

Ville-Valtteri Väyrynen ja Jeremias Vihanta

HTML-sivuston toteutus media-alan yritykselle

HTML-sivuston toteutus media-alan yritykselle

Ville-Valtteri Väyrynen ja
Jeremias Vihanta
HTML-sivuston toteutus media-alan
yritykselle
Kevät 2015
Tietojenkäsittely
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu Oy
Tietojenkäsittelyn koulutusohjelma,
Internetpalvelut ja digitaalinen media

Tekijät: Ville-Valtteri Väyrynen ja Jeremias Vihanta
Opinnäytetyön nimi: HTML-sivuston toteutus media-alan yritykselle
Työn ohjaaja: Teppo Räisänen
Työn valmistumislukukausi- ja vuosi: Kevät 2015 Sivumäärä: 41

Rapid River on Kuusamosta lähtöisin oleva, nykyään oululainen mediatoimisto ja tuotantoyhtiö. Saimme opinnäytetyön aiheen harjoittelun aikana, sillä yrityksen toimitusjohtajalla Tauno Kohosella oli tarve Siperia-projektin esittelysivustolle. Siperia-projektia varten on olemassa laaja kuva- ja videomateriaali, jota toimeksiantaja Tauno Kohonen halusi hyödyntää.

Opinnäytetyö jakautuu kahteen osioon, teoriaosuuteen ja web-sivuston toteutukseen. Opinnäytetyön web-sivuston toteutusvaiheen tavoitteena oli luoda visuaalisesti näyttävät esittelysivut Siperia-projektin materiaalia varten. Sivusto toteutettiin HTML5-tekniikkaa käyttäen. Esittelysivut toimivat introna varsinaiselle projektille ja sen jo olemassa olevalle web-sivustolle. Esittelysivuston avulla voidaan esitellä Siperia-projektia tehokkaasti sekä nopeasti uusille yhteistyökumppaneille.

Opinnäytetyön tietoperusta eli teoriaosuus on jaettu kahteen eri osaan, teknisiin ratkaisuihin ja ulkoasun suunnitteluun. Teknisissä ratkaisuissa käydään läpi esittelysivuston luomista varten käytetyt tekniikat, kuten HTML, CSS, JavaScript, jQuery sekä Bootstrap. Opinnäytetyössä kerrotaan myös yleistä tietoa HTML-editoreista ja responsiivisuudesta. Tietoperustan toisessa osuudessa, ulkoasun suunnittelussa kerrotaan yleisesti väreistä, typografiasta ja kuvien käytöstä. Esittelysivusto luotiin tämän tietoperustan pohjalta.

Toimeksiantajaa on ohjeistettu web-sivuston ylläpitämisestä, joten esittelysivustoa voidaan muokata jatkossa lisäämällä uutta materiaalia. Työn tuloksena saatiin toimeksiantajan kriteerit täyttävä web-sivusto, johon myös työn tekijät ovat tyytyväisiä.

Asiasanat: HTML, CSS, Bootstrap, jQuery

ABSTRACT

Oulu University of Applied Sciences
Degree Programme of Business Information Systems,
Option of Internet Services and Digital Media

Authors: Ville-Valtteri Väyrynen and Jeremias Vihanta
Title of thesis: Implementing a HTML Website for a Media Company
Supervisor: Teppo Räisänen
Term and year when the thesis was submitted: Spring 2015 Number of pages: 41

Rapid River is a media company which started out from Kuusamo but is now based in Oulu. The subject for the thesis emerged during the authors' work placement period as the company owner Tauno Kohonen had a need for a website showcasing his Siberia project. There is a vast archive of photo and video material which Tauno Kohonen wanted to make use of.

This thesis is divided into two different segments: theoretical framework and the implementation of the website. Authors' goal during the implementation of the website was to create a graphically beautiful website showcasing the material of the Siberia project. The website was made using HTML5-coding. The purpose of the website is to function as an introduction for the actual project and for the existing Siberia project website. The website can be used as a quick and efficient way to introduce the idea for new associates.

The theoretical framework has been divided into two different parts which are technical and graphic design. The theoretical framework introduces the different technologies used during the implementation of the website such as HTML, CSS, JavaScript, jQuery and Bootstrap. The authors also analyzed the basics of HTML-editors and responsive design. The second part of the theory segment which focuses on graphical design includes the basics of using colors, typography and pictures when creating a website. The website was made based on the knowledge and information the authors gathered during the thesis process.

Tauno Kohonen has been informed on how to maintain the website and how to update new material for the website. The resulting website fulfilled the criteria given by Tauno Kohonen and also the criteria of the authors before creating the website.

Keywords: HTML, CSS, Bootstrap, jQuery

SISÄLLYS

1	JOHDANTO.....	6
2	TEKNISET RATKAISUT	7
2.1	HTML	7
2.1.1	HTML-versiot.....	7
2.1.2	HTML5	8
2.1.3	Elementit	9
2.1.4	Yleiset määritteet.....	10
2.1.5	Attribuutit.....	11
2.2	CSS.....	12
2.2.1	CSS-versiot.....	13
2.2.2	CSS:n perusteet.....	13
2.3	HTML-editorit.....	15
2.4	Web-selaimet	16
2.5	JavaScript/jQuery	16
2.6	Bootstrap.....	18
2.7	Responsiivisuus.....	19
3	ULKOASUN SUUNNITTELU	21
3.1	Värit.....	21
3.2	Typografia	24
3.3	Kuvat.....	25
4	SIVUSTON TOTEUTUS	27
4.1	Toimeksiantaja	27
4.2	Sivuston esittely.....	28
4.2.1	Opinnäytetyön aihe	28
4.2.2	Sivuston rakenne.....	28
4.2.3	Sivuston graafinen ilme	34
4.3	Käyttöönotto	35
4.4	Haasteet.....	36
5	POHDINTA.....	37
	LÄHTEET	39

1 JOHDANTO

Lähes jokaisella yrityksellä on nykyisin omat web-sivut. Media-alan yrityksillä on yleensä käynnissä useita eri projekteja samanaikaisesti, ja projektit tarvitsevat usein oman web-sivuston. Web-sivut lisäävät projektin tunnettavuutta ja tekevät siitä helpommin lähestyttävän. Web-sivujen kautta on helppoa ottaa yhteyttä yritykseen. Opinnäytetyössä käydään läpi web-sivuston luomiseen liittyviä teknisiä ja graafisia perusteita.

Opinnäytetyönä suunnittelemme ja toteutamme esittelysivuston Rapid Riverin Siperia-projektille. Rapid Riverin omistaja Tauno Kohonen on kuvannut laajan kuva- ja videomateriaalin Venäjän Siperialta, ja Siperian materiaali halutaan esitellä web-sivustolla mahdollisille uusille yhteistyökumppaneille. Sivustosta pyritään tekemään visuaalisesti näyttävät ja materiaali pyritään esittämään mahdollisimman tehokkaasti, koska web-sivusto toimii pelkästään esittelynä laajemmalle Siperia-kuvakirjastolle.

Rapid River on oululaisen Tauno Kohosen omistama mediatoimisto, joka on lähtöisin Kuusamosta. Tauno Kohonen toimii myös Rapid Riverin toimitusjohtajana sekä kuvaajana. Rapid Riverin tärkeimpiä tuotteita ovat useat eri kuvagalleriat netissä, joita on tehty useille eri projekteille. Rapid River suunnittelee myös asiakkaille kuvagallerioita, nettisivuja ja videotuotantoja.

2 TEKNISET RATKAISUT

2.1 HTML

HTML on tekstiä ja rakennetta. Tässä yhteydessä sana rakenteinen tarkoittaa, että HTML-dokumentin tekstisisällön lisäksi se sisältää loogisen rakenteen osoittavaa merkkausta. Tätä kutsutaan englanniksi nimellä ”markup”. HTML-dokumentti sisältää pääsääntöisesti vain tekstiä. Se voi kuitenkin sisältää viittauksia muihin dokumentteihin. Näiden viittauksien avulla HTML-dokumentin esitysmuotoon saadaan mukaan esimerkiksi kuvia. (Korpela & Linjama 2005, 70.)

Yleisesti ottaen HTML:ää kutsutaan kieleksi, tarkemmin sanottuna merkkaukieleksi. HTML-lyhenne tulee englannin kielen sanoista ”hypertext markup language”, joka tarkoittaa ”hypertekstin merkkaukieltä”. On kuitenkin tärkeä ymmärtää, että se ei ole samanlaista kieltä kuin esimerkiksi suomi ja ruotsi, vaan tietotekniikan merkkaukieli. Nämä tietotekniikan ”kielet” ovat erittäin täsmällisesti määriteltyjä ja hyvin tarkkoja muodoltaan, koska niillä on oma erityistarkoituksensa. (Korpela & Linjama 2005, 70.)

2.1.1 HTML-versiot

HTML 1.0 oli ensimmäinen julkaistu versio HTML:stä. Tähän aikaan web-sivujen toteutus- ja merkkaukielet olivat hyvin rajoitettuja. HTML 1:llä sai aikaan vain yksinkertaista tekstiä web-sivuille. HTML 2.0 sisälsi kaiken HTML 1.0:stä muutamien lisäominaisuuksien kera. HTML 2.0 oli web-suunnittelun standardi vuoteen 1997 asti ja määritteli HTML:n ydinominaisuuksia ensimmäistä kertaa. (The History of HTML, Viitattu 14.11.2014.)

HTML 3.0:n version julkaisun jälkeen yhä suurempi osa web-sivujen tekijöistä siirtyi sen käyttämiseen, koska edellinen versio HTML:stä tarjosi ominaisuuksia vain tyydyttävästi. HTML 3.0 tarjosi uusia ja paranneltuja ominaisuuksia. HTML 3.0:n käyttöön sisältyi kuitenkin useita ongelmia. Esimerkiksi selaimet toimivat erittäin hitaasti uusien ominaisuuksien kanssa. Tämän takia HTML 3.0:n käyttö lopetettiin. Jatkossa päivitykset julkaistiin osissa, joka helpotti selainkehittäjien työtä. Vuonna 1994 perustettu W3C loi HTML 3.2:n ensimmäisenä projektinaan ja kehittää vieläkin näitä standardeja oikeaan suuntaan. HTML 3.2:n myötä selainkohtaiset tagit yleistyivät ja myös standardien

tarve kasvoi. HTML 3.2:n muutokset olivat kuitenkin hillittyjä ja isoimmat muutokset tulivat myöhemmissä versioissa. (The History of HTML, Viitattu 14.11.2014.)

HTML 4.0 oli suuri kehitysaskel HTML:n standardien kannalta. Monet sen toiminnollisuuksista tuotiin HTML 3:sta. Monet vanhoista tageista poistettiin käytöstä, ja keskityttiin kansainvälistämiseen ja uusien tyyli-tiedostojen tukemiseen. Vuonna 1997 W3C suositteli HTML 4.0:aa ja siitä tuli virallinen standardi huhtikuussa 1998. Myöhemmin korjattiin muutamia virheitä HTML 4.0:sta ja se sai loppulisen muotonsa HTML 4.01:n. XHTML 1.0 on uusi haarautuma HTML-kielissä. Vuodesta 2000 asti XHTML 1.0 on ollut yhteisstandardi HTML 4.01:n kanssa. XHTML 1.0 ei kuitenkaan lisännyt juurikaan uusia tageja tai attribuutteja. XHTML 1.0 on pääasiassa vain uusia koodaussääntöjä. (The History of HTML, Viitattu 14.11.2014.)

HTML 4.01:n ja XHTML 1.0:n tekijät yrittivät kehittää XHTML 2:ta. Sen kehittäminen alkoi kuitenkin vaikuttamaan epärealistiselta ja tarvittiin uusi lähestymistapa. WHATWG (Web Hypertext Application Technology Group) alkoi kehittämään projektia joka tunnetaan nykyisin HTML5:nä. HTML5 on suunniteltu käytettäväksi nyt ja tulevaisuudessa. Sen standardien mukaan tullaan työskentelemään useita vuosia, joten sen kehitysprosessi on suhteellisen hidasta ja harkittua. Monet sen ominaisuuksista ovat jo ennestään tuttuja, mutta se sisältää myös paljon uusia elementtejä, attribuutteja ja toiminnallisuuksia. (The History of HTML, Viitattu 14.11.2014.)

2.1.2 HTML5

HTML5:n avulla on mahdollista toteuttaa web-sivustolla animaatioita, musiikkia ja videoita ilman erillisiä sovelluksia ja sitä voidaan käyttää rakentamaan monimutkaisia sovelluksia, jotka toimivat web-selaimessa. HTML5:n käyttö ei ole patentoitu, joten sen käytöstä ei tarvitse maksaa rojalteja. HTML5:tä voidaan käyttää kaikilla eri alustoilla kuten tableteilla, älypuhelimilla ja smart-televisioilla, kunhan web-selain tukee HTML5:tä. HTML5:tä voidaan käyttää web-aplikaatioiden kirjoittamiseen, jotka toimivat myös ilman internet-yhteyttä.

Kaikki tunnetuimmat web-selaimet, kuten Internet Explorer, Firefox, Chrome, Safari, Opera ja älypuhelimien Safari- ja Android-selaimet tukevat HTML5:n käyttöä. Ne eivät kuitenkaan kaikki tue

samoja ominaisuuksia. Mozilla Firefox tukee suurinta määrää HTML5:n ominaisuuksia, ja heti sen perässä tulevat Chrome ja Safari. (HTML5: what is it? Viitattu. 13.12.2014.)

HTML5:n uusista ominaisuuksista tärkeimpiä ovat uudet elementit <header>, <footer>, <article> ja <section>. Tärkeitä ovat myös uudet graafiset elementit <svg> ja <canvas> sekä uudet multimedia elementit <audio> ja <video>. HTML5-dokumentin luonti on yksinkertaista, koska sen DOCTYPE ja merkistön koodaus ovat erittäin selkeitä. Katso kuvio 1. (HTML5 Introduction, Viitattu 14.12.2014)

```
<!-- Esimerkki HTML 5 dokumenttipohja -->
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Dokumentin otsikko</title>
</head>

<body>
Dokumentin sisältö.
</body>

</html>
```

KUVIO 1. (HTML5 Introduction, Viitattu 14.12.2014)

2.1.3 Elementit

HTML-dokumentit muodostuvat elementeistä. Harvoja poikkeuksia lukuun ottamatta elementit muodostuvat aina alkutagista, sisältöosasta ja lopputagista. Elementillä tarkoitetaan tätä kokonaisuutta alkutagista aina lopputagiin asti. Katso kuvio 2. (HTML Elements, Viitattu 20.11.2014.)

Alkutagi	Elementit sisältö	Lopputagi
<p>	Ensimmäinen html kappaleeni	</p>

KUVIO 2. (HTML Elements, Viitattu 20.11.2014)

Ne elementit joita käytetään HTML-dokumentin sisällössä, voidaan jakaa kahteen pääryhmään. Lohkoelementit ovat suuria rakenneosia, esimerkiksi tekstikappaleita ja taulukoita. Lohkoelementtien esitysmuoto on tavallisesti suorakulmainen. Lohkoelementtejä ovat muun muassa div, h1 – h6,

p, table ja ul. Sisäelementit ovat tekstitason elementtejä, jotka sisältävät yleensä lyhyitä tekstinpätkiä, esimerkiksi korostettuja sanoja kappaleen sisällä. Sisäelementtejä ovat muun muassa lomakkeiden kentät ja kuvat. (Korpela & Linjama 2005, 73 – 74.)

Elementit voivat myös olla sisäkkäisiä. Tämä tarkoittaa sitä, että elementtien sisällä voi olla vielä toisia elementtejä. Esimerkiksi HTML-elementti määrittää koko HTML-dokumentin, ja sen sisään kuuluu body-elementti ja monia muitakin elementtejä. Katso kuvio 3. (HTML Elements, Viitattu 20.11.2014.)

```
<!-- Esimerkki elementeistä -->
<!DOCTYPE html>
<html>

<body>
  <h1>Ensimmäinen otsikkoni</h1>
  <p>Ensimmäinen kappaleeni</p>
</body>

</html>
```

KUVIO 3. (HTML Elements, Viitattu 20.11.2014)

Kaikilla Elementeillä on oma tarkoituksensa. HTML-elementti määrittää koko dokumentin ja body-elementti määrittää dokumentin rakenteen. H1-elementti määrittää otsikon ja p-elementti määrittää kappaleen. Osa elementeistä toimii ilman lopputagia, mutta on aina suositeltavaa käyttää sitä ongelmien välttämiseksi. On myös suositeltavaa käyttää pieniä kirjaimia tageissa. (HTML Elements, Viitattu 20.11.2014.)

2.1.4 Yleiset määritteet

Joitain määritteitä voidaan liittää moniin elementteihin. Näitä määritteitä ovat class, id, lang ja title. Nämä määritteet ovat merkitykseltään yleisluonteisia. Ne eivät ole muodollisesti pakollisia, mutta niillä on paljon käyttöä. (Korpela & Linjama 2005, 75.)

Määrite class="nimi" sisällyttää elementtiin tiedon siitä, että elementti sisältyy nimettyyn luokkaan (class). Nämä luokat ovat erittäin hyödyllisiä tyylisäännösten yhteydessä. Voidaan esimerkiksi haluta, että jokin kappale näkyy web-selaimessa punaisen reunuksen sisällä, koska siinä on tärkeä varoitus. Kaikkien p-elementtien ei kuitenkaan haluta näkyvän tällä värillä, joten annetaan halutulle

kappaleelle luokka "varoitus". Tyylisäännöstössä voidaan nyt viitata rakenteella "p.varoitus", joka tarkoittaa niitä p-elementtejä, jotka kuuluvat varoitus-luokkaan. Katso kuvio 4. (Korpela & Linjama 2005, 75.)

```
<!-- Esimerkki varoitus-luokasta HTML-dokumentissa-->
<p class="varoitus">Varoitus: Sivun lopussa paljastetaan elokuvan loppuratkaisu!</p>

/* HTML-dokumenttiin liittyvä tyylisäännöstö*/
p.varoitus { border: solid red 0.1em; padding: 0.3; }
```

KUVIO 4. (Korpela & Linjama 2005, 75)

Määrite id="tunniste" lisää tunniste elementille. Tämän tunniste avulla elementti voidaan yksilöidä dokumentin sisällä. Tunniste saa sisältää ainoastaan englannin kielen kirjaimia. Lisäksi se saa sisältää numeroita, yhdysmerkkejä, alaviivoja, kaksoispisteitä ja pisteitä. Tunniste ei saa kuitenkaan alkaa numerolla. Edellinen esimerkki olisi myös voitu tehdä tunniste avulla. Tällöin HTML-dokumentin class-sana olisi korvattu id-sanalla ja tyylisäännöstössä olisi "p#varoitus". (Korpela & Linjama 2005, 76.)

Määrite lang="kielitunnus" ilmoittaa, että elementin kieli on se, jonka kielitunnus kertoo. Esimerkiksi suomen kielen koodi on "fi" ja englannin "en". Kielimerkkauksen käytöstä on monenlaista hyötyä. Esimerkiksi HTML-muotoa lukevat tekstinkäsittelyohjelmat voivat sen avulla päätellä mitä kieltä tekstin osat ovat ja osaavat sen takia soveltaa kielen mukaisia kielitarkastussääntöjä (Korpela & Linjama 2005, 76.)

2.1.5 Attribuutit

HTML-dokumentin elementeillä voi olla attribuutteja. Attribuutit tarjoavat lisätietoja elementistä. Attribuutit määritellään aina elementin alkutagissa. Esimerkiksi HTML-dokumentin HTML-elementissä voidaan määrittää dokumentin kieli. Tämä on erittäin tärkeää esimerkiksi hakukoneiden takia. Katso kuvio 5. (HTML Attributes, Viitattu 22.11.2014.)

```

<!-- Esimerkki lang-attribuutista -->
<!DOCTYPE html>
<html lang="fi">
<body>

<h1>Ensimmäinen otsikkoni</h1>
<p>Ensimmäinen kappaleeni</p>

</body>
</html>

```

KUVIO 5. (HTML Attributes, Viitattu 22.11.2014)

HTML-dokumentissa voidaan myös antaa esimerkiksi p-elementille title-attribuutti, joka nimeää tämän kappaleen. Kun hiirtä pidetään tämän nimetyn kappaleen päällä, tulee nimi näkyviin. HTML-dokumenteissa linkit määrittää a-elementti, ja sille voidaan antaa href-attribuutti, joka määrittää osoitteen tälle linkille. Myös kuvien leveys ja korkeus voidaan antaa attribuuttien avulla. Alt-attribuutin avulla voidaan määritellä vaihtoehtoinen teksti näytettäväksi, jos kuvaa ei ole saatavilla. Katso kuvio 6. (HTML Attributes, Viitattu 22.11.2014.)

```

<!--Esimerkkejä attribuuteista-->

<a href="http://www.w3schools.com">Tämä on linkki</a>


```

KUVIO 6. (HTML Attributes, Viitattu 22.11.2014)

2.2 CSS

Käytetyin ja lähes ainoa tyylisäännöstö kieli internetissä on CSS (Cascading Style Sheets). Voidaan sanoa, että tyylisäännöstö on dokumentit esitysasua koskeva ehdotus, joka on kirjoitettu sitä varten tehdyllä kielellä. Eri web-selaimet esittävät internetsivustot omalla tavallaan. Tyylitiedostot avulla voidaan vaikuttaa selaimen tavallisesti käyttämiin piirteisiin. (Korpela & Linjama 2005, 300.)

CSS-tiedosto mahdollistaa HTML- ja tyylisuuden erottelun toisistaan. HTML-osuutta käytetään sisällön järjestämiseen, mutta kaikki tyylimäärittelyt saadaan aikaan erillisen tyylisäännöstön (CSS) avulla. CSS:ssä voidaan muun muassa määritellä fontit, värit, taustat, tekstiformaatit ja linkkien efektit. (Chapter 1: Introduction to CSS, Viitattu 22.11.2014.)

Lyhyesti sanottuna CSS:n avulla voidaan vaikuttaa web-sivuston esittämiseen (presentation) ja rakenteeseen (layout). Esittämällä tarkoitetaan esimerkiksi web-sivuston tekstien väriä, fonttien kooka ja muotoa, taustavärejä ja -kuvia. Rakenteella puolestaan tarkoitetaan sitä, missä kohdassa web-sivuston eri elementit näkyvät näytöllä. (Duckett 2011, 6.)

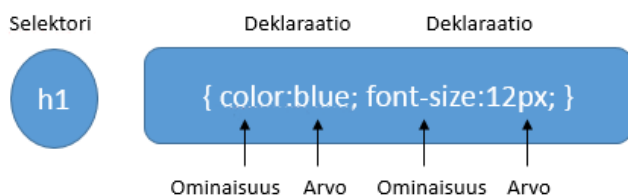
2.2.1 CSS-versiot

Toisin kuin HTML:ssä, jossa versioiden on oltava ennalta määrättyjä selaimelle näkyäkseen oikein, CSS:n syntaksi on aina pysynyt samanlaisena. Syntaksilla tarkoitetaan tapaa, jolla CSS kirjoitetaan. Tämä tarkoittaa sitä, että CSS1-määrittelyt ovat täysin toimivia myös CSS3:ssa. Ainoat todelliset erot CSS-versioiden välillä ovat uudet ominaisuudet ja valitsimet. (HTML & CSS: Versions of HTML and CSS, Viitattu 22.11.2014.)

CSS3:n myötä monet ennen mahdottomat tehtävät ovat nykyään helposti tehtävissä. Osa selaimista ei kuitenkaan vielä tue uusimman CSS3:n kaikkia ominaisuuksia. Nämä ei-tuetut ominaisuudet ovat kuitenkin vain huomaamattomia yksityiskohtia, eikä niiden puuttuminen ole merkittävä asia. (HTML & CSS: Versions of HTML and CSS, Viitattu 22.11.2014.)

2.2.2 CSS:n perusteet

CSS-sääntö koostuu selektorista ja deklaraatiosta. Selektori osoittaa sen elementin HTML-dokumentista, joka halutaan tyylitellä. Deklaraatio koostuu ominaisuuden nimestä, kaksoispisteestä ja ominaisuuden arvosta. Deklaraatio päättyy aina puolipisteeseen, ja deklaraatiot erotetaan toisistaan aaltosuluilla. CSS-koodista saadaan selkeämpää laittamalla jokainen deklaraatio omalle rivilleen. Katso kuvio 7. (CSS Syntax, Viitattu 22.11.2014.)



KUVIO 7. (CSS Syntax, Viitattu 22.11.2014)

CSS-selektorit mahdollistavat HTML-elementtien valinnan ja niiden tyylittelyn. Selektorit löytävät muokattavan HTML-elementin esimerkiksi id:n, luokan, tyyppin tai attribuutin avulla. Id:tä ei saa koskaan nimetä alkavaksi numerolla. Tyylimäärittelyt voidaan myös kohdistaa pelkästään tietyn luokan alla oleviin elementteihin. Katso kuvio 8. (CSS Selectors, Viitattu 22.11.2014.)

```
/* Esimerkki CSS-selektoreista */
#kappale1 {
    text-align: center;
    color: red;
}

.center {
    text-align: center;
    color: red;
}

p.center {
    text-align: center;
    color: red;
}
```

KUVIO 8. (CSS Selectors, Viitattu 22.11.2014)

Ulkoisen tyylitiedoston käyttö on erittäin kätevää, kun halutaan vaikuttaa sen avulla moneen eri sivuun. Sen avulla voidaan muuttaa kaikkien haluttujen sivujen tyylittelyä tekemällä muutokset vain yhteen tyylitiedostoon. Tyylitiedosto täytyy tallentaa .css-päätteellä. Tyylitiedosto voidaan tehdä millä tahansa tekstinkäsittelyohjelmalla. Tyylitiedosto ei saa sisältää html-tageja. Tyylejä voidaan määrittellä HTML-elementtien sisällä, HTML-dokumentin head-tagien sisällä ja ulkoisessa tyylitiedostossa. Katso kuvio 9. (CSS How To..., Viitattu 25.11.2014.)

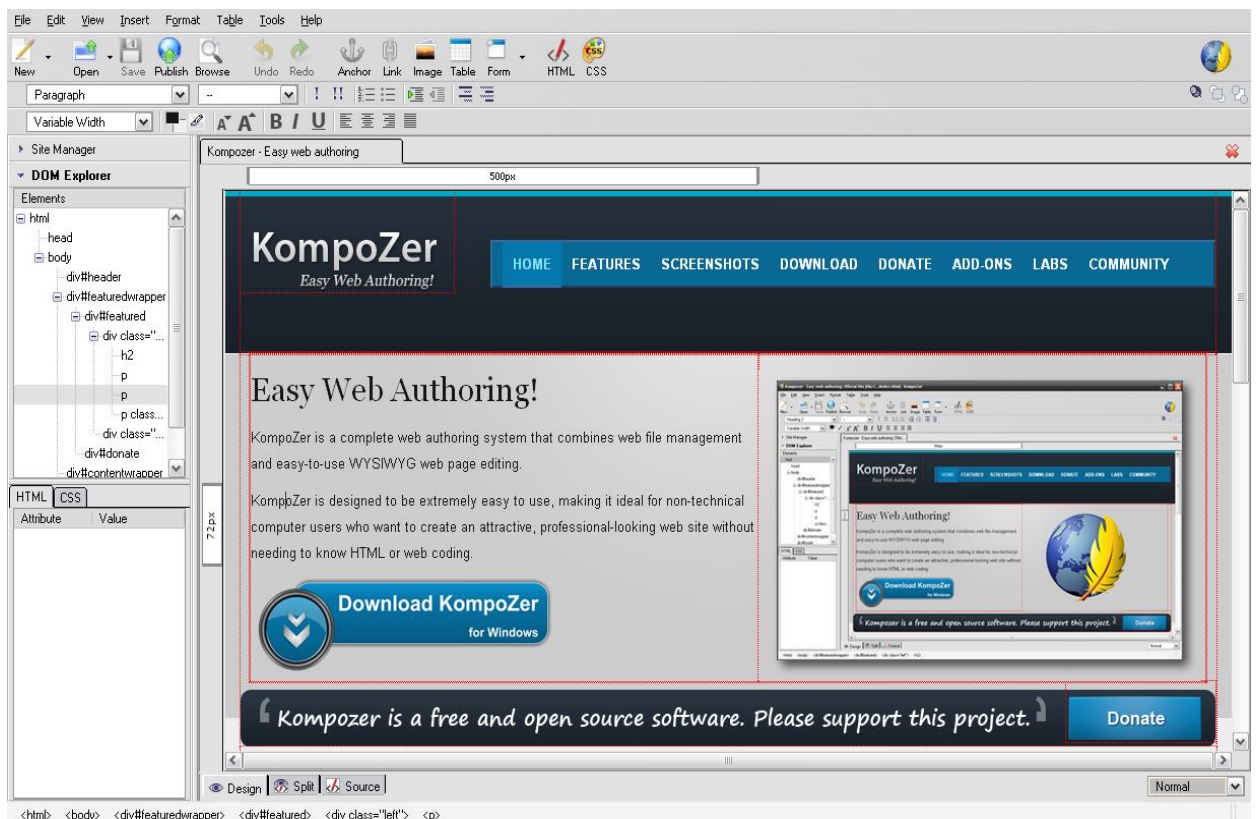
```
/* Esimerkki ulkoisesta tyylitiedostosta*/
<head>
<link rel="stylesheet" type="text/css" href="tyylitiedosto.css">
</head>
```

KUVIO 9. (CSS How To, Viitattu 25.11.2014)

2.3 HTML-editorit

HTML-editorit ovat ohjelmia, jotka ovat erikoistuneet tukemaan HTML- ja CSS-koodin kirjoittamista. Lähes kaikki web-sivustojen suunnittelijat käyttävät HTML-editoreja web-sivustojen ylläpitoon ja luomiseen, sillä hyvä HTML-editori nopeuttaa työskentelyä huomattavasti. HTML-editorilla on enemmän toiminnollisuuksia verrattuna tavalliseen teksti-editoriin. Esimerkiksi HTML-editori tekee automaattisesti lopputagin alkutagin kirjoittamisen jälkeen, mikä vähentää kirjoittamisen tarvetta.

WYSIWYG-editorit (What You See Is What You Get) renderoivat sivuston nähtäväksi web-editoriin samaan muotoon kuin se on webissä. Näiden editorien avulla web-sivuston elementtejä voidaan muokata tai liikuttaa yksinkertaisesti hiiren avulla. Voidaan esimerkiksi valita tietty osa tekstistä ja alleviivata se painamalla editorin alleviivauspainiketta, ilman minkäänlaista HTML- tai CSS-koodin kirjoittamista. Näillä editoreilla voidaan usein tehdä samanlaisia muokkauksia tekstiin kuten Microsoft Wordilla. Alla olevassa kuviossa 10 on esimerkki ilmaisesta WYSIWYG-editorista, nimeltään KompoZer.



KUVIO 10. (KompoZer, Viitattu 14.12.2014)

2.4 Web-selaimet

Saadakseen yhteyden web-sivuihin ihmiset käyttävät web-selaimia. Yleisimpiä selaimia ovat Firefox, Internet Explorer, Safari, Chrome ja Opera. Ihmiset pääsevät eri web-sivuille kirjoittamalla halutun osoitteen selaimen osoite-kenttään, klikkaamalla sivustolla olevia linkkejä tai käyttämättä kirjainmerkkejä. (Duckett 2011, 7.)

Web-selainten kehittäjät julkaisevat jatkuvasti uusia versioita selaimista. Uusissa versioissa on lisää ominaisuuksia ja lisäyksiä selainten kieleen. On erittäin tärkeää muistaa, että kaikki eivät kuitenkaan käytä näitä uusimpia versioita web-selaimissaan. Web-sivustoa tehdessä on muistettava, että vanhemmat versiot selaimista eivät välttämättä tue uusimpia web-sivuston ominaisuuksia ja toiminnollisuuksia. (Duckett 2011, 7.)

On myös olemassa web-selaimia, jotka on tehty pelkästään käytettäväksi mobiililaitteilla. Nämä selaimet on optimoitu näyttämään web-sivuston pienemmällä näytöllä tehokkaasti ja nopeasti, huolimatta mobiililaitteiden heikommasta suorituskyvystä. Mobiililaitteiden selaimet ovat yleisesti kevyempiä versioita tavallisista selaimista ja ne tarjoavat huomattavasti vähemmän ominaisuuksia tehokkaan selausnopeuden aikaansaamiseksi. (Web browser (browser). Viitattu 16.2.2015.)

2.5 JavaScript/jQuery

JavaScriptillä tehdään erilaisia toimintoja web-sivustoille. JavaScriptin avulla on mahdollista antaa käyttäjälle välitöntä palautetta, esimerkiksi kehoitus korjata jokin lomakkeen kenttä, jos se on virheellinen. JavaScript on kuitenkin yleiskäyttöinen ohjelmointikieli, joka ei ole mitenkään sidoksissa lomakkeisiin. (Korpela & Linjama 2005, 294.)

JavaScript on ohjelmointikieli, jota voidaan käyttää esimerkiksi selainskriptien tekemiseen. Oikein liitettyä JavaScriptiä HTML-dokumentissa mahdollistavat ohjelman suorittamisen. JavaScriptin lisäksi on useita muita selainskriptien tekemiseen tarkoitettuja kieliä, mutta JavaScript on ehdottomasti eniten selainten tukema. JavaScript-ohjelmilla voi tehdä monia erilaisia asioita, koska kielessä on paljon selainkäyttöön tarkoitettuja erityispiirteitä. (Korpela & Linjama 2005, 294 – 295.)

JQuery on ominaisuuksiltaan laaja JavaScript kirjasto. Sen avulla voidaan toteuttaa HTML-dokumentiin liikettä ja animointia. JQuery tukee suuria määriä selaimia. Sen monipuolisuus ja laajennettavuus ovat muuttaneet JavaScriptin käyttöä. (What is jQuery, Viitattu 25.11.2014.)

JQuery edistää sovellusten suorituskykyä ja auttaa kehittämään kaikilla web-selaimilla toimivan web-sivuston. Se auttaa toteuttamaan käyttöliittymälle olennaiset toiminnot ilman satoja rivejä koodia. JQueryä voidaan myös laajentaa ja muokata omaan käyttöön sopivaksi. JQueryä varten ei tarvitse opetella uusia syntakseja, JavaScriptin perusteiden osaaminen riittää. JQueryn koodi on yksinkertaista, ja sillä voidaan saavuttaa monimutkaisia toimintoja vähällä koodin määrällä. (What is jQuery and How to Start using jQuery? Viitattu 13.12.2014.)

Alle oleva kuvio 11 on otettu w3schools:n sivuilta. Se esittää yksinkertaisen esimerkin JQueryn käytöstä. JQueryn koodi on script-tagien sisällä ja se määrittää sen, että painettaessa html-dokumentin p-tagien sisällä olevaa tekstiä hiirellä teksti katoaa.

```
<!-- Esimerkki jQueryn käytöstä -->
<!DOCTYPE html>
<html>
<head>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
  $("p").click(function(){
    $(this).hide();
  });
});
</script>
</head>
<body>

<p>Jos painat minua katoan.</p>
<p>Paina minut pois!</p>
<p>Minut myös!</p>

</body>
</html>
```

KUVIO 11. (jQuery Tutorial, Viitattu 14.12.2014)

2.6 Bootstrap

Bootstrap on kokoelma ilmaisia työkaluja web-sivustojen ja web-ohjelmien luontia varten. Se sisältää HTML- ja CSS-mallipohjia tekstile, lomakkeille, painikkeille, navigoinnille ja muille komponenteille. Se sisältää myös valinnaisia JavaScript laajennuksia. Bootstrap on avointa lähdekoodia ja se on saatavissa GitHub:sta. (Bootstrap Introduction, Viitattu 26.11.2014.)

Bootstrapin ovat kehittäneet Mark Otto ja Jacob Thornton Twitterillä työskennellessään. Ennen Bootstrapia käytettiin useita eri kirjastoja käyttöliittymien kehittämiseen, mikä johti ristiriitaisuuksiin ja suuriin huoltotaakoihin. Bootstrap julkaistiin elokuussa 2011. (Bootstrap Introduction, Viitattu 26.11.2014.)

Bootstrap on rakennettu 12:sta responsiivisesta sarakkeesta, ulkoasuista ja komponenteista. Näytölaitteen koosta riippuen voidaan Bootstrapin avulla myös piilottaa tiettyjä osia web-sivustosta. Näitä määrittämiä voidaan tehdä käyttämällä luokkamäärittäjiä, esimerkiksi puhelimille, tableteille ja pöytäkoneille. Bootstrap sisältää useita valmiita komponentteja, kuten pudotusvalikkoja, navigointipalkkeja ja painikeryhmiä. Komponentit, kuten pudotusvalikot, ovat interaktiivisia useiden Bootstrapiin liitettyjen JavaScript-lisäosien kautta. Bootstrapin käyttäjälle on tarjolla laajat ohjeistukset ja demo-versioita, joten sen käytöstä on tehty mahdollista kaikille. (6 Reasons to Choose the Bootstrap CSS Framework, Viitattu 10.12.2014.)

Bootstrap 3:n myötä Bootstrap on sisältänyt mobiililaitteita suosivan lähestymistavan. Bootstrap tukee myös kaikkia tunnettuja web-selaimia. Bootstrapin CSS:n ansiosta se on responsiivinen pöytätietokoneilla, tableteissa ja mobiililaitteilla. HTML:ää ja CSS:ää tunteva voi aloittaa työskentelyn Bootstrapilla. (Bootstrap Introduction, Viitattu 26.11.2014.)

Alla oleva kuvio 12 esittää, kuinka Bootstrap otetaan käyttöön HTML5-dokumentissa. Head-elementissä on linkitettyä Bootstrapin tyyliä, JavaScript ja valinnainen tyyliä Bootstrapin valmiiseen teemaan.

```

<!-- Esimerkki Bootstrapin käyttöönotosta HTML5-dokumentissa -->
<!DOCTYPE html>

<html lang="fi">
<head>
  <!-- Latest compiled and minified CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap.min.css">
  <!-- Optional theme -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap-theme.min.css">
  <!-- Latest compiled and minified JavaScript -->
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/js/bootstrap.min.js"></script>
</head>

<body>

</body>
</html>

```

KUVIO 12. (Getting started, Viitattu 14.12.2014)

2.7 Responsiivisuus

Responsiivinen suunnittelu on tällä hetkellä web-suunnittelussa yksi ajankohtaisimmista puheenaiheista. Responsiivisen suunnittelun tarkoituksena on luoda yhdellä alustalla käytettävyydeltään hyvät näkymät useaan eri käyttötarkoitukseen. Samaa sovellusta tai web-sivua voi siis käyttää täysin samalla tavalla, oli sitten kyseessä kannettavatietokone, mobiililaitte tai tabletti. (Mitä on responsiivinen design? Viitattu 28.11.2014.)

Responsiivista suunnittelua kutsutaan myös usein mukautuvaksi verkkosuunnitteluksi tai mukautuvaksi suunnitteluksi. Nykyään on erittäin tärkeää, että web-sivut toimivat myös mobiililaitteilla, sillä mobiililaitteiden käyttö on kasvanut viime vuosien aikana. Mobiililaitteiden nettikäyttö tulee kasvamaan myös jatkossa. (Mitä on responsiivinen design? Viitattu 28.11.2014.)

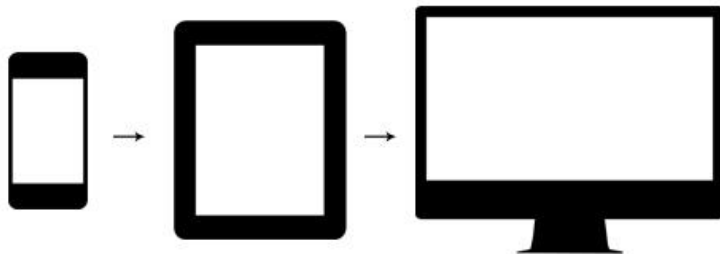
Responsiivinen suunnittelu voidaan toteuttaa CSS:n media queryjen avulla. Näiden media queryjen avulla määritellään breakpointeja, ja niiden kohdalla tietyt tyylit tulevat voimaan. Suurempi määrä breakpointeja kuitenkin aina kasvattaa ylläpidon määrää. Alla olevan kuvan tyylimäärite on voimassa, kun näyttö on korkeintaan 1200-pikseliä ja vähintään 800-pikseliä. Katso kuvio 13. (Mitä on responsiivinen design? Viitattu 28.11.2014.)

```
@media screen and (min-width: 800px) and (max-width: 1200px) {  
  .leftSidebar{  
    width: 40%;  
    float: left;  
  }  
}
```

KUVIO 13. (Timo Leiniö, Viitattu 28.11.2014)

Media queryja tuetaan laajasti uusissa web-selaimissa. Esimerkiksi Safari 3+, Chrome, Firefox 3.5+ ja Opera 7+ ja uusimpien mobiililaitteiden web-selaimet tukevat kaikki media queryja. Vanhemmat web-selaimen versiot eivät tue media queryja. (Responsive Web Design, Viitattu 4.12.2014.)

Ennen web-suunnittelijat ja heidän asiakkaansa ovat lähestyneet projekteja pöytätietokoneiden käyttäjien näkökulmasta, ja mobiililaitteiden käyttäjät jäivät toissijaisiksi. Vielä responsiivisen suunnittelun tärkeyden kasvetuakin, monet suunnittelevat projektinsa ensisijaisesti suurille näyttölaitteille. Alalla on kuitenkin kasvava trendi kääntää tämä ajattelutapa päinvastoin ja suunnitella projektit ensisijaisesti mobiililaitteille. Katso kuvio 14. (Mobile First Design: Why it's Great and Why It Sucks, Viitattu 10.12.2014.)



KUVIO 14. (Joshua Johnson, Viitattu 2013)

3 ULKOASUN SUUNNITTELU

Ulkoasua suunnitellessa lähdetään yleensä liikkeelle käyttäjäkunnan kartoittamisesta. Käyttäjäkunta vaikuttaa web-sivustolla käytettävään kuvitukseen, väreihin, tekstinmäärään ja tekstityyppiin. Myös sivustolle tulevien kuvien, tekstien ja muun grafiikan yhtenäistä tyyliä on mietittävä tarkkaan. (Korpela & Linjama 2005, 356.)

3.1 Värit

Web-sivuilla värien käytöstä tuli mahdollista vuosina 1993 – 1994. Tämä johtui näyttölaitteiden ja selainten kehittymisestä. Harmaasta ja mustasta siirryttiin käyttämään huomattavasti räikeämpiä värejä. Värien käytössä on kuitenkin käytettävä tarkkaa harkintaa hyvän lopputuloksen saamiseksi. Väärillä väreillä ja grafiikalla voidaan helposti pilata sivuston käytettävyys. Vaikka värit saataisiin keskenään sopimaan yhteen, voi värien sävy, käyttötapa tai esimerkiksi kylläisyys poiketa liikaa sivuston aihepiiristä. Taiteellisesti lahjatonkin voi oppia värien oikeanlaisen käytön web-sivuilla. On vain opeteltava värien käytön yleissääntöjä ja rajoituksia. (Korpela & Linjama 2005, 392.)

Tietokonenäytöt on tehty tuhansista pienistä neliöistä, joita kutsutaan pikseleiksi. Näytön ollessa päällä jokainen pikseli voi olla eri sävyinen punaisesta, sinisestä ja vihreästä väristä, jolloin ne muodostavat näytölle halutun kuvan. Jokaisella värin sävyllä (hue) on kaksi pääominaisuutta, kirkkaus (brightness) ja kylläisyys (saturation). Kylläisyys ilmoittaa, kuinka paljon värissä on harmaata. Kylläisyyden ollessa suurimmillaan ei värissä ole ollenkaan harmaata ja sen ollessa pienimmillään on väri melkein kokonaan harmaa. Kirkkaus ilmoittaa, kuinka paljon mustaa värissä on. Kirkkauden ollessa suurimmillaan värissä ei ole ollenkaan mustaa ja sen ollessa pienimmillään väri on erittäin tumma. (Duckett 2011, 251 – 252.)

Lähtökohtana värien käytössä on aina mietittävä, mitä väreillä halutaan sanoa. Väri on voimakas viesti ja sillä voi olla useita eri käyttötarkoituksia. Värit voivat järjestellä, erottaa, osoittaa tai korostaa haluttuja asioita. Värejä käyttämällä voidaan esimerkiksi keventää typografiaa, ja ne voivat toimia tunnistamisen apuna esimerkiksi yrityksen web-sivulla. Väreillä on symbolimerkityksiä, jotka ovat usein pitkän ajan kuluessa vakiintuneet, ja ne voivat perustua myös värin voimakuuteen ja

aktiivisuuteen. Sinistä väriä pidetään esimerkiksi etäännyttävänä, rauhoittavana ja viileänä. Punaista puolestaan pidetään kiihdyttävänä ja lähelle tulevana värinä. Keltainen on valovoimainen väri ja se mielletään iloiseksi, oranssi mielletään lämpimäksi ja vihreä rauhoittavaksi. (Korpela & Linjama 2005, 392.)

Värit vaikuttavat aina toisiinsa. Samat sävyt näyttävät erilaisilta erivärisellä taustalla. Myös värin pinta-alan koko vaikuttaa. Esimerkiksi pieni alue keltaista näyttää voimakkaammalta, kuin suurempi alue siniviolettiä. Värien käytöllä on mahdollista korostaa tiettyjä web-sivun osia. Esimerkiksi punainen kuva tai teksti muutoin mustavalkoisella web-sivulla saa katseen kiinnittymään siihen ensimmäisenä. Liian räikeä värien käyttö voi kuitenkin vaikeuttaa muun web-sivuston lukemista. Taustavärinä on hyvä käyttää himmeitä pastellivärejä ja tekstin väreinä vain vähän mustasta poikkeavia värejä. Näin kuvat saadaan paremmin oikeuksiinsa, koska tekstit ja tausta eivät ole liian häiritseviä. (Korpela & Linjama 2005, 392 – 393.)

Tekstin ja sen taustan värien valinta on myös tehtävä harkiten. Jos tekstin ja sen taustan värien kontrasti on liian heikko, teksti on silloin erittäin hankalaa luettavaa. Teksti on helpointa lukea, kun tekstin ja sen taustan värien kontrasti on suuri. Tästä hyvä esimerkki on valkoinen teksti täysin mustalla pohjalla. Liika kontrasti voi kuitenkin olla myös haitaksi, jos sivustolla on paljon luettavaa. Tällaisissa tilanteissa on parasta käyttää keskisuurta kontrastia, mistä hyvä esimerkki on valkoinen teksti tummanharmaalla pohjalla. Katso kuvio 15. (Duckett 2011, 253.)

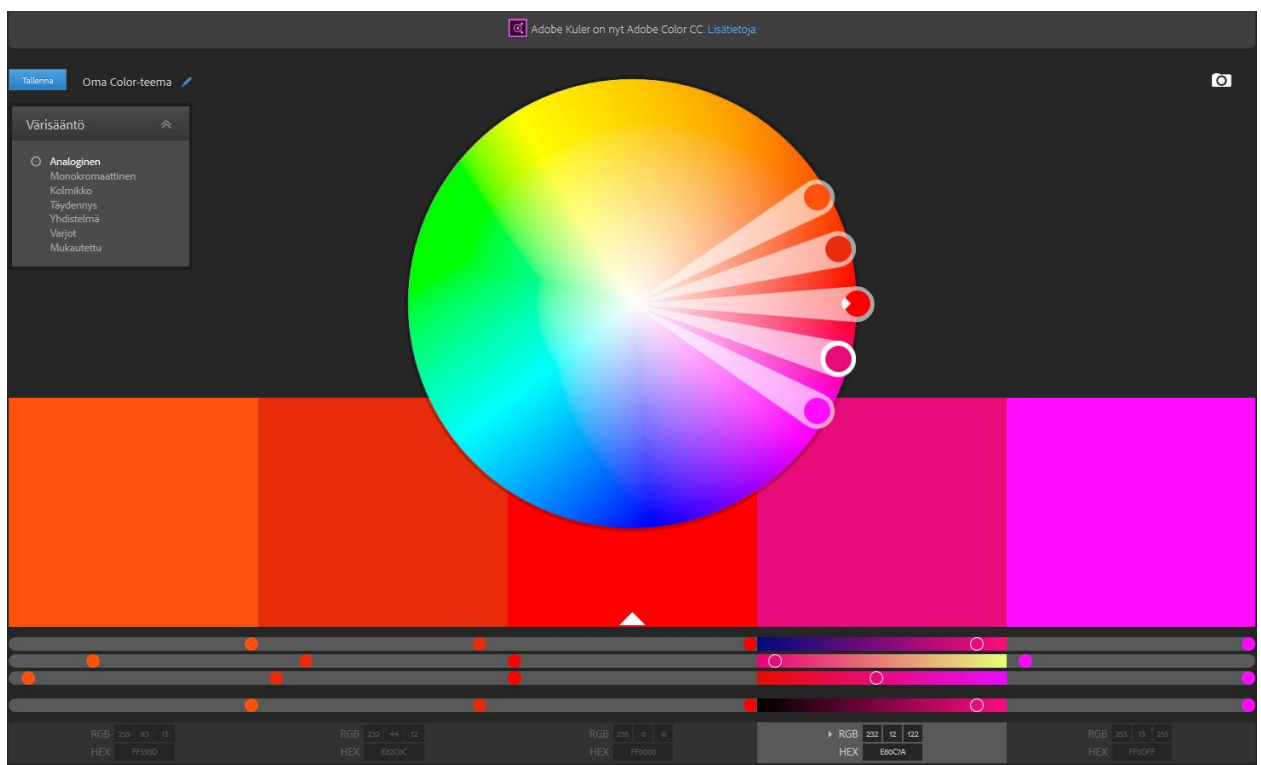


KUVIO 15. (Duckett 2011, 253)

Liian monien korostuskeinojen käyttöä kannattaa välttää. Tilanteen mukaan kannattaa valita yksi tai muutama korostuskeino. Värejä käytetään yleensä tukemaan tekstin sanomaa ja web-sivustolla käytettävää grafiikkaa, eikä huomion herättämiseksi. Jos sivustolla on vaikuttavaa grafiikkaa, olisi oheisväriyksen oltava hillitympi, jotta se ei ryöstä huomiota kuvilta. Yleissääntönä voidaan pitää, että sivustolla käytetään kolmea perusväriä. Nämä perusvärit voidaan ottaa esimerkiksi web-sivuston yrityksen logosta. (Korpela & Linjama 2005, 394.)

Värien valinta aloitetaan useimmiten suurimmasta ja hallitsevimmasta väripinnasta tai pakollisista väreistä, esimerkiksi yrityksen logon väreistä. On hyvä ottaa huomioon, että web-sivustolla käytettävät kuvat ja muu grafiikka tuovat sivulle lisää värejä. Vastavärejä on syytä käyttää harkiten, koska kaikki vastavärit eivät ole hyviä väryhdistelmiä. Kontrastivaikutuksen aikaansaamiseksi ne ovat kuitenkin erinomainen tehokeino. (Korpela & Linjama 2005, 395 – 397.)

Adobe Kuler on monipuolinen värinvalintaohjelma, joka on ollut käytettävissä internetissä jo vuodesta 2006. Adobe Kuler tarvitsee Flash-tukea toimiakseen. Adobe Kuler käyttää väriteemoja, jotka koostuvat viidestä eri väreistä. Adobe Kuler antaa jokaiselle värille oman värijärjestelmän koodin, joka tekee niiden käytöstä helppoa myös muualla. Kulerin avulla voidaan luoda kontrastia ja selvittää, kuinka värien eri sävyt toimivat keskenään. Katso kuvio 16. (Find The Perfect Color Scheme Every Time With Adobe Kuler, Viitattu 9.12.2014.)



KUVIO 16. (Adobe Color CC, Viitattu 9.12.2014)

3.2 Typografia

Typografialla tarkoitetaan kaikkien paino- tai digitaalisen töiden ulkoasuja ja niiden suunnittelua. Siihen kuuluvat esimerkiksi fontit, värit, asettelu, tasapainoinen kokonaisuus ja sen suhde sisältöön, harmonia, yhtenäisyys ja elementtien keskinäinen suhde. Typografia on merkkijärjestelmä, joka on sidoksissa tilanteisiin ja kulttuuriin. Sen vaikutus näkyy kaikissa kaksikulotteisissa pinnoissa esimerkiksi web-sivuilla, kirjoissa, sanomalehdissä ja mainoksissa. (Typografia, Viitattu 8.12.2014.)

Antiikvalla tarkoitetaan päätteellistä kirjasintyyliä. Siinä kirjaimet ikään kuin seisovat ohuen jalustan päällä, eri osilla on paksuuseroja ja kirjain liittyy päätteeseen pyöreästi ja sen muoto on soikio. Esimerkiksi Times New Roman ja Georgia ovat antiikva kirjasintyyliä. Katso kuvio 17. Groteski on päätteetön kirjasintyyli. Groteskissa kirjaimet ovat tikkukirjainmaisia, eikä paksuuseroja ole. Esimerkiksi Arial ja Verdana ovat groteski kirjasintyyppiä. Katso kuvio 18. (Typografia, Viitattu 10.12.2014.)



KUVIO 17. (Ulrika Juselius, Viitattu 10.12.2014)



KUVIO 18. (Ulrika Juselius, Viitattu 10.12.2014)

Leipätekstissä käytetään yleisesti selkeitä ja yksinkertaisia fontteja. Otsikot voivat olla enemmän tyylieltyjä. Perussääntönä voidaan pitää sitä, että leipätekstin ollessa päätteetön, on otsikon oltava päätteellinen tai päinvastoin. Leipätekstin fonttia valitessa tulee miettiä, millainen ulkoasu millekin sisällölle sopii. Web-sivustoilla varmasti toimivia fontteja ovat esimerkiksi Arial + Helvetica, Courier New ja Times New Roman. Tämä fonttivalikoima laajenee jatkuvasti. (Typografia, Viitattu 8.12.2014.)

Päätteettömät fontit ovat web-sivustoilla helppolukuisia, mutta päätteelliset näyttävät helposti luonnosmaisilta. Samalla web-sivustolla kannattaa käyttää korkeintaan kolmea erilaista fonttia. Useampien fonttien myötä teksti alkaa näyttämään sekavalta. Tärkeää tässäkin on selkeys, yhtenäisyys ja johdonmukaisuus. (Typografia, Viitattu 8.12.2014.)

Web-sivuston suunnittelussa kaiken materiaalin saatavuus alkuvaiheessa on erittäin tärkeää. Esimerkiksi jokainen kuva on oma kokonaisuutensa, ja kuvien sijoitus web-sivustolla riippuu niiden värikyksestä ja muodoista. On vaikeaa suunnitella web-sivusto väliaikaisten kuvien avulla, sillä oikealla kuvalla voi olla täysin erilainen olemus. (Typografia, Viitattu 8.12.2014.)

3.3 Kuvat

Web-sivustoa ensimmäistä kertaa katsoessa katsojan huomio kiinnittyy yleensä sivun linjoihin ja väreihin, näiden jälkeen katse kiinnittyy kuviin ja vasta viimeisenä teksteihin. Jos kuva on voimakas, voi se saada ensisijaisen huomion. Kuva on usein helpommin ymmärrettävissä kuin teksti. (Korpela & Linjama 2005, 202.)

Kuvien käyttötarkoitus voidaan karkeasti jakaa olennaisen asiasisällön esittämiseen ja kaikkeen muuhun. Asiasisällön esittämisessä voidaan käyttää korvaamatonta kuvaa, jonka tietoa on mahdotonta esittää sanallisesti. Asiasisällön esittämisessä voidaan myös käyttää sanallista esitystä tukevaa kuvaa, jolloin kuva korostaa tai havainnollistaa sanallista esitystä. (Korpela & Linjama 2005, 203 – 204.)

Kuvaa voidaan siis käyttää asiasisällön esittämisen lisäksi myös muissa tarkoituksissa. Kuvaa voidaan käyttää katseenvangitsijana. Tällöin kuva ei välttämättä ole tekemisissä tekstin sisällön kanssa, tämä on erittäin tyypillistä mainonnalle. Kuva voi olla tylsyyden katkaisija, sillä pelkkää tekstiä on tylsä lukea. Kuva voi myös toimia koristeena taiteellisen vaikutelman luomiseksi. Kuvaa voidaan käyttää merkinä, mistä ovat esimerkkinä kuvamuotoiset painikkeet. Yleinen käyttötarkoitus kuvalle voi olla myös sen toimiminen tunnuskuvana, kuten yrityksen logo. Kiinneohtana kuvaa käytetään lukijan selailun helpottamiseksi. Lukija löytää etsimänsä kohdan tekstistä kiinneohtan avulla. Kuva voi luoda vireen ja tunnelman, jolloin se auttaa lukijaa eläytymään tekstin maailmaan. (Korpela & Linjama 2005, 204 – 205.)

Tarpeellisen kuvituksen tunnistamisen avulla saadaan rajoitettua kuvien määrä kohtuulliseksi. Vähemmän olennainen kuvitus häiritsee sivuston olennaista kuvitusta, koska web-sivuston käyttäjän voi olla vaikeaa tunnistaa olennaiset kuvat. Sivuston tekijän on aina päätettävä, mihin tarkoitukseen kuvia käytetään. Jos kuvaa käytetään koristeena, se ei saa häiritä asiasisällön välittämistä. Tämä on erityisen tärkeää web-sivuston taustakuvissa. (Korpela & Linjama 2005, 208.)

Web-sivuston kuvissa on myös tärkeää käyttää oikeaa kuvaformaattia. Väärä formaatti voi saada kuvat näyttämään epäteräviltä, ja sivuston lataaminen saattaa hidastua. Yleisimpiä käytettyjä kuvaformaatteja ovat JPEG, GIF ja PNG. JPEG on oikea kuvaformaatti, jos kuvissa on paljon eri värejä kuten esimerkiksi maisemakuvissa. GIF ja PNG formaatteja käytetään, jos kuvissa on vain vähän eri värejä ja suuria alueita täysin samaa väriä. Näistä kuvista hyviä esimerkkejä ovat logot ja diagrammit. (Duckett 2011, 107 – 112.)

4 SIVUSTON TOTEUTUS

4.1 Toimeksiantaja

Opinnäytetyön toimeksiantaja on oululainen mediatoimista ja tuotantoyhtiö Rapid River, jonka omistaja on Kuusamosta kotoisin oleva Tauno Kohonen. Tauno Kohonen toimii Rapid Riverillä tuottajana ja kuvaajana.

Rapid Riverillä on paljon kuvagallerioita netissä useista eri teemoista ja paikoista. Kohosen ottamia kuvia näissä gallerioissa on erittäin paljon, ja niitä voi halutessaan käyttää maksua vastaan omissa projekteissaan. Rapid River suunnittelee myös asiakkaille kuvagallerioita ja web-sivustoja.

Rapid River tekee tilauksesta myös valokuvateoksia omasta aineistostansa. Näistä tunnetuin on palkittu Valoa erämaassa -teos. Tämä teos on käännetty seitsemälle eri kielelle. Rapid River on myös tehnyt useita muita kuvateoksia.

Videotuotanto on myös tärkeä osa Rapid Riverin toimintaa. Luontoa on kuvattu eri puolella Suomea, esimerkiksi Siperiassa Full-HD tasoisena videona. Videotuotannoissa hyödynnetään lyhytfilmiä ja kuvaesitysten yhteydessä opittuja taitoja ja kerrotaan tarinaa selkeästi ja tiivistetysti. Tälläkin hetkellä tekeillä on suuren luokan videotuotantoa Yleisradion kanssa.

Rapid River on pitkän uransa aikana järjestänyt myös useita valokuvanäyttelyitä. Näissä näyttelyissä esitellään Tauno Kohosen ottamia valokuvia. Ensimmäinen Tauno Kohosen valokuvanäyttely, Valoa erämaassa, on ollut tähän mennessä laajin kansainvälinen valokuvanäyttelykierrätys Suomesta ulkomaille.

4.2 Sivuston esittely

4.2.1 Opinnäytetyön aihe

Opintäytetyön tarkoituksena oli luoda Siperia-projektille esittelysivusto Tauno Kohosen antaman ohjeistuksen mukaan. Kaikki sivustolla käytetty kuva- ja videomateriaali on Tauno Kohosen kuvamaa. Kuvattu materiaali on Venäjän Siperian Kamchatkalta, Altaylta, Baikaliilta, Yeniseiltä ja Yakuatialta. Sivusto toimii introna varsinaiselle Siperia-projekti -sivustolle. Koko Siperia-projektin tarkoituksena on luoda TV-tuotanto Siperiasta. Opinnäytetyön valmistuessa sivusto siirrettiin Rapid Riverin omalle palvelimelle.

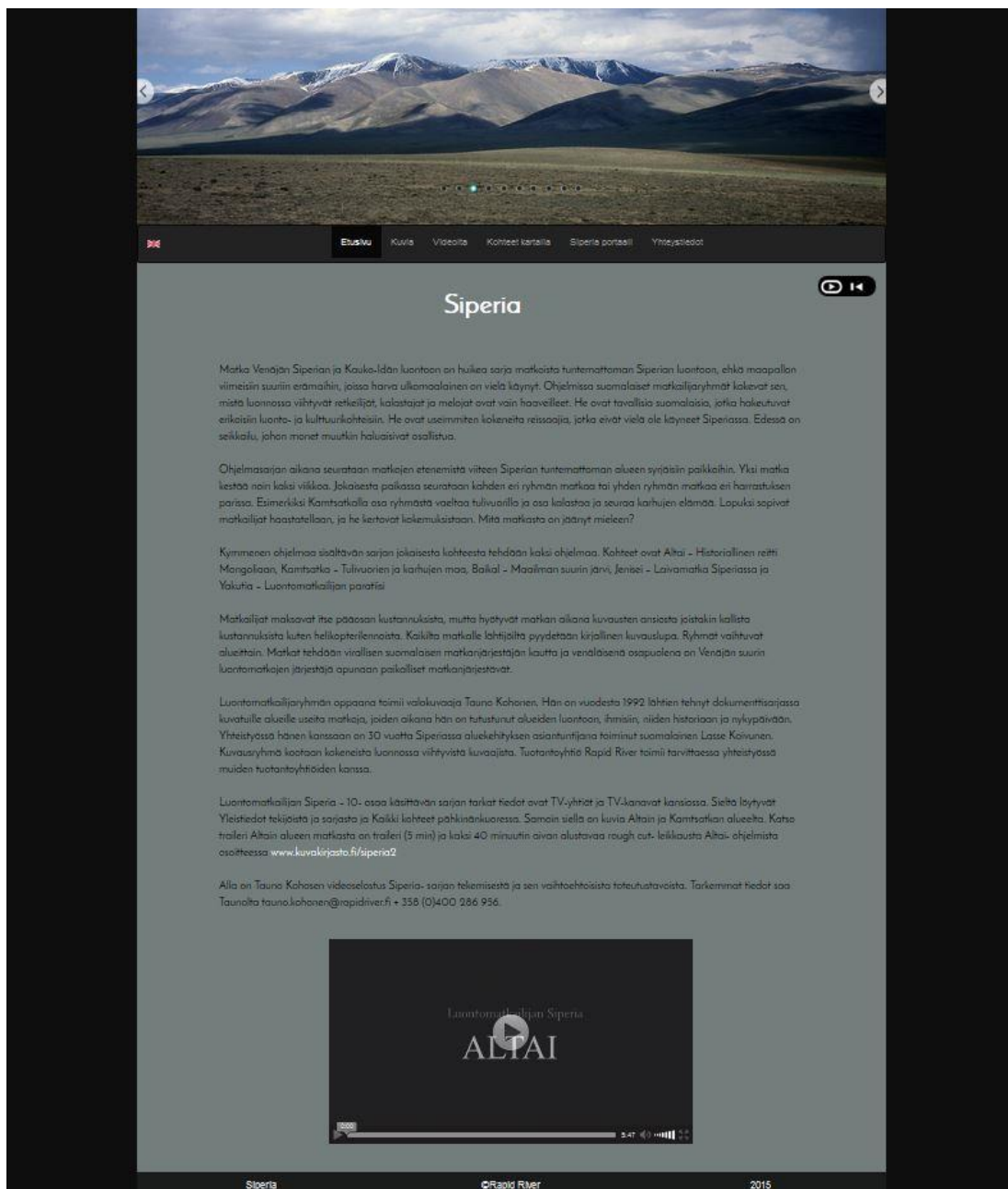
Sivusto on toteutettu käyttäen HTML5-tekniikkaa. Responsiivisuuden aikaansaamiseksi sivusto on rakennettu Bootstrapia ja sen lisäosia käyttäen. Kaikki sivustolla olevat kuva-, musiikki-, kartta- ja videositykset ovat tehty valmiiden JavaScript-, HTML5- ja jQuery-lisäosien avulla.

4.2.2 Sivuston rakenne

Sivuston rakenne koostuu viidestä eri sivusta: etusivu, kuvia, videoita, kohteet kartalla ja yhteystiedot. Navigointipalkissa on myös suora linkki varsinaiseen Siperia-portaaliin. Sivuston navigointipalkki on toteutettu Bootstrapin responsiivisen navbar-komponentin avulla. Koko sivusto on toteutettu Bootstrapin gridin pohjalle, koska rakenne on silloin helppoa pitää selkeänä ja responsiivisena. Navigointipalkista on myös lippua painamalla mahdollista vaihtaa koko sivuston kieli suomesta englantiin tai päinvastoin.

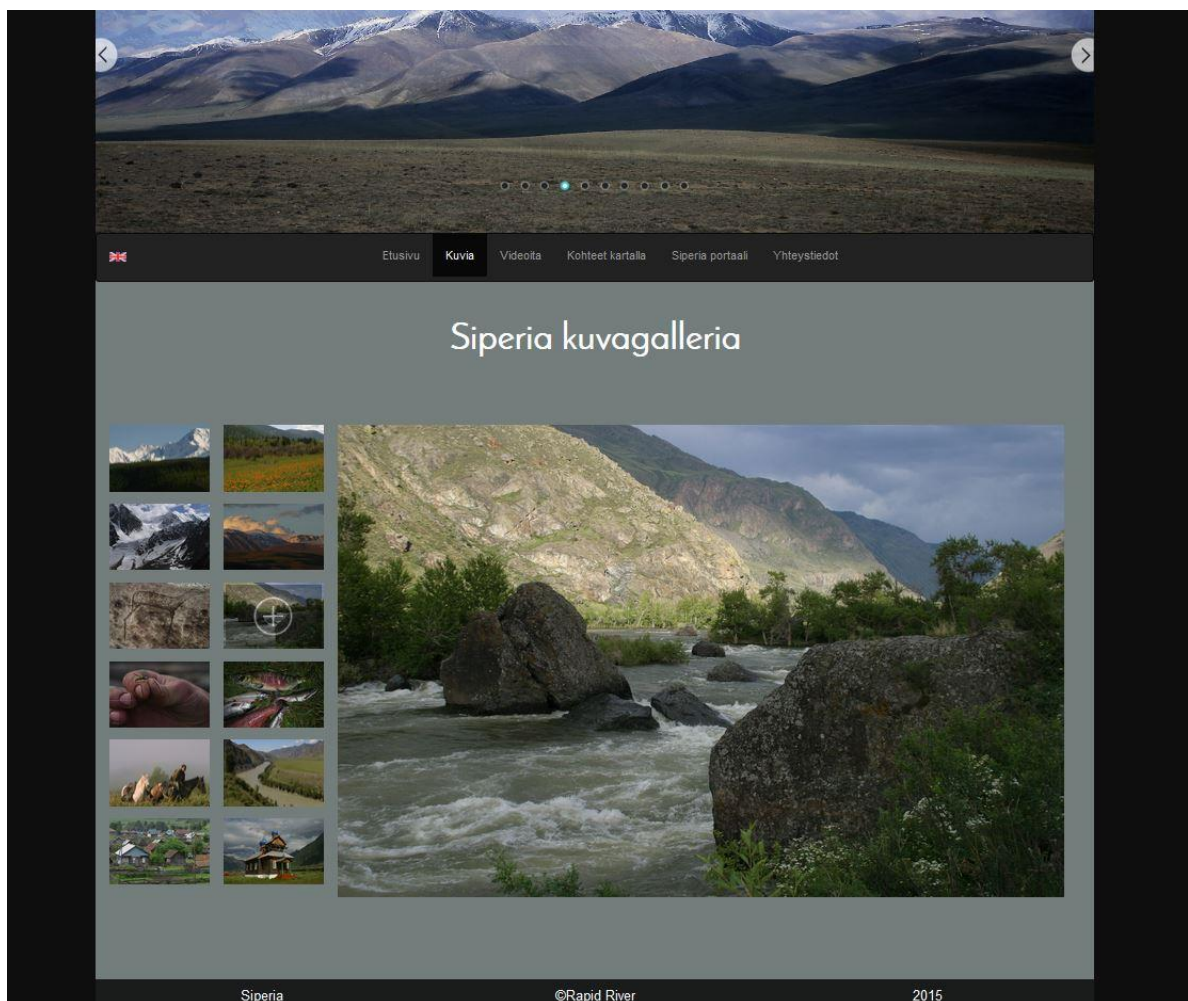
Jokaisen sivun bannerissa on käynnissä kuvaesitys, joka koostuu kymmenestä Siperiassa kuvattusta panoraamakuvasta. Päädyimme tähän kuvamäärään, koska kyseessä on kuitenkin vain esittelysivusto, jonka tarkoitus on välittää katsojalle tietoa tiivistetysti Siperia-projektista. Nämä kuvat antavat sivuston katsojalle nopeasti käsityksen projektiin kuvatuista näyttävistä luontokuvista, ja sen takia sama kuvaesitys toistuu kaikilla eri sivuilla. Jokaisella sivulla on myös footer, jossa on projektin nimi, yrityksen nimi sekä toteutusvuosi.

Etusivu koostuu Tauno Kohosen kirjoittamasta introsta Siperia-projektiin. Intron alla on video, jossa Tauno Kohonen kertoo Siperia-projektista. Etusivun taustamusiikkina soi siperialaisen kurkkulaulajan musiikkia, joka on lisätty sivustolle toimeksiantajan toiveesta. Kurkkulaulajan musiikki on Tauno Kohosen itse äänittämää hänen Siperia-matkansa aikana. Etusivustolla olevasta tekstistä selviää, mitä varten sivusto on tehty ja tietoa esittelysivuston rakenteesta. Etusivun rakenne on pidetty erittäin yksinkertaisena, jotta sivuston olennaisin tieto saadaan välitettyä lukijalle mahdollisimman selkeästi ja vaivattomasti. Katso kuvio 19.



KUVIO 19.

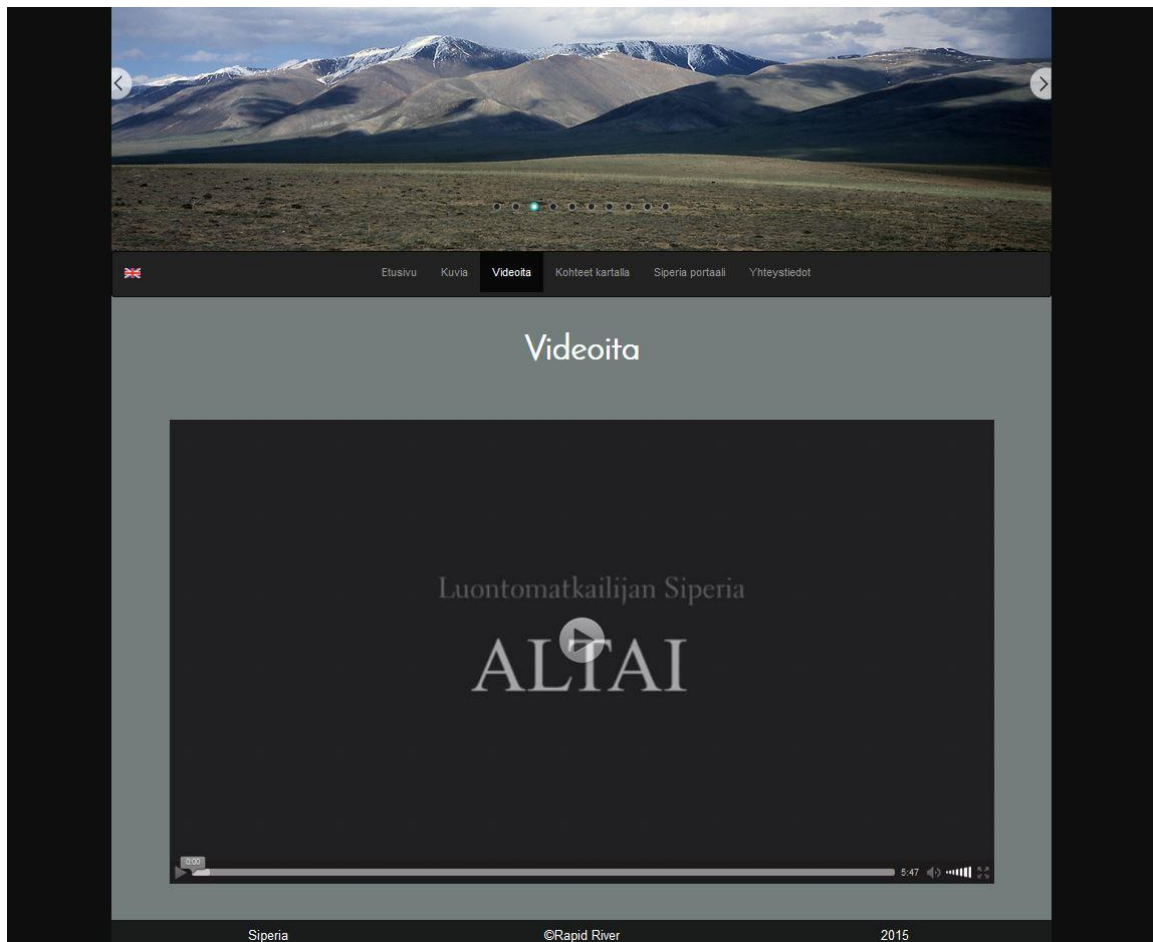
Kuvia-sivulla käytetty kuvagalleria on tehty Jssor foundationin jQuery-lisäosan avulla. Kuvagalleria sisältää 16 kuvaa eri Siperian alueilta. Kuvagalleriassa käytetyt kuvat on valinnut opinnäytetyön toimeksiantaja Tauno Kohonen. Kuvagallerian kuvat vaihtuvat automaattisesti, mutta niitä voi myös vaihtaa manuaalisesti. Kuvien vaihtumisen efektinä on slideout, jossa vanha kuva himmenee vähitellen uuden tullessa taustalta sen tilalle. Tämä efekti on valittu sivulle, koska se on yksinkertainen, mutta samalla näyttävä. Koska kyseessä on vain Siperia-projektin esittelysivusto, tämä lisäosa riittää hyvin kuvamateriaalin esittelyyn. Katso kuvio 20.



KUVIO 20.

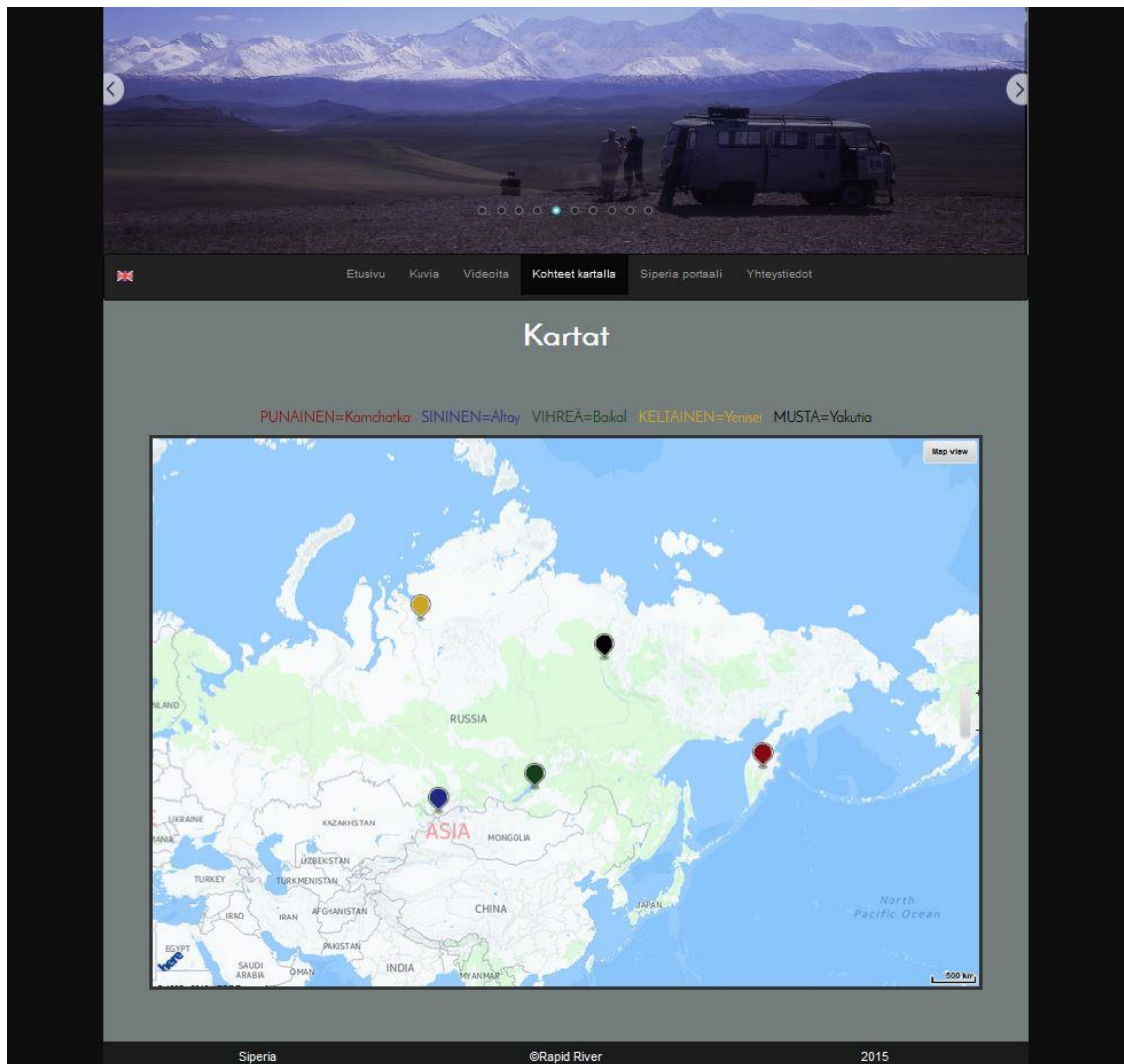
Videoita-sivun rakenne on mahdollisimman yksinkertainen ja samalla selkeä. Sivustolla on ainoastaan HTML5:n videosoitin, jonka avulla voidaan katsoa videoita suuressa koossa. Videoita voidaan myös katsoa kokoruudulla ja videosoitin on täysin responsiivinen. Sivulla on tällä hetkellä ainoas-

taan Siperia-projektia varten tehty traileri, mutta uusia videoita voidaan lisätä jatkossa. Tauno Kolehmainen on kuvannut yhdessä Kari Sorsan kanssa videolla nähdyt kohtaukset kesällä 2010. Katso kuvio 21.



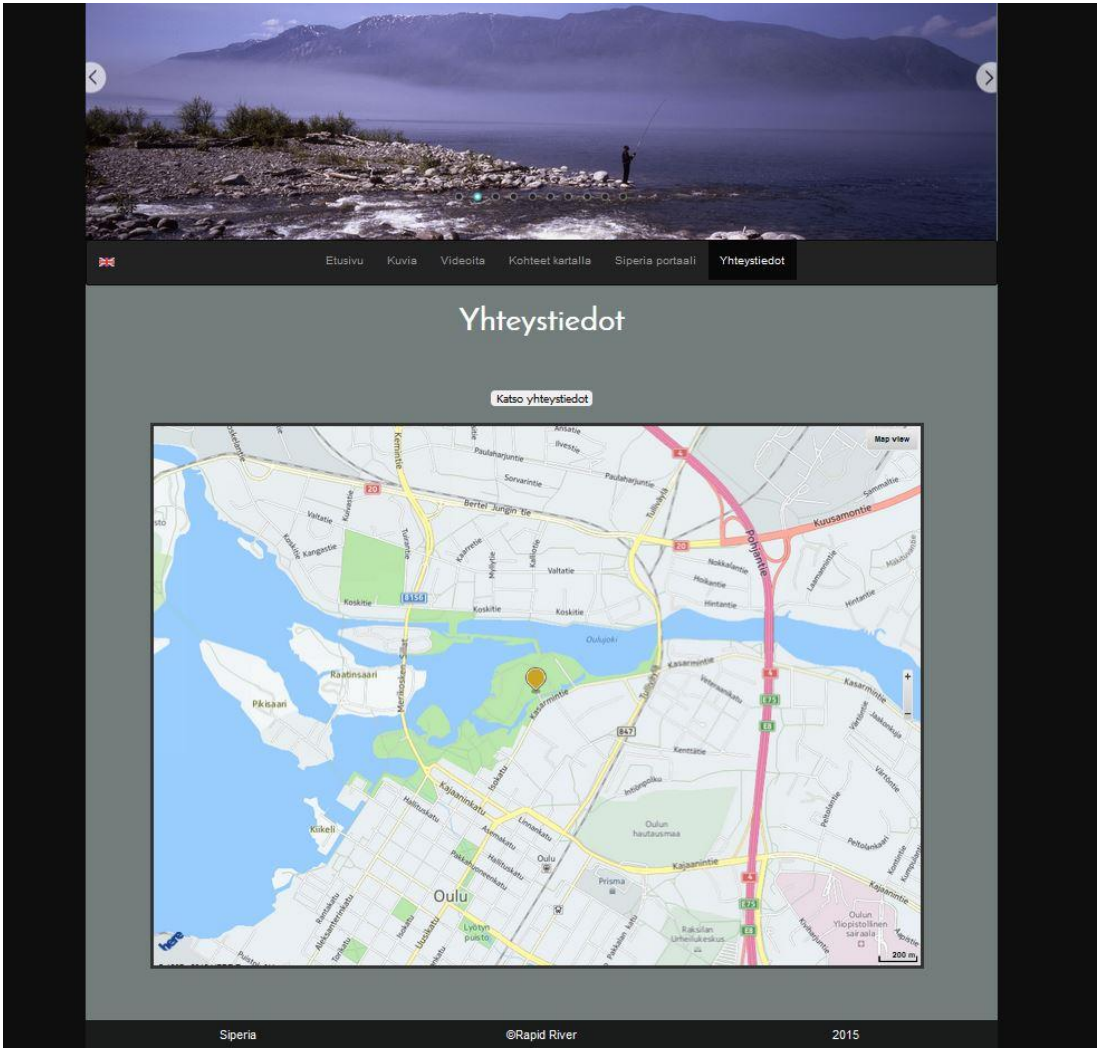
KUVIO 21.

Kohteet kartalla -sivu koostuu kartasta, johon on merkattu kaikki Siperia-projektia varten kuvatut kohteet: Kamchatka, Altay, Baikal, Yenisei sekä Yakutia. Kartan päälle on kirjoitettu kohteet väreittäin, ja tällä kyseisellä värikoodilla yksittäinen paikka on merkitty karttaan markerilla. Mielestämme tällä tavalla on helppo havainnollistaa Siperia-projektiin kuvattujen paikkojen sijainti kartalla. Karttana on Nokian ilmainen HERE-kartasto, koska sen käyttöönotto oli varsin vaivantonta ja täysin ilmaista. Kartan näkymäksi voidaan valita tavallisen karttanäkymän lisäksi satelliittinäkymä tai maastonäkymä. Katso kuvio 22.



KUVIO 22.

Viimeinen eli Yhteystiedot-sivu koostuu samasta Nokian HERE-kartasta, johon on merkitty Rapid Riverin toimipisteen sijainti Oulussa. Sivustolla on myös Katso yhteystiedot -painike, jota painamalla aukeaa uusi ikkuna, jossa on yrityksen henkilöiden yhteystiedot. Tämä toiminto on toteutettu jQuery-lisäosan avulla. Tämän lisäosan ja kartan avulla saadaan selkeästi esitettyä, missä Rapid Riverin toimipiste on, ja keitä siellä työskentelee. Katso kuvat 23 ja 24.



KUVIO 23.



KUVIO 24.

4.2.3 Sivuston graafinen ilme

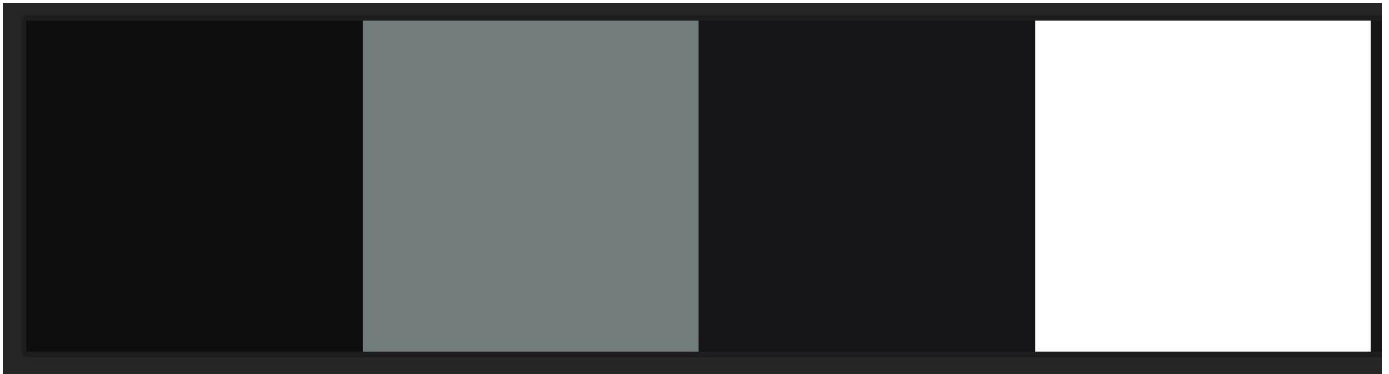
Sivuston pääväreinä ovat koko sivuston erittäin tumman harmaa (#0e0e0e) taustaväri, sisällön harmaa taustaväri (#727D7C), nav-barin ja footerin tumman harmaa väri (#161617) ja tekstien valkoinen väri (#ffffff) ja erittäin tumman harmaa (#0e0e0e) väri. Alla oleva kuvio on tehty Adobe Kulerilla. Katso kuvio 25.

#0e0e0e

#727D7C

#161617

#ffffff



KUVIO 25.

Sivuston päävärejä valittaessa tavoitteena oli sivuston selkeän ja yhtenäisen ulkoasun luominen. Sivuston pääväreiksi valittiin vain nämä neljä pääväriä, jotta lopputulos olisi hillitty. Värit pyrittiin pitämään mahdollisimman neutraaleina, mutta kuitenkin teemaan sopivina. Liian räikeät värit antaisivat helposti sivustolle teemaan sopimattoman liian hyökkäävän ilmeen. Sivuston päävärit ovat sävyiltään lähellä toisiaan, lukuun ottamatta valkoista, jota käytettiin tehostekeinona osalle teksteistä.

Sivuston eri osat on pyritty erottelamaan toisistaan värejä käyttämällä. Esimerkiksi nav-bar ja footer erotetaan harmaasta sisältöosasta harmaan tummemmalla sävyllä. Värejä on myös käytetty ohjaamaan sivuston lukijan katsetta haluttuun suuntaan. Sivuston tumman yleisilmeen takia lukija katse kiinnittyy helpommin vaalean harmaalla taustalla olevaan sisältöosuuteen. Sisältöosuuden taustaväriä valittaessa otettiin myös huomioon, että musta teksti on helposti luettavissa neutraalilta taustalta.

4.3 Käyttöönotto

Valmis toimeksiantajan hyväksymä sivusto siirrettiin Rapid Riverin omalle palvelimelle. Sivusto testattiin siirtämisen jälkeen kaikilla yleisimmillä tietokoneiden selaimilla (Mozilla Firefox 36, Internet Explorer 11 ja Google Chrome 40) ja muutamalla hyvin yleisellä mobiililaitteella (Apple iPhone 4S, Samsung Galaxy S2 ja LG G2). Viimeiset korjaukset sivustolle tehtiin tässä vaiheessa. Sivusto siirrettiin palvelimelle ilmaisella WinSCP SFTP-ohjelmalla.

Sivuston siirron ja virheiden korjauksen jälkeen toimeksiantajaa ohjeistettiin sivuston päivittämisestä. Toimeksiantajalle kerrottiin ja koulutettiin sivuston perustoiminnallisuudet, kuten kuva- ja videogallerian päivittäminen ja uusien tekstien lisääminen. Toimeksiantajalle neuvottiin myös, millaiseen muotoon kuvat ja videot kannattaa muuntaa ennen palvelimelle siirtämistä.

4.4 Haasteet

Sivuston toteutus HTML5-tekniikalla oli molemmalle tämän opinnäytetyön tekijälle lähes kokonaan tuntematonta entuudestaan. Tämän takia opinnäytetyön tekninen toteutus oli kokonaisuutena opettavaista. Lisäosien käyttöönotto ja niiden yhteensopivuuden toteuttaminen oli myös aluksi vaikeaa. Sivuston kaikkien eri osien responsiivisuuden toteuttaminen oli myös erittäin haasteellista.

Pienenä ongelmana oli myös yhteisen ajan löytäminen opinnäytetyön tekemiseen. Teimme aina yhdessä opinnäytetyötä, jotta molemmat oppisivat mahdollisimman paljon. Toisena ongelmana oli ajoittainen hidas materiaalin saaminen toimeksiantajan kiireiden takia. Ulkoasun suunnittelu aloittaminen oli hankalaa ilman sivustolla käytettävää lopullista materiaalia.

Graafisen ilmeen suunnittelu ja tyylikkään lopputuloksen saaminen oli myös hyvin paljon aikaa vievää. Emme olleet aikaisemmin suunnitelleet web-sivuston ulkoasua alusta loppuun saakka. Sopivien fonttien ja värien valinta sivustolle oli ongelmallista ja haastavaakin, sillä sivustolla käytettävän värimaailman ja fonttien tuli sopia hyvin yhteen saadun materiaalin kanssa, ja sivuston ulkoasun tuli kuvastaa Rapid Riverin imagon mukaista teemaa.

Koska molemmat työn tekijät ovat suuntautuneet opiskeluissaan internet-palveluihin ja digitaaliseen mediaan, yksinkertaisenkin koodin tekeminen oli välistä hankalaa. Olemme koulussa opiskelleet ainoastaan peruskurssit HTML-sivustojen toteutuksesta. Virheiden löytäminen ja niiden korjaaminen koodista oli meille haaste itsessään. Nykyisin kaikilla Oulun ammattikorkeakoulun tietojenkäsittelyn opiskelijoilla on pakollisia web-teknologioihin liittyviä kursseja, joten koodin tekeminen on helpompaa uusille opiskelijoille.

5 POHDINTA

Opinnäytetyön valintaprosessi lähti liikkeelle harjoittelun aikana Rapid Riverillä. Tauno Kohonen kertoi useasti hänen Siperia-projektistaan, joka oli jäänyt keskeneräiseksi muiden projektien takia. Tätä projektia oli helppo jatkaa, koska jo olemassa oleva materiaali oli runsasta. Opinnäytetyön teko Rapid Riverille oli meille molemmille hyvä jatkumo harjoittelulle. Tämän takia Tauno Kohonen tarjosikin esittelysivuston luomista meille opinnäytetyön aiheeksi.

Sivustolla käytettävä materiaali valittiin Tauno Kohosen ohjeiden pohjalta. Osa materiaalista täytyi kuitenkin karsia pois, koska sivusto toimisi vain introna Siperia-projektille. Sivuston ulkoasu toteutettiin yhdessä toimeksiantajan kanssa. Teimme aluksi mallipohjan, jota muokattiin yhdessä toimeksiantajan toiveiden mukaiseksi. Tämä järjestely toimi hyvin, ja näin ulkoasu sai lopullisen muotonsa vaivattomasti.

Ulkoasun värit pyrittiin pitämään neutraaleina, jotta katse saadaan kiinnittymään olennaiseen eli Siperia-projektin materiaaleihin. Sivustolla käytetyt lisäosat kuvien esittelyyn ovat visuaalisesti näyttäviä, mutta eivät kuitenkaan liian raskaita selaimelle. Mielestämme sivuston ulkoasusta tuli visuaalisesti houkutteleva ja onnistunut.

Sivuston rakenne toteutettiin myös yhdessä Tauno Kohosen kanssa. Navigointi sivustolla on tehty mahdollisimman yksinkertaiseksi. Sivustolla on heti bannerin alapuolella navigointipalkki, josta pääsee vaivattomasti eri sivuille. Mielestämme sivuston rakenteesta tuli hyvin käyttäjäystävällinen ja tarpeeksi selkeä esittelysivustoksi.

Sivusto toteutettiin HTML5-tekniikalla ja jQuery, Bootstrap ja JavaScript-lisäosien avulla. Sivusto olisi ollut helpompi tehdä julkaisujärjestelmällä, mutta tällöin emme olisi oppineet läheskään yhtä paljon nettisivuston luomisesta. Halusimme oppia opinnäytetyötä tehdessä mahdollisimman paljon uutta, ja siksi loimme sivuston meille uutta HTML5-tekniikkaa käyttäen.

Opinnäytetyön parissa työskentely oli erittäin mielenkiintoista ja uuden oppiminen palkitsevaa. Itse tekemällä opimme erittäin paljon uutta lyhyessä ajassa. Sovelsimme koulussa opittuja taitoja opinnäytetyötä tehdessä. Opinnäytetyö valmistui ajallaan tiukasta aikataulusta huolimatta, koska teimme

aktiivisesti yhdessä töitä toimeksiannon eteen, minkä vuoksi myös halusimme onnistua hyvin. Molempien motivaatiotaso opinnäytetyön tekoon oli hyvin korkealla lähestyvän valmistumisen takia.

Lähtökohdat huomioon ottaen onnistuimme mielestämme kiitettävästi. Lopullinen web-sivusto vastasi toimeksiantajan toiveita ja odotuksia. Saimme sivustosta responsiivisen ja löysimme yhteensopivat lisäosat, jotka toimivat moitteettomasti. Toimeksiantaja oli tyytyväinen lopputulokseen ja otti heti sivuston käyttöönsä.

LÄHTEET

Bootstrap. 2014. Getting started. Viitattu 14.12.2014. <http://getbootstrap.com/getting-started/#download>.

CSS BASICS. 2014. Chapter 1 : Introduction to CSS. Viitattu 22.11.2014. <http://www.cssbasics.com/introduction-to-css/>.

CODE PROJECT. 2011. What is jQuery and How to Start using jQuery? Viitattu 13.12.2014. <http://www.codeproject.com/Articles/157446/What-is-jQuery-and-How-to-Start-using-jQuery>.

DEV HQ. 2012. HTML & CSS: Versions of HTML and CSS. Viitattu 22.11.2014. <http://www.devhq.net/html-css/19--versions-of-html-and-css>.

Duckett, J. 2011. HTML&CSS design and built websites. Indianapolis: John Wiley & Sons, Inc.

Ethan Marcotte. 2014. Responsive Web Design. Viitattu 4.12.2014. <http://alistapart.com/article/responsive-web-design>.

Gary Marshall. 2011. HTML5: what is it? Viitattu. 13.12.2014. <http://www.techradar.com/news/internet/web/html5-what-is-it-1047393>.

Harish Chouhan. 2013. 6 Reasons to Choose the Bootstrap CSS Framework. Viitattu 10.12.2014. <https://www.ostraining.com/blog/coding/bootstrap/>.

Joshua Johnson. 2013. Mobile First Design: Why It's Great and Why It Sucks. Viitattu 10.12.2014. <http://designshack.net/articles/css/mobilefirst/>.

KomPozer. 2014. Screenshots. Viitattu 14.12.2014. <http://www.kompozer.net/screenshots.php>.

Korpela, J. & Linjama, T. 2005. Web-suunnittelu. Jyväskylä: Docendo Finland.

Quackit. 2014. What Is An HTML Editor. Viitattu 15.12.2014. http://www.quackit.com/html/html_editors/what_is_an_html_editor.cfm.

Shannon, R. 2014. The History of HTML. Viitattu 14.11.2014. <http://www.yourhtmlsource.com/start-here/historyofhtml.html>.

The jQuery Foundation. 2014. What is jQuery? Viitattu 25.11.2014. <http://jquery.com/>.

Tim Brookes. 2013. Find The Perfect Color Scheme Every Time With Adobe Kuler. Viitattu 9.12.2014. <http://www.makeuseof.com/tag/find-the-perfect-color-scheme-every-time-with-adobe-kuler/>.

Timo Leiniö. 2014. Mitä on responsiivinen design? Viitattu 28.11.2014. <http://www.sofokus.com/blogi/mita-on-responsiivinen-design/>.

Ulrika Juselius. 2013. Typografia. Viitattu 8.12.2014. <http://www.phpoint.fi/ulrikaj/www/typo.htm>.

Vangie Beal. 2015. Web browser (browser). Viitattu 16.2.2015. <http://www.webopedia.com/TERM/B/browser.html>.

W3Schools. 2014. Bootstrap Introduction. Viitattu 26.11.2014. http://www.w3schools.com/bootstrap/bootstrap_intro.asp.

W3Schools. 2014. CSS How To. Viitattu 25.11.2014. http://www.w3schools.com/css/css_howto.asp.

W3Schools. 2014. CSS Selector. Viitattu 22.11.2014. http://www.w3schools.com/css/css_selectors.asp.

W3Schools. 2014. CSS Syntax. Viitattu 22.11.2014. http://www.w3schools.com/css/css_syntax.asp.

W3Schools. 2014. HTML5 Introduction. Viitattu 14.12.2014. http://www.w3schools.com/html/html5_intro.asp.

W3Schools. 2014. HTML Attributes. Viitattu 22.11.2014. http://www.w3schools.com/html/html_attributes.asp.

W3Schools. 2014. HTML Elements. Viitattu 20.11.2014. http://www.w3schools.com/html/html_elements.asp.

W3Schools. 2014. jQuery Tutorial. Viitattu 14.12.2014. <http://www.w3schools.com/jquery/default.asp>.