita. Isojen elementtien korkeutta rajoitetaan, jotta useampia elementtejä mahtuu sivulle samanaikaisesti. Näiden elementtien korkeudeksi määritellään 320 pikseliä. Taustakuvana käytetään taustaista harmaata taustaa, joka asetetaan hallintaan käyttövalmiiksi (kuva 4).

Elementit ovat koko selaimen levyisiä työpöytäleveyteen asti. Työpöytäleveydestä eteenpäin ne ovat keskitettyjä ja niiden leveyttä on rajoitettu. Tämä lisää elementtikirjaston käytettävyyttä ja auttaa suunnittelijoita hahmottamaan paremmin sivustolla esitetyt elementit. Ratkaisu tehtiin pitäen mielessä tulevat kehitysversiot, joissa jokaisella elementillä on oma uniikki sivustonsa. Näillä uniikkeilla sivuilla elementit täyttävät koko selaimen leveyden. Etusivulla ei siis ole erityistä tarvetta näyttää kokoleveitä elementtejä ja voidaan keskittyä selkeämpään käyttöliittymään.



Kuva 4. Kuvallinen elementti harmaalla taustalla ja rajoitetulla korkeudella.

7 Tulokset ja pohdinta

7.1 Toiminnallisen opinnäytetyön tulokset

Opinnäytteen tuloksena syntyivät elementtikirjaston konsepti ja prototyyppi. Ne ovat valmiita esitettäväksi ja kehitettäväksi. Toteutetun konseptin ja prototyypin pohjalta verkkosivujen tuotantoprosessi on mahdollista muuttaa modulaariseksi. Konseptin ensimmäinen kehitysversio antaa määritteet elementtikirjastolle ja

mahdollistaa niiden kriittisen arvioinnin. Yritys voi kehittää elementtikirjaston konseptia eteenpäin käyttäen hyväksi opinnäytteessä esitettyjä kehitysideoita. Opinnäytetyön toiminnallisena osuutena toteutettu prototyyppi on toimiva työkalu. Elementtikirjastoon voi manuaalisesti lisätä elementtejä kategorioiden alle sekä ladata elementtejä kopioimalla ne suoraan elementtikirjaston hakemistosta. Elementtien rakentamiseen ehdotetaan atomista suunnittelujärjestelmää ja 8-point gridin käyttöä, jolloin elementit ovat modulaarisempia.

Elementtikirjaston käyttöliittymän päälle on yksinkertaista rakentaa seuraavien kehitysversioiden toiminnallisuuden vaatimaa tekniikkaa. Käyttöliittymän visuaalisuutta on myös helppo kehittää eteenpäin tuotantotiimiltä saadun palautteen perusteella. Elementtikirjaston responsiivisen käyttöliittymän työpöytäversio löytyy liitteestä 1, tablettiversio liitteestä 2 ja mobiiliversio liitteestä 3.

7.2 Opinnäytteen tulosten tarkastelu

Opinnäytteen tuloksena syntynyt elementtikirjaston konsepti ja prototyyppi onnistuivat ratkaisemaan, miten verkkosivujen tuotantoprosessi saadaan muutettua modulaariseksi. Uuden tuotantoprosessin ja elementtikirjaston käyttöönotto vaatii aikaa, mutta parhaimmillaan ne mahdollistavat rakennettujen elementtien uusiokäytön. Tämä on yritykselle erittäin arvokasta. Kustannustehokkaiden verkkosivustojen luominen mahdollistaa resurssien siirtämisen muihin huomiota vaativiin projekteihin.

Elementtien kasvavan määrän hallitseminen ja luokittelu on suurin haaste elementtikirjaston käytössä. On tärkeää kehittää toimiva järjestelmä tietomäärän hallitsemiseen ja määriteltävä kuka yrityksen sisältä on päävastuussa elementtikirjaston päivittämisestä ja jatkokehityksestä.

Atomisen suunnittelu toimii hyvänä pohjana modulaarisuuden ajatuksen esittämisessä eteenpäin yritykselle ja tuotantotiimin jäsenille. Sen käyttöönotto elementtien rakentamiseen käytettävänä suunnittelujärjestelmänä riippuu kuitenkin

tuotantotiimin työtavoista ja halukkuudesta omaksua isoja uusia kokonaisuuksia ja toimintamalleja.

Elementtikirjaston käyttöönotto tulisi toimia porrastetusti, jolloin sinne voidaan ladata jo valmiiksi rakennettuja elementtejä, jotka eivät välttämättä ole rakennettu ottaen huomioon atomista suunnittelua tai 8-point gridiä. Tällä tavoin nähdään, onko tarve suunnittelujärjestelmän käyttöönotolle aiheellinen ja miten elementtikirjasto integroituu prosesseihin. On parempi, että tarve atomisen suunnittelutavan kaltaisille suunnittelujärjestelmille tulee tuotantotiimin sisältä. Näin niiden käyttöönotto on kivuttomampaa.

Modulaarinen verkkosuunnittelu on mahdollista ja hyvä ratkaisu tuotantoprosessien optimoimiseen. Elementtikirjaston avulla voidaan tuottaa verkkosivuja nopeammin ja tehokkaammin. Verkkosivujen laatu riippuu täysin elementtikirjastoon tuotettujen elementtien laadusta. Koska Marketplacesta ostettavat verkkosivuteemat vaativat enemmän toiminnallisia ominaisuuksia kuin taiteellisia, on elementtien uudelleenkäytettävyys tärkeää ja hyödyllistä. Verkkosuunnittelun muuttuminen modulaariseksi on looginen seuraava askel alan kehittyessä ja kasvaessa.

7.3 Oppimisprosessi ja ammatillinen kasvu

Opinnäytetyöprosessin aikana sain tutkimastani lähdeaineistosta laajan kuvan verkkosuunnittelun historiasta ja sen nykytilasta. Opin ajattelemaan verkkosivuja modulaarisina kokonaisuuksina ja muokkaamaan omaa suunnittelutapaani sen mukaisesti. Pääsin tutustumaan alan ammattilaisiin ja kasvattamaan sanastoani. Elementtikirjaston rakentaminen ja suunnittelu auttoi minua ymmärtämään prototyypittämisen tärkeyden. Prototyypittämällä voidaan nopeasti tuottaa interaktiivisia luonnoksia, jotka ovat paljon konkreettisempia kuin esimerkiksi rautalankamallit ja staattiset sommitelmat.

On tärkeää suunnitella oma työnsä ja miettiä tarkasti, miten kirjoitetusta koodista saa mahdollisimman luettavaa ja uudelleenkäytettävää. Tekemällä tulevai-

suuden kannalta kestäviä ratkaisuja varmistetaan, että mahdolliset muutokset voidaan toteuttaa hyvin pienillä korjauksilla. Yhteisen kielen löytäminen yrityksen sisällä on myös tärkeää. Jos jokainen työntekijä osaa lukea toisen tuottamaa koodia, ollaan hyvässä tilanteessa. Suunnittelujärjestelmät ovat mielestäni yksi hyvä keino tämän yhteisen kielen luomiseen.

Tapani rakentaa verkkosivuja muuttui opinnäytteen myötä. Vesiputousmallin korvaaminen modulaarisella ajattelulla ja tuotantoprosesseilla on kasvattanut minua suunnittelijana. Pystyn toteuttamaan verkkosivuja nopeammin ja osaan keskittyä niiden tarkoitukseen ja syvimpään toiminnallisuuteen ennen visuaalisuutta. Aion soveltaa modulaarista suunnittelutapaa myös muuhun suunnittelu- ja kehitystyöhön.

Lähteet

- Berners-Lee, T. 2012. Modularity.
<https://www.w3.org/DesignIssues/Modularity.html>. 09.04.2017.
- Berners-Lee, T. 2013. Principles of Design.
<https://www.w3.org/DesignIssues/Principles.html#Modular>.
09.04.2017.
- Butler, J., Holden, K. & Lidwell, W. 2010. Universal Principles of Design. Beverly: Rockport Publishers, Inc.
- Carver, M. 2015. The Responsive Web. New York: Manning Publications Co.
- Dahl, E. 2016. Intro to the 8-Point Grid System. <https://builttoadapt.io/intro-to-the-8-point-grid-system-d2573cde8632>. 12.04.2017.
- Duckett, J. 2014. HTML & CSS Design and Build Websites. Indianapolis: John Wiley & Sons, Inc.
- Friedman, V. 2015. Responsive Design Patterns and Components. Smashing Magazine. (toim.) Smashing Book 5 – Real-Life Responsive Web Design. Saksä: Smashing Magazine, 49-97.
- Frost, B. 2016. Atomic Design. Pittsburgh: Brad Frost.
- Macdonald, N. 2003. What is Web Design? Mies: RotoVision.
- Mall, D. 2015. The Modern Responsive Designer's Workflow. Smashing Magazine. (toim.) Smashing Book 5 – Real-Life Responsive Web Design. Saksä: Smashing Magazine, 13-46.
- Marcotte, E. 2010. Responsive Web Design. A List Apart.
<https://alistapart.com/article/responsive-web-design>. 25.03.2017.
- O'Toole, J. 2014. Mobile apps overtake PC Internet usage in the U.S.
<http://money.cnn.com/2014/02/28/technology/mobile/mobile-apps-internet/>. 26.03.2017.
- Park, T. 2013. How To Customize Twitter's Bootstrap.
<https://www.smashingmagazine.com/2013/03/customizing-bootstrap/>. 10.04.2017.
- Snook, J. 2017. Scalable and Modular Architecture for CSS.
<https://smacss.com/book/categorizing>. 09.04.2017.
- Starkov, V. & Stepanova V. 2017. Introduction to BEM.
<http://getbem.com/introduction/>. 09.04.2017.
- Sullivan, N. 2013. Object Oriented CSS.
<https://github.com/stubbornella/oocss/wiki>. 09.04.2017.

W3C. 2012. Media Queries. <https://www.w3.org/TR/css3-mediaqueries/>.
25.03.2017.

Warren, S. 2017. Style Tiles. <https://styletil.es>. 12.04.2017.

Wroblewski, L. 2009. Mobile First. <http://www.lukew.com/ff/entry.asp?933>.
25.03.2017.

Elementtikirjaston tablettiversio



Headers

headerArea
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.

headerAreaFlow
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.

featureLeft

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

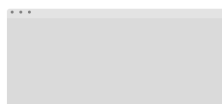
Sample button

Video headers

headerAreaBoxed

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Sample button



Articles



article

subtext

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Sample button

Call to actions

callToActionStrip Sample button

Back to top

Elementtikirjaston mobiiliversio



Headers

headerArea

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

headerAreaFlow

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

featureLeft

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisi ut aliquip ex ea commodo consequat.

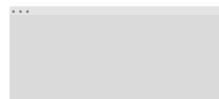
Sample button

Video headers

headerAreaBoxed

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Sample button



Articles



article

subtext

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Sample button

Call to actions

callToActionStrip

Sample button

[Back to top](#)