

RESPONSIIVINEN WEB DESIGN

TEKIJÄ/T:

Niko Kauppinen

Koulutusala Kulttuuriala	
Koulutusohjelma Viestinnän koulutusohjelma	
Työn tekijä(t) Kauppinen Niko	
Työn nimi Responsiivinen web Design	
Päiväys 15.09.2013	Sivumäärä/Liitteet 67 + DVD (verkkosivu-tiedostot)
Ohjaaja(t) Jouni Valtonen	
Toimeksiantaja/Yhteistyökumppani(t) --	
<p>Tiivistelmä</p> <p>Opinnäytetyössä tutustutaan responsiivisen verkkosuunnitteluun ja määritellään sen sisältöjä niin teoriassa kuin käytännön kautta. Työn tavoitteena on selkeä ymmärrys siitä, mitä responsiivisuus käytännössä tarkoittaa ja mitä tekniikkoja siihen sisältyy. Aluksi tutustutaan tekniikoihin, joihin responsiivinen suunnittelu nojautuu. Näitä ovat HTML5, CSS3 sekä sivuavasti Javascript. Tavoitteena ei ole rakentaa koodin kirjoittamiseen keskittyvää opusta, vaan keskittyä responsiivisen suunnittelun käytänteisiin ja suunnittelufilosofiaan. Opinnäytetyön lopputuotokseksi kootaan demoverkkosivu, jonka rakentamisessa käytetään HTML5:n ja CSS3:n uusia ominaisuuksia. Sivusta tehdään responsiivinen.</p> <p>Opinnäytetyön pohja-aineistona toimivat pääasiassa alan foorumeilta kerätyt käytänteet, artikkelit ja blogit. Pohjalla on myös alan kirjallisuutta, joskin vähäisemmässä määrin. Opinnäytteen kirjoitusasu olettaa, että lukijalla on ainakin perustava tuntemus verkko-ohjelmoinnista.</p>	
Avainsanat responsiivinen suunnittelu, HTML5, CSS3, taittopohjat	

Field of Study Culture	
Degree Programme Degree Programme in Communication Design	
Author(s) Kauppinen Niko	
Title of Thesis Responsive web design	
Date 15.09.2013	Pages/Appendices 67 + DVD(website-files)
Supervisor(s) Jouni Valtonen	
Client Organisation/Partners --	
<p>Abstract</p> <p>In this thesis we explore the meaning and definition of responsive web design via theory and practice. The goal of the thesis is to achieve clear understanding of responsive techniques and their common usage. In the beginning we get familiar with the main techniques usually responsible for responsivity in websites. These are mainly HTML5, CCS3 and a bit of Javascript on the side. It is not our intention to create a coding guidebook but a tutorial on features of responsive design and its design philosophy. As the final outcome we compile a new demosite, in the building of which we use new properties introduced in HTML5 and CCS3. The site will be responsive.</p> <p>The thesis is mostly based on information gathered from web forums, blogs and articles from the sites meant for web design developers. There are also literary sources, though their significance is lesser. The wording in thesis assumes at least basic understanding of web-programming/development from the reader.</p>	
<p>Keywords</p> <p>Responsive web design, HTML5, CSS3, layouts</p>	

Sisällys

1. Johdanto	6	5.Responsiivisuuden osat	28
2. Verkkosuunnittelun historiaa	9	5.1. Layout eli taittopohja: Fluid Grid, Flex/Elastic, Static	29
2.1. HTML internetin yhteinen kieli (Hypertext Markup Language)	10	5.2. Fluid images: Joustavat kuvat	31
2.2. CSS tyylinhallinta (Cascading Style Sheets)	11	5.3. HTML5	32
2.3. 960 Grid	12	5.3.1. Taustat ja käytettävyys	32
3.Responsiivinen Web	13	5.3.2. Uusia elementtejä	36
3.1. Mitä on responsiivinen verkkosuunnittelu	14	5.3.3. Dokumenttityypin määrittely (doctype)	38
3.2. Millainen on responsiivinen sivu, entä staattinen?	15	5.3.4. Viewport	39
3.3. Onko mobiililla väliä?	18	5.4 Media Queries (CSS3)	41
3.4. Millainen on "hyvä" responsiivinen sivu?	20	5.4.1 Perusidea	41
4. Käytännön näkökulmia responsiivisuuteen	23	5.4.1 Murtumispisteet (breakpoints)	43
4.1. Suunnittelijan näkökulma "uusia työkaluja... vai leluja?"	24	5.5 . Edistyneemmät toiminnot	45
4.2. Käyttäjän näkökulma "mitä hyötyä?"	26	5.5.1. Javascript	45
4.3. Asiakkaan näkökulma " miksi maksaa"	27	5.5.2. jQuery	46

6. Demosivu	47
6.1. Mitä ja kenelle	48
6.2. Pohjan luonti	50
6.3 Navigaatio	52
6.4. Kommenttipalsta	53
6.5. Galleria	54
6.6. Optimointi	58
6.7. Testaus ja lopputulos	59
7. Päätäntä	60
Lähteet	62
Kuvalähteet	65

1. Johdanto

Vielä runsas vuosikymmen takaperin internetin käyttäjäkunta istui hartaasti oman pöytänsä ääressä, tietokone kun ei niinkään ollut laitteista helpoin kuljettaa mukana. Tekniikan kehittyessä kannettavien koneiden suosio kasvoi nopeasti, vaikka ne olivatkin teholtaan isoveljiään heikompia. Kännyköiden kehityskäyrän tapaan kannettavien tietokoneiden koko pieneni merkittävästi sitä mukaa, kuin tekniikka sen salli. Ymmärrettävähän on kuinka paljon helpompaa on kantaa mukana pienempää laitetta, vaikkei näyttöpäätteen koko vastaisikaan pöytäkoneen päätettä.

Pian olimme kuitenkin jo pisteessä, jossa kannettavien ja pöytäkoneiden rinnalle markkinoille saapuivat älypuhelimet ja tabletit, jotka mahdollistivat internetin hyödyntämisen käytännössä missä tahansa. Kullakin laitteella oli omat vahvuutensa ja siten omat käyttäjänsä, jotka usein hyödynsivät useampaa kuin yhtä väylää. Tämä kuitenkin tarkoitti sitä, **ettei verkkosivujen ylläpitäjä enää tiedä, millaisella laitteella kävijä hänen sivujaan katselee.**

Kävijöiden aiheet myös vaihtelevat, mobiilikäyttäjät hakevat usein verkkosivulta eri asioita kuin pöytäkonekäyttäjät ja tätä myötä arvostavat toisenlaisia sivuja.

Perinteiset verkkosivut eivät palvele tällaista käyttöä erityisen hyvin. Ongelman voisi tietenkin ratkaista luomalla useita eri käyttäjille suunnattuja verkkosivuja, mutta käytännöllisyyden nimissä on huomattavasti tehokkaampaa luoda **yksi verkkosivu, joka mukautuu käyttäjän valitsemaan formaattiin.** Näin helpotetaan sivun ylläpitoa ja päivittämistä. **Tällaista sivua kutsutaan responsiiviseksi verkkosivuksi.**

Seuraavassa käymme lyhyesti läpi verkkosuunnittelun historiaa, jonka jälkeen tutustumme toimintopakettiin, jolla responsiivisuus toteutetaan. Rakennamme lopuksi tyhjältä pohjalta uuden responsiivisen sivun, jossa pyrimme käyttämään responsiivisia ratkaisuja. Emme tutustu sivun rakentamiseen keskittymällä koodiin, vaan valittuihin käytänteisiin ja toiminnallisuuteen.

Alla lyhyt sanasto auttamaan opinnäytteen tulkisemisessa. Termistöä avataan vielä tarkemmin omassa kontekstissaan.

Selain: Verkkosivujen lukuohjelma, yleisimmät ovat Mozilla Firefox, Google Chrome, Internet Explorer (IE), Opera ja Apple Safari. Selain on tärkein verkkosivua tulkitseva väline, sillä se piirtää verkkosivun näkymän lukijalle, verkkosivua suunnittelussa on hyvä tuntea eri selainten heikkouksia ja vahvuuksia.

Lukija/Käyttäjä: Tässä opinnäytetyössä käyttäjällä ja lukijalla viitataan aina verkkosivulla vierailevaan kävijään tai sivuston asiakkaaseen.

HTML: (Hypertext Markup Language) Verkkosivun rakenteen ja sisällön määrittävä tiedosto, sivun selkäranka ja sydän. HTML tiedosto sisältää verkkosivun varsinaisen, sisällön, mutta ei yleensä muotoilua, jolla sivu saataisiin näyttämään tietynlaiselta lukijan selaimessa.

CSS: (Cascading Style Sheet) Verkkosivun ulkonäön määrittävä tiedosto. Määrää millaisena selain näyttää kulloinkin luetavan internet sivun. Tämä tiedosto linkitetään yleensä HTML tiedostoon, jolloin selain tietää mistä hakea ulkoasun määräävä komentosarja sivun ulkoasun piirtämistä varten.

Linkki: Painike tai vastaava osa verkkosivua, jota painamalla käyttäjä pääsee uudelle sivulle tai avaa uudenlaista sisältöä nähtäväksi.

Navigaatio: Verkkosivuista puhuessa navigaatio tarkoittaa kokoelmaa linkkejä, joiden avulla käyttäjä voi liikkua ja suunnistaa sivuston eri lehdillä.

Elementti/Objekti: Puhuessamme elementeista tarkoitamme verkkosivun osasia, kuten navigaatiopalkkia, tekstikenttiä tai painikkeita. Elementti voi olla myös kokoelma pienempiä sivun palasia.

Skaalautuminen: Mukatuminen käytössä olevaan tilaan, koon dynaaminen muuttuninen. Tällä termillä viittaamme opinnäyteessä elementteihin, jotka muuntautuvat sopimaan käytössä olevaan tilaan.

Taittopohja/Layout: Taittopohja on nimensä mukaisesti taiton apuväline. Kyse on tavasta, jolla sivu tai muu sisältö on jaettu järjestelmällisesti osioiksi, jotta sisällön ja ulkoasun hallitseminen olisi helpompaa.

Media query: CSS3:n tukema toiminto, jonka avulla voidaan luoda verkkosivun ulkomuotoon sääntöjä liittyviä sääntöjä, jotka koskevat vain tiettyjä tilanteita/medioita. Tätä toimintoa hyödyntämällä voimme rakentaa verkkosivuja joiden ulkoasu muuttuu käytettävän tilan mukaan, vaikka käskemällä sivua hävittämään elementtejä kun tila on vähissä.

Fluid grid: Taittopohja, joka perustuu siihen, että sivun kaikki mitat ovat suhteellisia toisiinsa nähden, siten, että sivun rakenne skaalautuu. Tällaisessa taittopohjassa sivun eri osat ovat tietyn kokoisia suhteessa toisiinsa, mutta niiden yhteismitat muuttuvat käytettävän tilan mukaan.

Käytettävä tila: Tätä mainintää käytetään opinnäytteesssä viittaamaan eri näyttöjen pikselittojen eroihin. Esimerkiksi **puhelimien näytössä voi olla vähemmän pikseleitä kuin tietokoneen ruudulla = vähemmän käytettävää tilaa.**

2. Verkkosuunnittelun historiaa

2.1. HTML internetin yhteinen kieli (Hypertext Markup Language)

Kirjoittaessa verkko-osoitteen selaimen hakukenttään serveri lähettää selaimelle osoitetta vastaavan sivun. Tämän sivun keskeisin sisältö on tiedosto, joka on kirjoitettu HTML-tietokonekielellä sisältäen sivun rakenteen ja luultavasti pääosan varsinaisesta sisällöstä. HTML on ollut käytössä yhdeksänkymmentäluvun alusta lähtien. Kielen isä on Sir Tim Berners-Lee, joka julkaisi 1991 kirjoituksen (HTML Tags), jossa hän ehdotti kieleen alle tusinaa valikoitua elementtiä, joita voisi käyttää internetsivujen kirjoittamiseen. Berners-Lee tunnetaan paremmin koko internetin isänä.

Vuosien saatossa HTML:sta on kehitetty lukuisia revisioita vaihtelevalla menestyksellä. Alkuvaiheessa HTML:n kehityksestä ja virallisista määrittelyistä vastasi IETF (the Internet Engineering Task Force), jonka paikan myöhemmin otti W3C (the World Wide Web Consortium).

W3C-organisaatio toimii edelleen 2013 internetin yhteisten pelisääntöjen, suositusten ja yhteisten tekniikoiden kehittäjänä, sen merkitys on vain kasvanut aikaa myöten, vaikkakin organi-

saation asemasta ”vastaajana” kiistelläänkin välillä. Esimerkiksi HTML5, HTML-kielen uusin iteraatio pohjautuu ideaan, jonka toteutusta W3C alkuperiään vastusti haluten mieluummin kehittää XHTML spesifikaatiota. XHTML:n polku ei kuitenkaan ollut aivan ruusuinen. (Keith 2010; Nykänen 2003)

Mainittakoon vielä, että HTML5 pohjautuu Applen, Mozillan ja Operan muodostaman WHATWG työryhmän ehdotuksiin. Hy-lätessään viimein XHTML:n jatkokehityksen W3C otti nimellisesti johdon tulevan HTML5:n kehityksessä.

Käytännössä kuitenkin WHATWG kehittää ehdotuksia ja W3C hyväksyy ne. (Keith 2010; Nykänen 2003)

2.2. CSS tyylinhallinta (Cascading Style Sheets)

CSS on HTML:n ulkoinen keino hallita sivun ulkoasun esitystä. Käytännössä se on tekstitiedosto, joka sisältää listan sääntöjä, joita selain soveltaa HTML tiedostoon.

Internetin kasvaessa ja sivujen sisällön rikastuessa syntyi tarve hallita sisällön esitystapaa tehokkaammin tekemättä HTML:n hallinnasta liian monimutkaista. Tämä oli otettu huomioon jo alkuperäisen HTML:n suunnitelmassa, jossa oli pyritty siihen, että sivujen ulkoasua tulnaisiin hallitsemaan ulkoisilla tyylitiedostoilla. Varsinaista syntaksia näitä ulkoisia tiedostoja varten ei kuitenkaan oltu kehitetty, sillä sivun ulko-

asun tulkitseminen jäi selainten tehtäväksi. Kestikin vuosia ennen kuin selainkehittäjät innostuivat lisäämään tyylinhallintaan liittyviä keinoja selaimiinsa tai ylipäätään tukemaan niiden kehitystä. Ensimmäinen luonnos CSS:ta julkaistiinkin vasta 1994 norjalaisen Håkon Wium Lie:n toimesta. Luonnos ei kuitenkaan ollut ainoa ja W3C:n sisällä käyntiinkin pitkä kädenvääntö ennen kuin CCS (versio 1.0) valittiin suositukseksi joulukuussa 1996. (Bos & Wium Lie 1999)

2.3. 960 Grid

960 Grid on leveyteen perustuva taittopohja, jonka tarkoitus on toimia **aputyökaluna verkkosivun taiton suunnittelulle**. Nimi tulee 960 pikselin leveydestä, joka toimi pitkään hyvänä mittana sivuille, sillä se sopi pääosalle käytettäviä näyttöpäätteitä. Lisäksi 960 pikseliä oli käytännöllinen luku, sillä se voitiin jakaa 12 tai 16 kolummiin taiton helpottamiksi (vastaavasti 60 ja 40 pikseliä). Mitta ei koskaan ollut virallinen standardi, mutta se nauttii laajaa suosiota ja sitä pidetään edelleen hyvänä käytänteenä, jos sivusta ei tehdä responsiivista tai muuten skaalautuvaa.

Responsiivisessa suunnittelussa sivun leveyden vakioimi-

sesta ollaan pitkälti luovuttu, mutta jako 12/16 kolummiin on edelleen hyödyllinen työkalu. Responsiivisessa suunnittelussa jako vain tehdään prosentuaalisesti pikseleiden sijaan. Responsiivisuus ei kuitenkaan tarkoita, että 960 gridista on pakko luopua. Taittopohjaa voi käyttää myös sellaisenaan, mikäli haluaa täyden hallinnan käyttäjän näkymästä. Tällöin sivun responsiivisuus toteutetaan täysin media queryja hyödyntäen.

3.Responsiivinen Web

3.1. Mitä on responsiivinen verkkosuunnittelu

Nykyään verkkosivua kehitettäessä ei enää tiedetä varmaksi, millaista mediaväylää eli lukulaitetta käyttäjä tulee käyttämään. Responsiivinen suunnittelu on yksi ratkaisu tähän verkkosuunnittelun ongelmaan.

Suunnittelufilosofian näkökulmasta responsiivinen suunnittelu merkitsee loppukäyttäjän erityistä huomioon ottamista. Sivut suunnitellaan ja kehitetään ottamaan huomioon lukijan käyttämä ympäristö ja mukautumaan tähän. Lisäksi lukijan tarpeet ja motiivit otetaan entistä enemmän esille.

Teknisestä näkökulmasta responsiivisuus on tekniikka, jos-

sa serveri lähettää aina saman HTML-koodin kaikille laitteille ja CSS muotoilee sen sopivaksi media queryja käyttäen. Tekniikan tärkein etuus on, ettei serverin tarvitse lähettää verkkosivua kuin kerran, eikä kävijää tarvitse ohjalle laitepohjaisesti eri sivuille. (Google Developers 2013)

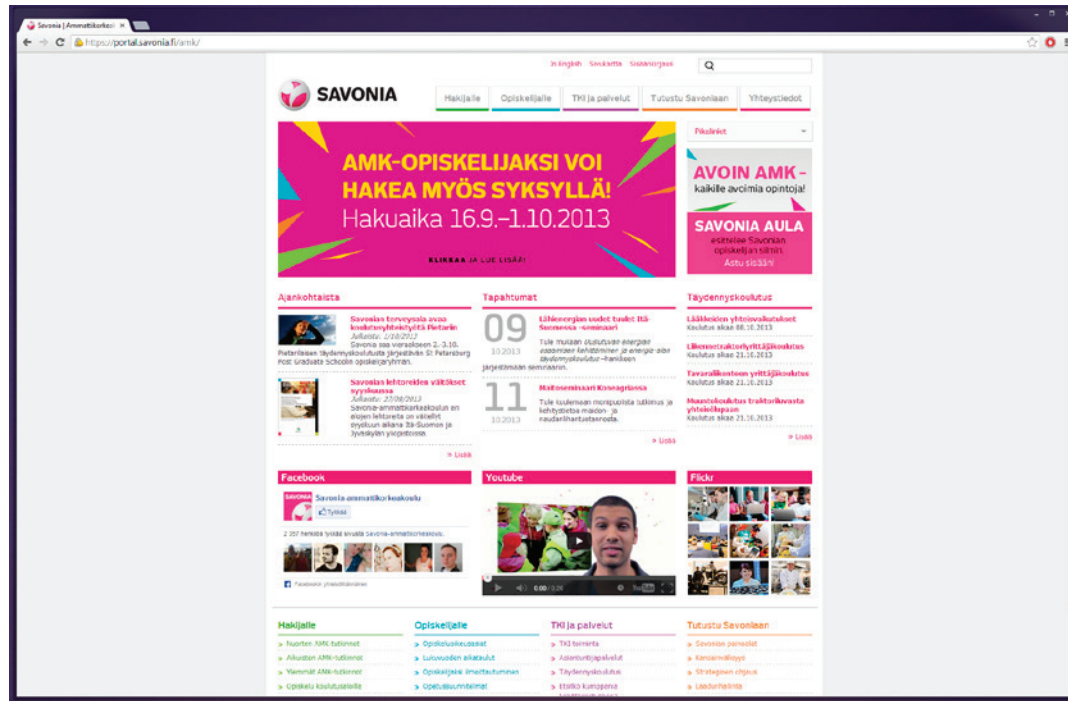
3.2. Millainen on responsiivinen sivu, entä staattinen?

Staattinen eli pysyvärakenteinen verkkosivu on tietyn kokoinen. Sen sisällöllä on selkeä mitta, eikä sitä kiinnosta, millaista lukupäätettä kävijä käyttää. Helppo tapa testata sivun responsiivisuus on ottaa kiinni selaimen laidasta, ahdistaa sisältöä ja katsoa miten sivu reagoi. Staattinen sivu ei osaa suhtautua mitenkään sivun kutistumiseen, joten se jää vain sivun reunan alle tai sallii sisällön painua muodottomaksi massaksi. Jotkin sivut sisältävät osittain skaalautuvia elementtejä ja saattavat näyttää päällisin puolin responsiivisilta kunnes niitä testataan merkittävästi rajoittuneemmassa ikkunassa. Mobiililaitteella staattisen sivun käyttö vaatii usein zoom-toiminnon ahkeraa käyttöä ja pakottaa käyttäjän etsimään hakemaansa sivulta.

Responsiivinen sivu on suunniteltu olemaan tietoinen ympäristöstään, mukautumaan siihen ja reagoimaan tämän muutoksiin. Kun responsiivisen verkkosivun sisältöä ahdistetaan, sen sisältö joustaa käytettävän tilan muuttuessa ja elementit saattavat tippua allekkain, muuttaa muotoaan tai vain kadota kokonaan. Tietyissä pisteissä sivun näkymä saattaa muuttua

merkittävästi, navigaatiopalkki muuttua erilaiseksi, painikkeet muuttua ikoneiksi. Nämä ovat niin sanottuja murtumispisteitä, joissa sivu olettaa selainikkunan koon perusteella käyttäjän olevan sivulla jonkin mobiililaitteen kuten älypuhelimien tai tabletin kautta ja muuttaa ulkoasuun vastaamaan laitteen ominaisuuksia. Ideana tässä on, että sisältö pysyy luettavissa ja sivun esteetön käyttö on helppoa lukulaitteesta huolimatta.

(Responsive Web Design. Marcotte, E. 2010)



kuva 1.

Sivu toimii näkyvä toimivan täydessä näkymässä ongelmitta.



kuva 2.

Sivua ahdistassa sen sisältö peittyi kuitenkin selaimen reunan alle. Pienempää ikkunaa tai älypuhelimien selainta käytettäessä lukija joutuu vierittämään sivua suuntaan jos toiseen.



kuva 3.

Tämän responsiivisen sivuesimerkin perusnäkyvä on suhteellisen normaali.

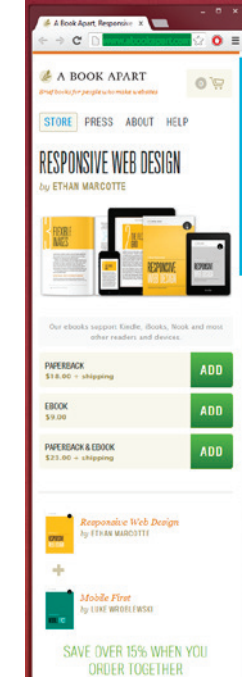
(Mainittakoon, että selainikkunaa levennettyessä sivu keskittyy ja täyttää reunat tyhjällä tilalla.)



kuva 4.

Kaventaessa selainikkunaa sivu poistaa ensin reunojen tyhjän tilan ja alkaa sitten kaventamaan elementtejä. Yläkulman painikkeet seuraavat selaimen reunaa. Tilan vähetessä painikkeet kasvavat koossa ja hyppäävät allekkain.

Tämä vastaa karkeasti näkymää tablet-laitteen ruudulla.



kuva 5.

Lisää kavennettaessa. Kuvamaailmakin alkaa kaventua ja elementit putoilevat allekkain. Napit säilyttävät kuitenkin kokonsa, jotta ne olisivat edelleen painettavissa, samoin kuin tekstit, jotka säilyttävät luettavan koon.

Tämä vastaa karkeasti älypuhelimien näkymää.

3.3. Onko mobiililla väliä?

Mobiililaitteet nousevat usein esille responsiivisesta suunnittelusta keskusteltaessa. Välillä niinkin usein, että luulisi, ettei responsiivisuudessa muusta olekaan kyse. Ajatus ei ole täysin perusteeton, vaikka responsiivisuudella on muitakin hyötyjä. Mutta ovatko mobiililaitteet kaiken kohinan arvoisia?

Ei ole juuri tarpeen tehdä erillistä tutkimusta huomatakseni, että älypuhelimet ja tabletit populaari-ilmiö nykymaailmassa. Ne ovat joka paikassa ja vanhat pöytäkoneet ovat pikku hiljaa jäämässä tehokäyttäjien yksityisomaisuudeksi. Kannettavat myyvät edelleen, mutta tabletit suorittavat jo pitkälti samat tehtävät kuin mitä keskiverto kannettavan omistaja koneeltaan tarvitsee.

Älypuhelinien ja tablettien suosion myötä internetin käyttö mobiililaitteilla lisääntyy nopeasti, sillä väylä muodostuu koko ajan tutummaksi käyttäjäkunnalle ja kokemus tekniikasta lisääntyy. Mobiililaitteita hyödyntävien kävijöiden määrän kasvaessa kasvaa myös tarve tehokkaille ja käytännöllisille mobiiliratkaisuille. Täysin vakiintuneita malleja ei vielä ole, mutta tarve on jo täällä, eivätkä asiakkaat ole ymmärtäväisiä älypuhelimella huonosti toimivia

sivuja kohtaan. Responsiivisuutta voidaan pitää yhtenä tehommimmista ratkaisuista, sillä sen avulla mobiilikäyttäjille ei ole tarpeen rakentaa erillisiä sivuja.

Tulevaisuutta ajatellen on myös huomattava, että tottuessaan mobiilisivujen käyttöön, selaajat alkavat odottaa vastaavaa toiminnallisuutta kaikilta sivuilta. Mobiilisti käytettävien sivujen puute muodostuu ongelmaksi. **Vielä toistaiseksi mobiilisivu on kilpailuetu, pian sen puute on heikkous.**

Frain nostaa esille mielenkiintoisia tilastoja, joissa nähdään mobiilikäytön kasvavan nopeasti ja vanhojen IE versioiden käytön vähenevän jatkuvasti. Hän mainitsee ettemme voi jäädä jumiin toteuttamaan suunnittelua vanhaa tekniikkaa varten vain koska pieni osa käyttäjäkuntaa ei ole halukas tai osaa käyttöönottaa uudempaa tekniikkaa. Hän näkee kasvavan mobiilikäyttäjäkunnan tärkeämpänä yleisönä kuin marginaalisen IE käyttäjäkunnan. (Käytämme IE käyttäjäkuntaa negatiivisena esimerkkinä vain koska suurin osa yhteensopivuusongelmista kohdistuu vanhempiin IE selaimiin, joita on vielä runsaasti käytössä maailmalla. Uudemmat 9+ versiot IE:sta toimivat responsiivisuuden näkökulmasta huomattavasti paremmin.) (Frain 2012, 8.)

Gemertilla on mielenkiintoinen kanta verkon muutokseen. Hän nostaa esille monia aikaisempia olettamuksia ja miten ne eivät enää ole täysin sovellettavissa käytäntöön. Esimerksi, emme enää voi olettaa lukijan käyttävän hiirtä tai omaavan langallisen laajakaistan. Lisäksi laitekehitys on tavallaan pysähtynyt, sillä vaikka pöytäkoneet olisivat millaisia supertietokeita, on mobiilitekniikka muutaman askeleen perässä. Hän myös korostaa, ettei internet ole oikeastaan muuttunut, vaan ainoastaan sen vaatimukset. **Meidän on alettava toteuttaa verkkosuunnittelua toiselta pohjalta kuin aiemmin.** Internet on alusta alkaen ollut outo rajaton muodoton ympäristö, nyt vain toteutamme sitä enemmän kuin ennen. (Gemert 2013)

laskelmat ovat suuntaa antavia ja alati muuttessa infografiikka [Marketingland](#) sivustolta [02.08.2013]



kuva 6.

3.4. Millainen on "hyvä" responsiivinen sivu?

Tärkein kriteeri verkkosivuille, niin responsiivisille kuin staattisille on se, että niiden on palveltava tarkoitusta. Oli tämä sitten mikä tahansa, olla tiedonlähde, myydä tuotteita, toimia yhteisönä tai työnäytteenä. Erona responsiivisessa verkkosivussa staattiseen nähden on kuitenkin se, että sivun kävijöiden odotetaan jakaantuvan laitekohtaisesti, jolloin kävijät eivät välttämättä etsi sivulta samoja asioita. **Mobiilikäyttäjillä onkin usein erilaiset syyt verkkosivulla käymiselle kuin pöytäkonekäyttäjillä**, sillä vierailukynnys, käyttöominaisuudet ja käytettävissä oleva aika ovat erilaiset.

Usein mainitaan, että mobiilikäyttäjän yleisimmin verkkosivulta hakemat asiat ovat yhteystietojen haku tai olemassaolevan tiedon vahvistaminen. Tämän puolesta puhuu myös Googlen tekemä käyttäjätutkimus, joka tosin keskittyy kaupanalan verkkosivujen kävijäkuntaan, joiden haetuimmiksi asioiksi osoittautuivat asiakaspalvelun yhteystiedot ja aukioloajat, sekä sähköpostiosoitteet (kuva 6.). (Sterling 2012)

Hyvässä suunnittelussa on siis otettava huomioon laitekohtaiset erot vierailijoiden motiiveissa. Tämä voi tarkoittaa esimerkiksi tarpeettoman tiedot piilottamista, kun sivua luetaan mobiililaitteilla tai tiettyjen elementtien korostamista. **Nyrkkisääntönä mitä vähemmän tilaa on käytettävissä, sitä vähemmän epäoleelliselle sisällölle kannattaa antaa tilaa.** Tämä pitää paikkansa myös sivuilla liikkumisen suhteen. Yleensä voidaan sanoa, että mitä vähäisemmällä määrällä klikkauksia lukija pääsee haluamansa sisällön luokse sitä parempi. Tämä on kaksinverroin totta mobiilikäyttäjien suhteen. Etusivua ei silti kannata täyttää linkeillä, vaan järkevämpää on pohtia mitä sisältöjä eniten tullaan käyttämään ja korostaa näitä.

Kaikkien laitteiden ei tarvitse olla tasa-arvoisia, jos sivulla kosiskellaan juuri jotain tiettyä asiakaskuntaa. Voi olla, että sivu keskittyy juuri mobiiliasiakkaisiin tai selviää, että 90% asiakkaista käy sivulla juuri pöytäkoneella. Tällaisessa tilanteessa ei ole haittaa keskittyä juuri tämän laitteen käyttökokemuksen optimoimiseen, koska näin palvellaan keskeisen asiakaskunnan

tarvetta. On silti aina otettava huomioon, että asiakaskunnan jakauma voi johtua huonosti toteutetusta responsiivisuudesta, jolloin sivut ajavat lukijaa suosimaan tiettyä laitetta.

Mahdollisia sudenkuoppia responsiivisessa toteutuksessa!

Sivu on teknisesti ottaen responsiivinen, muttei hyödynnä tätä mitenkään, sivun rakenne on sama kaikissa skaalauksen kokoluokissa. (tulos on kuin sivu ei olisi responsiivinen lainkaan, sillä koko sisältö kutistuu laitteen mukaan)

Sivun sisältö on responsiivista ja rakennettu älypuhelimia ajatellen, sisällön määrä ja koko on optimoitu juuri kapealle näytölle (Isommat näyttökoot ovat unohtuneet, suurella näytöllä katsottaessa sisältö venyy pahasti ja lopulta hajoaa, kuvien laatu laskee sietämättömäksi ja teksti asettuu kaikki samalle riville)

Sisältö on responsiivinen ja kaikki tarpeettomina pidetyt materiaalit poistetaan pienemmässä näkymässä. Sivun rakennus on niin, että älypuhelimella luettaessa, ainoastaan välttämättömät tiedot näkyvät. (Prioriteettinäkökulma on hyvä, mutta lukijalle kannattaa jättää vaihtoehto, jotta hän halutessaan pääsee käsiksi koko sivun sisältöön laitteesta huolimatta.)

Sivu on responsiivinen ja elementit romahtavat allekkain, mutteivat ole missään loogisessa järjestyksessä (On riski, että käyttäjä selatessaan sivua alaspäin kohtaa vähemmän oleellista sisältöä ja luonnollisesti olettaa ettei sen alapuolelta löydy enää mitään tärkeää)

Sivu on responsiivinen ja sisältää suuren määrän navigaatiolinkkejä menujen muodossa. Kapeassa näkymässä navigaatiot asettuvat kaikki allekkain sivun yläreunaan. (Navigaatioita on liikaa, vaikka ne muutetaan näkymään sopiviksi joutuu lukija selämaan sivua merkittävästi alaspäin ennen kuin pääsee varsinaisen sisällön kimppuun, tässä tilanteessa hyvä ratkaisu olisi piilottaa osa linkeistä/navigaatiosta oman avattavan menun sisään. Näin kaikki ominaisuudet säilytetään, vapauttaen samalla tilaa muulle sisällölle)

4. Käytännön näkökulmia responsiivisuuteen

4.1. Suunnittelijan näkökulma ”uusia työkaluja... vai leluja?”

Responsiivisuus on suunnittelijalle kaksiteräinen miekka. Se on hyödyllinen ja tehokas työkalu, jolla joitain askelia verkkosivun rakennuksesta voidaan tehostaa. Toisaalta responsiivisen sivun rakentaminen on työläämpää kuin staattisen ja siinä on otettava huomioon monia eri tilanteita, joita lukija voi sivulla kohdata. Vaikka suunnittelija ei olisikaan sivun varsinainen koodaaja on hänen silti ymmärrettävä tekniikan perusajatus ja osattava hahmottaa, miten sivu reagoi eri tilanteissa.

Vaadittavasta sisällöstä riippuen erot mobiiliin ja täyden näkymän välillä voivat olla suuretkin ja vaatia useita muutoksia

sivun tyylitiedostoon. CSS3:n uudet media queryt ovat tehokkaita, mutta eivät ratkaisu kaikkien mahdollisiin tilanteisiin, joten responsiivinen näkökulma tulisi ottaa huomioon jo suunnitteluvaiheessa ja rakennetta laatiessa. Parhaassa tilanteessa elementit on järjestetty niin, että ne on helppo skaalata ja ”romahduttaa” mobiilia näkymää varten. Kaikkia täyden näkymän ominaisuuksia ei välttämättä kannata pakottaa mobiilinäkymään, eikä tilaa pystylle sivunavigaatiolle varmaankaan ole, ellei sitä piilota sivunäkymään.

Sivun skaalautuvaksi muuttaminen ei tarkoita, että kaikki muutetaan skaalautuvaksi. On mietittävä mikä säilyttää kokonsa ja mikä mukautuu. Jos jokainen elementti skaalautuu sivun kokonaishallinnasta tulee hankalaa.

Responsiivisuuden ei kuitenkaan ole pakko muuttaa sivun rakentamisen prosessia. Mikään ei estä ensin rakentamasta tietynkokoista sivua, joka sitten muunnetaan responsiivikseksi. Sivun voi myös rakentaa toisin päin eli ensin suunnitella mobiilin näkymän ja sen pohjalta siirtyä kohti suurempia näkymiä.

Jos alusta asti on tiedossa, että sivusta tehdään responsiivinen voi joustavaa layouttia hyödyntää jo suunnitellessa. Skaalautuvuus ei koske ainoastaan pienempiä medioita, vaan voi myös ottaa huomioon suuremmat näkymät. Ovathan pöytäkooneiden näytöt tänä päivänä jo keskimäärin huomattavasti suurempia kuin vuosikymmen takaperin. Sivun ei tarvitse mukailla esimerkiksi normaaleja sivun leveyteen liittyviä suosituksia, sillä sivun sisältö skaalautuu joka tapauksessa lukijan näyttöpäätteeseen. Esimerkiksi galleriat ovat herkullinen paikka toteuttaa rajoittamatonta sivunleveyttä.

Jos toteutuspuoli jätetään sivummalle ja ajatellaan responsiivisuutta suunnittelijan ammatillisesta näkökulmasta puhutaan uudesta työkalusta, uudesta formaatista, joka olisi hyvä

hallita. Kyse on varsin keskeisestä tekniikasta, jonka osaamisen tärkeys varmasti kasvaa jatkossa. Responsiivisuuden osaamisen tarve on tuskin katoamassa aivan pian.



”mitä tapahtuu horisontaalisen tilan vähetessä?”

”mitä tapahtuu kun tilaa tulee lisää?”

”mitä käyttäjä haluaa sivulta?”

”tila on rajallista, mikä ansaitsee paikan?”

”miten säilytän sivun ilmeen uudessa näkymässä?”

4.2. Käyttäjän näkökulma ”mitä hyötyä?”

Loppukäyttäjä eli verkkosivuilla asioija kaipaa responsiivista suunnittelua lähinnä silloin, kun se puuttuu. Käyttäjä harmittelee helposti vaikeakäyttöisiä sivuja, mutta pysyy hiljaa kun sivut toimivat. Tämä ei ole ongelma, sillä työkaluna responsiivisuuden päätehtävä on usein vain saada sivut toimimaan normaalisti ja hävittää käytännön puolen esteet sivun mobiilinäkymästä. On odottavissa, että käyttäjät tottuvat aikaa myöten yhä enemmän verkon mobiilikäyttöön ja tätä myötä mobiilisivuihin. Mitä normaalimmaksi sivujen käyttö tässä muodossa tulee sitä vähemmän siihen kiinnitetään huomiota.

Päinvastainenkin voi olla totta, kaikki käyttäjät eivät mahdollisuudesta huolimatta halua tai hoksaa kokeilla sivustojen toimivuutta mobiilissa. Suunnittelijan ja verkkosivun ylläpitäjän kannalta voi olla hyvä idea kertoa sivulla, että sivu toimii loistavasti, myös älypuhelimella tai tabletilla katsottuna, jotta vaihtoehto olisi edes tiedossa.

4.3. Asiakkaan näkökulma ” miksi maksaa”

Responsiivinen tekniikka on jotain, jota sivujen loppukäyttäjät osaavat pian vaatia, eikä käyttäjän kärsivällisyys riitä lopputömiin, etenään kun kilpailijoiden sivut toimivat älypuhelimella moitteettomasti. Mitä suositummat sivut sitä enemmän kävijöitä ja tätä myötä mobiilikäyttäjiä. Tarve tekniikalle on suoraan suhteessa siihen miten tukeunutta asiakkaan liiketoiminta on verkkosivuihin. Tästä huolimatta monet asiakkaat, eivät avustamatta näe responsiivisen tekniikan etuja tarpeeksi suurina, että panostaisivat siihen.

Asiakas eli tässä tilanteessa verkkosivujen omistaja voi epäillä mobiilin liikenteen merkittävyyttä tai olla epäluuloinen uuden tekniikan suhteen. Onhan mobiileja sivuja rakennettum jo jonkin aikaa erillisinä sivuina tai esimerkiksi Wordpress plugineja käyttäen. Jotkut ovat täysin tyytyväisiä siihen miltä heidän sivunsa näyttävät mobiililaitteilla, tällaisissa tapauksissa on tarpeen muistuttaa asiakkaalle, että toimivuus on ulkonäköä tärkeämpää.

Ensimmäinen asia, jonka moni asiakas nostaa pöydälle on kuitenkin responsiivisen suunnittelun hinta. Responsiivisuus vaatii enemmän aikaa ja tätä myötä työtunteja, mikä suoraan nostaa hintaa. **Valitettavasti kyse ei ole kalliimmaasta verkkosivusta, vaan normaalista tämän päivän verkkosivusta.** Verkkosivu on pian sama asia kuin responsiivinen verkkosivu, eikä asialle erityisesti voi mitään. (Grace 2013)

5.Responsiivisuuden osat



5.1. Layout eli taittopohja: Fluid Grid, Flex/Elastic, Static

Fluid Grid tarkoittaa joustavaa taittopohjaa, kyseessä on idea dynaamisesta pohjasta, jossa taitto jakaantuu vakiomittaisten osioiden sijaan suhteellisen mittaisiin lohkoihin. Idea on pitkälti sama, kuin 960 grid taittopohjassa, mutta vain joustavilla mitoilla. Hienosta nimestä huolimatta Fluid grid ei ole ympäristö, kuten HTML ja CSS, vaan yksinkertaisesti layoutin suunnittelutapa. Termiä käytetään jonkin verran ristiin, sillä se yhdistetään helposti Fluid grid idealla toteutettuihin valmiisiin taittopohjiin.

Jotkin verkkosivueditorit tarjoavat suoraan Fluid Grid nimeä käyttäviä template-pohjia, joilla tässä yhteydessä tarkoite-

taan valmiiksi skaalautuvaksi suunniteltua pohjaa. Esimerkiksi Dreamweaver-ohjelmassa voi luoda Fluid Grid sivupohjia, joita muokatessa on käytettävissä erillisiä lisäominaisuuksia, joilla hallita layoutia, kuten kyky säätää montako kolumnia on käytössä minkäkin kokoisella lukulaitteella. Käytännössä pohjaa luodessa valitaan, montako kolumnia otetaan käyttöön mihinkin näkymään,



Fluid gridin rinnalla on myös määrätyn koon (Fixed) ja venyviä (Flex tai Elastic) taittopohjia, jotka voivat samalla tavalla olla responsiivisia. Pohjien keskeisin ero on se miten ne mukautuvat sisältöön.

Määrätyn koon taittopohja on tietyn kokoinen, se voi käyttää media queryjä, muuttaakseen ulkoasuun tietyissä hajoamispiisteissä, mutta muuten se ei skaalautu sisältönsä tai ympäristönsä mukaan.

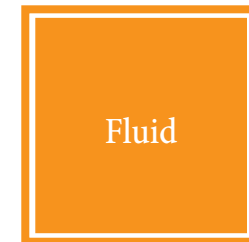
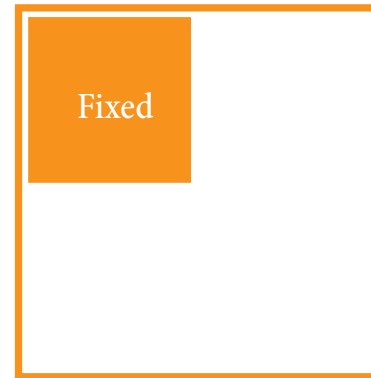
Venyvä taittopohja toimii samalla periaattella kuin joustava taittopohja. Se skaalautuu ja muuntuu tilan mukaan, mutta vain tiettyyn pisteeseen saakka. Saavutettuaan maksimi tai minimi mittansa pohja lopettaa venymisen. Tätä käyttämällä voi lopettaa esimerkiksi sisällön venymisen liian suureksi kun käytössä on suuri näyttölaite.

Eri taittotavoista eli layouteista käytettävät nimitykset ja niiden määritelmät vaihtelevat paljon ja voivat kuullostaa ristiriitaisilta. Välillä termejä käytetään suoranaisesti ristiin ja tässäkin artikkelissa mainitut kuvaukset eivät usein vastaa, sitä mistä toinen osapuoli puhuu. Taittotavoille ei ole olemassa virallisia standardeja tai yleisesti hyväksyttyä määritelmää, joten ongelmaan ei ole suoraa ratkaisua. Pääasia on kuitenkin eri taittotapojen mahdollisuudet tiedostaa, vaikka termit olisivatkin sekavia (Knight 2009)

”We used to design inside fixed grids with the 960px system. Then the layout became adaptive by using different pixel sizes at different screen sizes. It was all about pixel based layout design. Now the time has come to create percentage based designs which are known as fluid designs.” (Nimesh 2013)

5.2. Fluid images: Joustavat kuvat

Lyhyesti sanottuna Fluid Image on skaalautuva kuva. Skaalautuvan taiton tapaan se reagoi käytössä olevaan tilaan ja mukautuu siihen. Teknisesti ottaen kyse on kaikessa yksinkertaisuudessaan kuvasta, jolla prosentuaalinen mitta kiinteä mitan sijaan. Kuvan liiallinen skaalaus suuntaan tai toiseen pilaa kuitenkin sen laadun, joten liiallisen skaalautumisen kanssa kannattaa olla varovainen. Esimerkiksi gallerioissa, joissa kuvanlaatu on tärkeää voi olla parempi ratkaisu käyttää vakiokokoisia kuvia ja toteuttaa tilan hyödyntäminen mosaiikkina skaalautumisen sijaan.



HTML

5.3. HTML5

5.3.1. Taustat ja käytettävyys

HTML5 on uusin HTML-kielen eli internetin yhteisesti käyttämän kielen iteraatio. Standardi on vielä tuore ja edelleen kehityksen alla, vaikka onkin ollut kehityksessä jo vuodesta 2004. HTML5 nimi tuli tosin käyttöön vasta 2008, kun joukko pienempiä uudistuksia yhdistettiin. Joidenkin lähteiden mukaan standardin suositukseksi muuttuminen voi vielä viedä vuosia, mutta ainakin toistaiseksi W3C:n suunnitelmissa on virallistaa vakaa suositusmalli HTML5:lle vuoden 2014 loppuun mennessä. Huolimatta kandidaattituksesta, suurin osa selaimista tukee jo HTML5 standardia. Lisäksi HTML5 on suhteellisen yhteensopiva takautuvasti vanhempien selainten kanssa, joten se on jo nyt täysin käytettävissä oleva vaihtoehto. (Fiedler [s.a])



kuva 7.

		Score	Bonus
Maxthon 4.0 »		476	15
Chrome 29 »		463	13
Opera 16 »		442	9
Firefox 24 »		414	10
Safari 6.0 »		378	8
Internet Explorer 10 »	<i>Microsoft Surface and others</i>	320	6

kuva 8.

Pöytäkoneet ja kannettavat
(selaimet)

		Score	Bonus
BlackBerry 10.1 »	<i>BlackBerry Q or Z series</i>	485	11
Opera Mobile 15 »	<i>Android</i>	428	11
Firefox Mobile 22 »	<i>Multiple platforms</i>	422	14
Chrome 25 »	<i>All Android 4 devices</i>	417	11
Opera Mobile 12.10 »	<i>Multiple platforms</i>	406	12
iOS 7.0 »	<i>Apple iPhone, iPad and iPod Touch</i>	399	9
Windows Phone 8 »	<i>Nokia Lumia 822, HTC 8X and others</i>	320	6
Android 4.0 »	<i>Samsung Galaxy Nexus</i>	297	3
Bada 2.0 »	<i>Samsung Wave and others</i>	283	9
Nokia Belle FP 2 »	S60 5.5 <i>Nokia 808 PureView and others</i>	272	9
Nokia Xpress »	<i>Windows Phone or S40</i>	239	2

kuva 9.

Älypuhelimet
(selaimet)

pisteytykset ovat suuntaa antavia ja alati muuttessa
taulukot [HTML5 test](#) sivustolta [05.10.2013]

		Score	Bonus
Opera Mobile 15 »	<i>Android</i>	428	11
Firefox Mobile 22 »	<i>Multiple platforms</i>	422	14
Chrome 25 »	<i>All Android 4 devices</i>	417	11
RIM Tablet OS 2.1 »	<i>BlackBerry PlayBook</i>	411	9
Opera Mobile 12.10 »	<i>Multiple platforms</i>	406	12
iOS 7.0 »	<i>Apple iPhone, iPad and iPod Touch</i>	399	9
Silk 2.2 »	<i>Amazon Kindle Fire</i>	358	13
Internet Explorer 10 »	<i>Microsoft Surface and others</i>	320	6
Android 4.0 »	<i>Asus Transformer Prime and others</i>	297	3
webOS 3.0 »	<i>HP TouchPad</i>	224	7

kuva 10.

Tabletit
(selaimet)

		Score	Bonus
Sony Playstation Vita »		284	4
Nintendo Wii U »	NetFront NX 2.1	258	4
Xbox 360 »	Internet Explorer 9	120	5
Nintendo 3DS »	NetFront	109	0
Sony Playstation 3 »	NetFront	68	0

kuva 11.

Pelialustat
(selaimet)

pisteytykset ovat suuntaa antavia ja alati muuttessa
taulukot [HTML5 test](#) sivustolta [05.10.2013]

			Score	Bonus
Toshiba »	Espial 6.0.8	<i>Toshiba L7200 televisions</i>	365	6
Sharp Aquos »	Espial 6.0.10	<i>Sharp Aquos televisions</i>	365	6
Sony Internet TV »	Opera Devices 3.2	<i>Sony televisions and Bluray players</i>	357	8
Philips NetTV »	Opera Devices 3.2	<i>Philips televisions</i>	342	16
GoogleTV »		<i>Sony Internet TV, Logitech Revue an...</i>	341	8
Toshiba »	NetFront NX 2.1		325	2
Samsung Smart TV 2012 »		<i>Samsung televisions</i>	302	12
LG NetCast 2012 »		<i>LG televisions</i>	299	8
Panasonic Smart Viera »		<i>Panasonic Viera televisions</i>	240	2
Boxee »		<i>Boxee Box by D-Link, Iomega TV wit...</i>	222	14

kuva 12.

Äly-Tv:t
(selaimet)

Suoranaisesti HTML5:n käyttö ei ole edellytys responsiiviselle suunnittelulle, sivuista voi tehdä joustavia ilman sitäkin. Responsiivisesti suunnitellut verkkosivut rakennetaan kuitenkin yleensä tulevaisuutta ja mobiililaitteita silmälläpitäen ja HTML5 tukee erityisesti näitä alueita. Kielen rakenne ja tehokkuus suosivat mobiililaitteita ja se näyttäisi olevan rakentumassa internetin seuraavaksi kielistandardiksi. Kieli tukee myös CSS3:n ja Javascriptin ominaisuuksia ja sallii toimintoja, jotka ennen vaativat ulkoisten lisäosien kuten Adobe Flashin tai Microsoft Silverlightin käyttöä. (Wołejko 2013)

5.3.2. Uusia elementtejä

HTML5 tuo mukanaan koko joukon uusia rakenne-elementtejä, jotka on luotu tulevaisuutta silmälläpitäen kuvaamaan omaa sisältöään. Käytännössä elementit toimivat perinteisen div-elementin korvikkeina, eivätkä toiminnoiltaan juuri eroa tästä. Elementtien oikeaoppinen käyttö onkin täysin koodaajan vastuulla ja on vielä standardien puuttuessa hivenen tulkinnanvaraista. Tarkoituksena näillä elementeillä on, että jonain päivänä hakukoneet pystyisivät paremmin erittelemään sisältöjä käyttäen tarkempia rakenne-elementtejä hyväkseen.

<nav>: Kuten arvata saattaa <nav> elementti on tarkoitettu pitelemään verkkosivulla liikkumiseen liittyviä linkkejä, navigaatiopalkkia tai muuta keskeisesti sivuilla navigoimiseen liittyvää elementtiä.

<section>: Sisältää elementtiryhmän tai kokonaisuuden. Kuvastaa esimerkiksi kokonaista kommenttia kaikkine osineen, artikkelia otsikkoineen ja muine lisukkeineen.

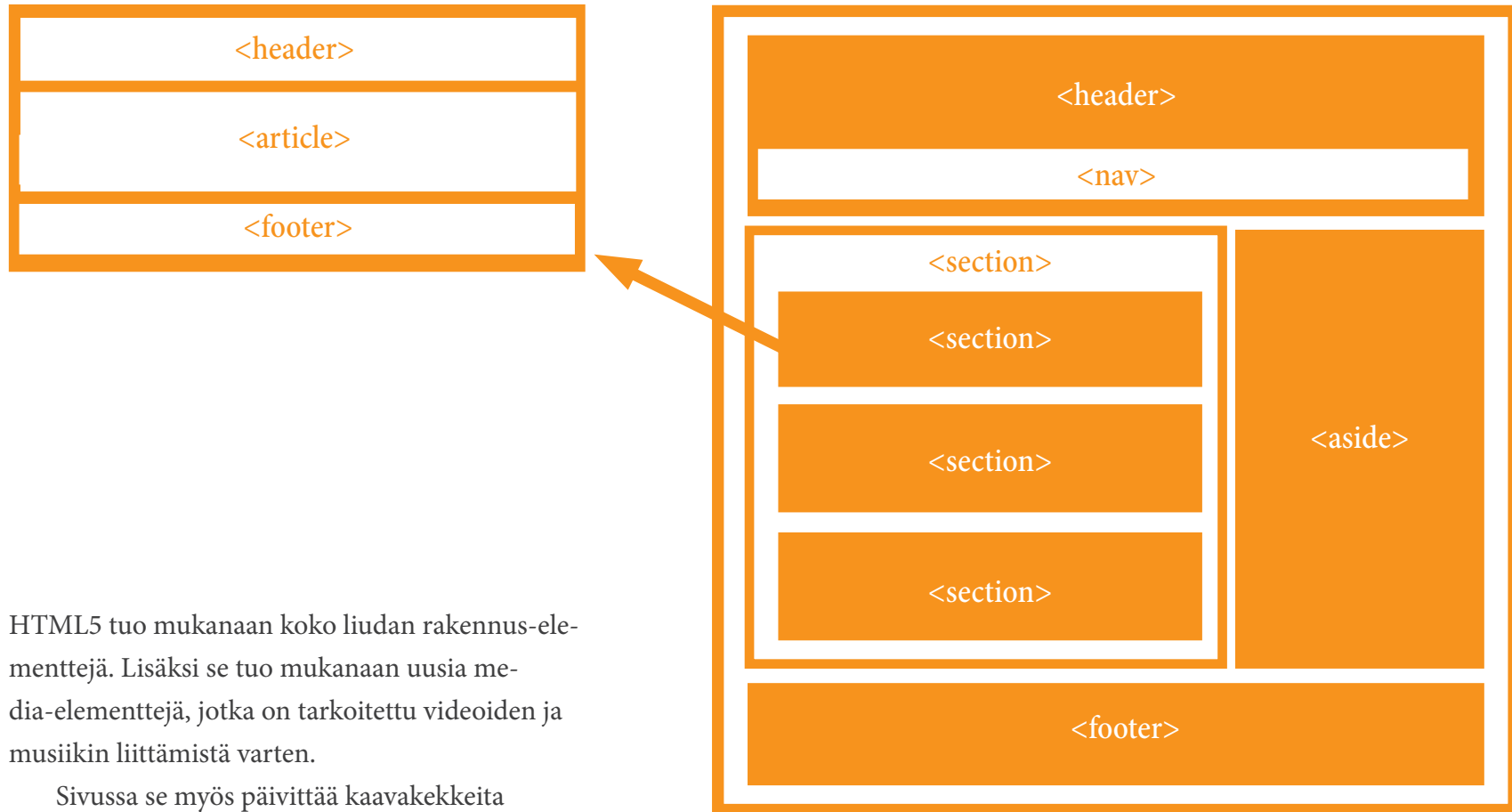
<article>: Sisältää itsenäistä sisältöä esimerkiksi artikkelin sisältöä, kommentin tekstiosaa tai vastaavaa osuutta. Tärkeintä

on, että sisältö on itsessään kokonaisuus. W3C validaattori (työkalu, jolla voi tarkistaa koodin oikeellisuuden) jopa valittaa, jos yrität sijoittaa muita elementtejä <article> elementin sisään.

<footer>: Tuttu nimitys verkkosivun alatunnisteelle, mutta HTML5 <footer> elementin merkitys on laajempi. <Footer> sisältää lisätietoa tekijästä, paikasta, osoitteista tai vaikka ajasta. Esimerkiksi artikkelin lähetysaika ja kirjoittaja voidaan merkitä <footer> elementin sisään.

<header>: Sisältää yläbannerin tai minkä tahansa elementin joka aloittaa kokonaisuuden. Esimerkiksi otsikot ja jopa navigaation voi laittaa <header> elementin sisään.

<aside>: Sisältää sivukommentin tai vähemmän tärkeän artikkelin, esimerkiksi monilla sivuilla vilisevät sivuutiset ja ilmoitukset voi laittaa <aside> elementtien sisään. Tämän elementin sisus on ensimmäinen ehdokas, jos sivulta on tarpeen piilottaa turhaa sisältöä.



HTML5 tuo mukanaan koko liudan rakennus-elementtejä. Lisäksi se tuo mukanaan uusia media-elementtejä, jotka on tarkoitettu videoiden ja musiikin liittämistä varten.

Sivussa se myös päivittää kaavakekkeita (**form**) lisäämällä uusia toimintoja. Kuten kyvyn asettaa kaavakeen kenttä pakolliseksi tai rakentaa oma liukuvalitsin tai selausvalikko.

5.3.3. Dokumenttityypin määrittely (doctype)

HTML uudistaa myös aiemmin käytettyä dokumenttityypin määrittelyä eli `<!doctype>` merkintää. Hankalasti muistettava ja ikuinen päänvaiva:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">.....
```

Voidaan nyt korvata huomattavasti helpommin muistettavalla versiolla...

```
<!doctype html>
```

Käytettävän merkistön metailmoitus lyhenee myös.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<meta charset="utf-8" />
```

Myös CSS merkintä on hitusen yksinkertaisempi

```
<link href="style.css" rel="stylesheet" type="text/css">
```

```
<link rel="stylesheet" href="style.css"/>
```

5.3.4. Viewport

Seuraava merkintä ei ole osa HTML5:n uudistuksia, eikä varsinaisesti minkään virallisen standardin mukainen merkintä, mutta jos puhumme responsiivisesta suunnittelusta on sen maininta ainakin toistaiseksi tarpeen.

Mobiiliselaimilla on nimittäin ikävä tapa olettaa, että käyttäjä on selaamassa suuremmille näytöille tarkoitettuja verkkosivuja. Tästä syystä ne **tapaavat zoomata näkymän väkisin niin kauas, että käyttäjä näkee sivun kokonaisuudessaan**, eikä joudu vain tuijottamaan sivun vasenta yläkulmaa. Perinteisen verkkosivun kohdalla käytäntö palvelee tarkoitusta loistavasti ja auttaa käyttäjää. Valitettavasti käytäntö pakottaa responsiivisesti skaalautuvat sivut täyteen näkymään.

Esimerkiksi iOS laitteiden Safari olettaa perusasetuksena sivujen kooksi 980 pikseliä ja vastikkeeksi zoomaa näkymää ulos kunnes saa koko sivun näkymään 320 pikselin alueellaan (malli iPhone4).

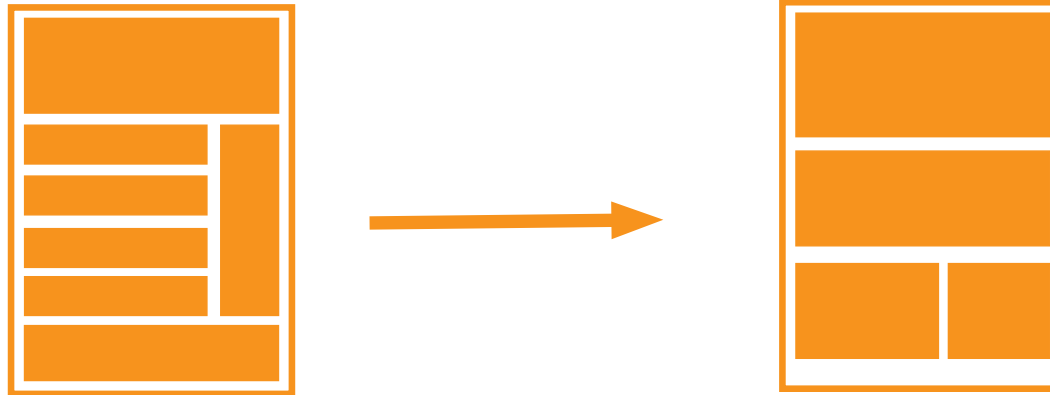
Kuullostaa responsiivisuuden kannalta hankalalta, mutta onneksi ongelmaan on ratkaisu, vaikka se ei olekaan virallinen

menettely tai tue kaikkia selaimia täysin.

```
<meta name="viewport" content="width=device-width" />
```

Yksinkertainen meta-merkintä lisätty sivun HTML koodiin ohittaa selaimen oman sisäisen näkymäasetuksen ja käskää sitä näyttämään sivun käyttäen laitteen omaa natiivia resoluutiota, eli resoluutiota, joka responsiivisuuden suunnittelijalla oli kehitysvaiheessa mielessä.

Meta-asetukseen voi lisätä myös koko joukon erillisiä asetuksia, joilla voidaan hallita kuinka paljon kävijä voi vaikuttaa näkymään zoomaamalla sitä. Jotkut vastustavat merkinnän käyttöä, sillä se voi rajoittaa käyttäjän kykyä hallita näkymäänsä. Verkkosivua suunnitellessa kannattaakin ottaa huomioon, että lukijan vaihtoehtoja ei kannata rajoittaa liikaa.



Merkintä ei ole virallisesti tuettu, vaikka suurin osa selaimista tukeekin sitä. Lähinnä kyse on eräänlaisesta puolivirallisesta laastariratkaisusta. Toinen ehdotettu ratkaisu ongelmaan on uusi CSS merkintä `@viewport`, jonka omaksumista Microsoft ajaa takaa tukemalla sitä IE10+ selaimessaan.

Toistaiseksi ei ole selvää tuleeko toisesta menetelmästä standardi ratkaisu, mutta viewport metamerkintä on parhaillaan enemmän käytetty ratkaisu.

Varmuuden vuoksi mikään ei estä käyttämästä molempia merkintöjä, sillä ne eivät aiheuta ongelmia keskenään, kunhan molemmilla on samat määrittelyt.

```
<meta name="viewport" content="width=device-width" />
```

meta osiossa

```
@viewport { width: device-width; }
```

CSS:n puolella.

Varmuuden vuoksi tosin CSS merkintään kannattaa lisätä selainkohtaiset variaatiot kuten kaikkien uudempien merkintöjen kohdalla.

```
@-ms-viewport { width: device-width; }
```

```
@-o-viewport { width: device-width; }
```

IE ja Opera -selaimille vastaavasti suunnatut argumentit

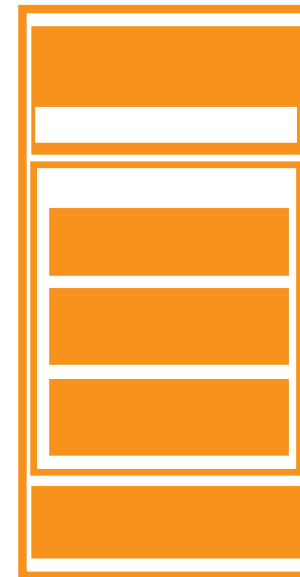
(Bovens 2012, Cant 2012, Koch [s.a])

5.4 Media Queries (CSS3)

5.4.1 Perusidea

Kaikista CSS3:n tuomista ominaisuuksista media queryt ovat ehdottomasti responsiivisen suunnittelun kannalta tärkein uudistus. Käytännössä media queryt ovat jo olemassa olevan media attribuutin laajennus, sallien sivunkehittäjän luoda sääntöjä, jotka koskevat esimerkiksi vain tiettyä ruudunleveyttä tai näkymää.

Esimerkiksi voimme ohjeistaa CSS tiedostoa piilottamaan tarpeettomia elementtejä sivulta, kun sivua katsellaan laitteella, jonka näytöllä elementeille ei ole riittävästi tilaa. Vaihtoehtoisesti, voimme ottaa käyttöön kokonaan erilaisen navigaatiopalkin, kun tila alkaa käydä ahtaaksi.



Käytännön esimerkki media querysta:

```
@media only screen  
and (min-device-width : 768px) {  
    CSS käsky (tee tekstistä punaista)  
}
```

Niin kauan kuin näkymän leveys on vähintään 768 pikseliä, teksti on punaista. Jos leveys on vähemmän kuin 768 pikseliä, sääntö ei enää ole voimassa.

```
@media only screen  
and (max-device-width : 1024px) {  
    CSS käsky (tee tekstistä punaista)  
}
```

Niin kauan kuin näkymän leveys on korkeintaan 1024 pikseliä, teksti on punaista. Jos näkymän leveys on enemmän kuin 1024, pikseliä sääntö ei ole enää voimassa.

```
@media only screen  
and (orientation : landscape) {  
    CSS käsky (tee tekstistä punaista)  
}
```

Jos laite on vaakakuva-tilassa teksti on punaista. Jos laite on jossain muussa tilassa (kuten pystykuva/portretti tilassa), sääntö ei ole voimassa.

(Powers 2011)

5.4.1 Murtumispisteet (breakpoints)

Murtumispisteellä tarkoitetaan kokoa, jossa verkkosivun ulkoasu, muuttuu media queryjen ansiosta merkittävästi. Yleensä nämä ovat **kokoja, joissa** kehittäjä olettaa kävijän käyttävän tablettia tai älypuhelinlaitea ja **sivun layout muuttuu** vastaavasti.

Yleisin tapa (mitä taittopohjaa sitten käytetäänkin) on hyödyntää kahdesta kolmeen murtumispistettä, jotka on yleensä kohdistettu tableteille ja älypuhelimille. Pisteiden määrällä ei ole mitään tiettyä rajaa ja niitä voidaan halutessaan rakentaa, vaikka jokaiselle mahdolliselle laitekoolle.

Välttämättä breakpointtien kanssa ei silti kannata lähteä liioittelemaan. Vaikka kuullostaisi hauskalta rakentaa sivu, jolla on kymmenittäin breakpointteja Hay muistuttaa verkkoyhteisössään, ettei ole järkeä lisätä tarpeettomia murtumapisteitä, jos yksinkertainen skaalauskin toimii tarpeeksi hyvin. Hän myös mainitsee, ettei pisteitä kannata suunnitella pikseli pikseliltä tietyille merkkilaitteille kuten Apple iPadille ja Samsung Galaxy Tabille, vaan on järkevämpää suunnitella sivuston breakpointit yleisluontoisesti laitetyypin mukaan, eli tehden oman pisteen

älypuhelimelle, tableteille, tv:lle, pöytäkoneelle ja niin edelleen. Tämä käy järkeen, sillä kaikkien laitteiden palvelemisen ongelma on juuri se alkuperäinen ongelma, jota varten responsiivista tekniikkaa kehitetään. **Ei ole tarpeen tehdä designista monimutkaisempaa, jos sillä ei saavuteta mitään.** Yksinkertaisempi kaava parantaa myös sivun yhdenmukaisuutta. (Hay 2013)



Dynaamisuuuden voi viedä breakpointtien kanssa vieläkin pidemmälle, onhan se responsiivisen suunnittelun bravuuri. Jotkut kehittäjät kyseenalaistavat kokonaan tarpeen kohdentaa murtumispisteitä tietyille laitetypyeille. Greig ja Gemert puhuvat blogikirjoituksissaan breakpointtien asettamisesta palvelemaan sivua selaimen sijaan. Käytännössä idea perustuu siihen, että **breakpointit asetetaan kohtiin, joissa verkkosivun sisällön muuttaminen on loogista** luettavuuden kannalta, etukäteen päätettyjen laitemittojen sijaan. (Greig 2012, Gemert 2013)

Esimerkkinä voimme rakentaa CSS käskyn joka **piilottaa navigaatiopalkin kun lukulaitteen kuvanleveys on 860 pikseliä**. Tämä on esimerkki breakpointista, joka on kohdennettu 860 pikselin leveyttä käyttäville laitteille.

Vaihtoehtoisesti voimme yksinkertaisesti rakentaa säännön, joka **piilottaa navigaatiopalkin kun selamessa ei ole sille tilaa**. Käytännössä tämä piste löytyisi testaamalla sivua ja kokeilemalla mistä oikea leveys löytyy. Tällä ajattelumallilla rakennettu breakpoint palvelee sivua, eikä tiettyä pikselimittaa käyttäviä laitteita. Metodi voi kuitenkin olla verkkosivusta riippuen työläs tai hankala ja menetelmän tehokkuus riippuu pitkälti sivun kehittäjästä.

5.5 . Edistyneemvät toiminnot

5.5.1. Javascript

Javascript on kevyt ohjelmointikieli, jota voidaan hyödyntää HTML:n kanssa, kun sivuille halutaan lisätä edistyneempiä toimintoja tai ominaisuuksia. Kieli on erittäin tehokas ja sen avulla verkkosivu voidaan muuntaa kokonaiseksi nettisovellukseksi (Web Application). Raja verkkosivujen ja verkkosovellusten välillä onkin hämärtyvässä.

Javascriptia käyttääkseen se on kirjoitettava HTML:n sekaan erityisiä script merkintöjä käyttäen tai liitettävä omaksi ulkoiseksi tiedostokseen. HTML5:n myötä Javascriptia tuetaan nyt myös HTML:n natiivina ohjelmointikielenä, jolloin sen käyttö ei enää vaadi erillistä käytettävän kielen määrittelyä.

(Ennen)

```
<script type="text/javascript" >  
    koodi  
</script>
```

(HTML5)

```
<script>  
    koodi  
</script>
```

5.5.2. jQuery

jQuery on Javascriptin ohjelmakirjasto, ja käsite saattaa kuulostaa hankalalta, mutta lyhyesti sanottuna kyse on joukosta valmiiksi rakennettuja koodinpätkiä, joiden avulla rakentaa toimintoja sivulle. Ideana on, että Javascript koodia kirjoitettaessa samantapaisia koodinpaloja käytetään usein. jQuery kokoaa näitä usein käytettyjä pätkiä ja tarjoaa niiden toiminnallisuuden lyhyemmässä muodossa. Vaihtoehtoisesti jQuerya voidaan käyttää hoitamaan vaikeita osuuksia, joiden kanssa koodaaja olisi muuten joutunut tuskailemaan. Esimerkiksi joidenkin animaatioiden lisääminen on todella helppoa, sillä jQueryä käyttäen pitkä koodikirjoitus voidaan tiivistyä yhdeksi käskyksi.

Kirjasto on vapaata lähdekoodia ja näin vapaasti kaikkien käytössä. Sen toimintoja käyttääkseen kirjaston tekstiversio on kuitenkin ladattava sivulle itselleen ja verkkosivun HTML koodiin on lisättävä viittaus tähän tiedostoon. Tiedoston voitoin linkittää ulkoisena, esimerkiksi Google tarjoaa kirjastoa suoraan omilta servereiltään, ettei scriptitiedostoa tarvitse sijoittaa lokaalisti sivun varsinaiseen sijaintiin. Googlen serverit ovat

tunnetusti tehokkaita, joten suurimmassa osassa tapauksia tiedoston linkittäminen näiden kautta on nopeampaa kuin lokaalin tiedoston käyttö. (Ward 2013)

6. Demosivu

6.1. Mitä ja kenelle

Opinnäytteen käytännön osuutena rakensin demosivun, jonka tarkoituksena on havainnollistaa responsiivisen suunnittelun ideologiaa ja tekniikoita. Koska opinnäytteen aiheena on responsiivisuus, eikä visuaalisuus, keskityn kuvauksessa lähinnä sivun toiminnallisuuden valintoihin ja näiden valintojen syihin. Graafisen ilmeen eli ulkoasun suunnittelun jätämme taka-alalle, ellei valinnoilla ole suoraa yhteyttä sivun toiminnallisuuteen.

Vaikka kyseessä on vain demonstraatio on sivusuunnittelu helpompaa, jos työllä on selkeä teema ja päämäärä. Responsiivisuuden kokeilemisen nimissä halusin rakentaa sivun, joka on muuta kuin staattinen yhteystietosivu. Halusin sen näyttävän interaktiiviselta ja laajemmalta, kuin mitä se on. Sisällön tuli näyttää rikkaalta tai ainakin sisällön laajentamiselle oli jätettävä perustellusti tilaa. Lisäksi halusin antaa demolle yhteisöllistä näkökulmaa, millä en tarkoita Facebook/Twitter linkkejä jokaisen elementin alareunassa, vaan käyttäjäprofileja ja kävijäkohtaisia asetuksia(vain simuloidusti, varsinaisia tallennettavia ominaisuuksia ja oikeaa toimivaa rekisteröitymistoimintoa en lähtenyt


demolle rakentamaan käytännön syistä). Aikani suunniteltua- ni otin demosivun aiheeksi kuvitellun yhteisösivun deviantAr- tin ja blogisivustojen hengessä.

Sivuston nimi: pixelhome

Käyttäjäkunta: Rekisteröityneet jäsenet, joukossa itsenäisesti sisältöä tuottavia jäseniä. Jäsenet ovat visuaalisesti suuntautuneita, aikaansa seuraavia ja omapäisiä.

Tarkoitus: Oleskelu ja kommentointisivusto. Julkaisee artikkeleita ja ylläpitää omaa galleriaa, joka koostuu jäsenten tuotoksista.


Funktio: Sivusto julkaisee artikkeleja ja toimii eräänlaisena oleskelutilana, jossa ryhmän jäsenet voivat kommunikoida keskenään ja tutustua toistensa töihin, sekä kommentoida näitä. Artikkeleista huolehtivat seniorijäsenet ja sivuston ylläpito. Sivustoa käyttääkseen (kommentointi/gallerian sisäänkäynti) on kävijän rekisteröidyttävä ja kirjaututtava sisään omalla käyttäjänimellään. Useimpien yhteisö sivujen mukaisesti jokaisella käyttäjällä on omat asetukset ja profiilikuva.

Home Forum Gallery VGallery Storymode  malicious potato

piXelhome: Long Road Ahoy!

Authored by malicious potato // posted 24.09.2013

piXelhome is an online community showcasing various forms of user-made artwork. It was first launched on August 7, 2000 by Scott Jarkoff, Matthew Stephens, Angelo Sotira and others.



Site started as a site connected with people who took computer applications and modified them to their own tastes, or who "deviated" the applications from the original designs. As the site grew, members in general became known as "deviants" and submissions as "deviations". DeviantArt was originally launched on August 7, 2000 by Scott Jarkoff, Matt Stephens, Angelo Sotira and others, as part of a larger network of music-related websites called the Dmusic Network. The site flourished largely because of its unique offering and the contributions of its core member base and a team of volunteers after its launch, but was officially incorporated in 2001 about eight months after launch. On November 14, 2006, deviantArt gave its users the option to submit their works under Creative Commons licenses giving the artists the right to choose how their works can be used. A Creative Commons license is one of several public copyright licenses that allow the distribution of copyrighted works.

[quote](#) [multiquate](#) [report](#)

Second chance

Authored by malicious potato // posted 24.09.2013

Week 26: brings more to the table

Occasionally, deviantArt hosts members to come together to interact, exchange, and have been meetings for the birthday called 'Birthday Bashes', as well as general get-togethers around

Uprising it the making

The site has over 265 million have been uploaded by its members. As of July 2011, the largest internet art platform. deviantArt may leave comments on individual deviation pages site to be called "a [free] peer application".

Really running out of

This is side commentary. On showcasing various forms of artwork

6.2. Pohjan luonti

En erittele tässä osiossa erikseen, jokaista sivun rakentamisen vaihetta, sillä suurin osa prosessista ei poikkea mitenkään perinteisen verkkosivun rakentamisesta. Nostan kuitenkin esille responsiivisuuteen ja käyttökokemukseen liittyvät seikat. Demon laajuudeksi päätin kolmesta neljään sivua, joista jokainen toimisi hivenen eri tavalla.

Päätin jo aloittaessani, ettei sivulla käytettäisi täysin joustavaa taittoa, sillä käytän suurikokoista näyttöä ja tiedän miten pahasti sisältö voi venyä, jos minkäänlaisia maksimimittoja ei määrätä. Sopivaksi maksimileveydeksi sivulle päätin 1050 pikseliä. Toteutin maksimileveyden lukitsemisen media querylla, joka lukitsi sivun sisältöelementin oikeaan mittaan kun selaimen leveys ohitti tietyn mitan. Toinen vaihtoehto olisi ollut antaa elementille suoraan maksimimitta.

Sivun varsinaiseksi leveydeksi määräsin 90 % käytettävästä leveydestä (Vain 1050 pikselin maksimileveyteen saakka) ja lopuksi vielä keskitin materiaalin, jotta sisältö ei jäisi suurimmilla näytöillä selaimen nurkkaan.

```
@media only screen
and (min-width: 1167px{
  .content{
    width:1050px;
  }
}

.content{
  max-width:1050px;
}
```

**Molemmat toteuttavat
pitkälti saman asian, mutta
eri tavalla**

Päädyin pian tulokseen, että liika skaalautuminen sai sivun tuntumaan epämukavalta. Kaikkien elementtien ei ollut tarpeen muuttaa kokoaan luettavuuden säilyttämiseksi. Sivun tuntui huomattavasti rauhallisemmalta kun, osa elementeistä toimi maamerkkeinä ja säilytti kokonsa. Muutin muun muassa osan marginaaleista staattisiksi mitoiksi, jottei sivu ”pomppisi” ikkunan kokoa muutettaessa.

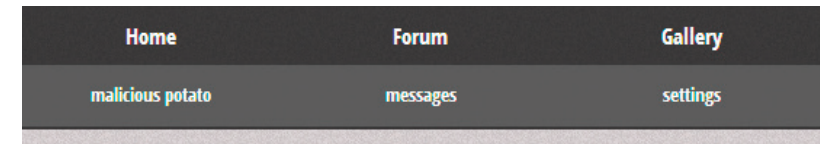
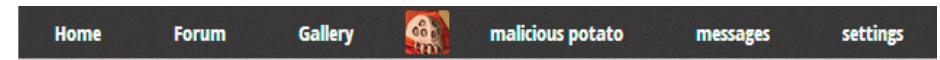
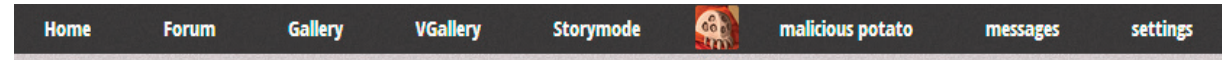
Fonttikokojen määritykseen käytin Chaefferin suosituksen mukaisesti prosenttiarvoja. Sillä tällöin käyttäjä pystyy selaimensa asetusten kautta vaikuttamaan fontin kokoon, jonka näkee omalla laitteellaan. (*Chaeffer, 2011*)

Kehitysvaiheessa suunnittelin, että sivulle tulisi kaksi tärkeää breakpointtia. Yksi tabletille ja toinen älypyhelimelle. Sivun kehittyessä huomasin kuitenkin, ettei tablet alusta oikeastaan tarvinnut suurempia muutoksia sivun taittoon, sillä kiitos tilavan asettelun elementit asettuivat siististi ja luettavasti pienemmässäkin näkymässä. Galleriaa lukuun ottamatta erillistä tablet breakpointtia ei tarvittu.

```
@media only screen
and (max-width: 640px) {
.hideOnMobile {
display:none;
}
}
```

CCS argumentti, joka piilottaa kaikki elementit joiden luokka on `hideOnMobile` kun lukulaitteen leveys on alle 640 pikseliä

Erittäin hyödyllinen kun halutaan piilottaa sisältöä mobiilikäyttäjiltä



kuvasarja 1.

Navigaatiopalkin muutos

6.3 Navigaatio

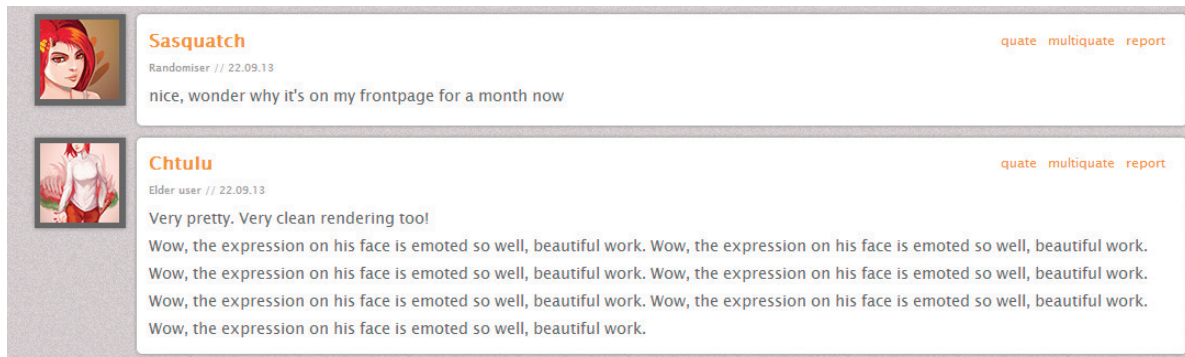
Navigaation voi toteuttaa monella tapaa. Yleisin on sivulle keskitetty poikittainen navigaatiopalkki tai vasemman laidan pystynavi. Tällä kertaa halusin hyödyntää kaiken mahdollisen horisontaalisen tilan ja sivujen responsiivisuuden. Päätin myös jättää suuren tilaavievan bannerin toteuttamatta, sillä sivun teemana oli toimia yhteisösivuna jolloin banneri ei toisi sivulle pidemmän päälle mitään tarpeellista hyödykettä. **Yhteisösivulla jäsenet ovat sivun omaa ilmettä tärkeämpiä.**

Asetin minimaalisen navigaatiopalkin sivun yläreunan. En halunnut se olevan liikaa tiellä, enkä saavuttamattomissa. Valinta palveli responsiivista toteutusta erinomaisesti. Lisäksi helpotin käyttäjän asemaa erottamalla tilin hallintaan liittyvät toiminnot navigaatiotoiminnoista. Yleisen käytännön mukaisesti asetin tilin hallintaan liittyvät painikkeet oikeaan laitaan ja navigaatioon liittyvät painikkeet vasempaan aloittaen etusivupainikkeesta.

Oikean yläkulman kuvasarja näyttää, miten elementit muuttuvat tilan vähetessä. Vähemmän tarpeellisia linkkejä pudotetaan pois, jolloin ne eivät ole enää käytössä mobiilinä-

kymässä, hyvä käytäntö olisi tarjota linkkien toiminnot jonkin muun reitin kautta.

Lopulta profiilikuva toimii tilanjakajana ja tilan loputtua tärkeimmät linkit tipputautuvat allekkain niin, että ne on edelleen helppo erottaa toisistaan.



kuvasarja 2.

Normaali ja tiivistetty näkymä



6.4. Kommenttipalsta

Kommenttisivun toteutuksella pyrin lähinnä havainnollistamaan, miten viestien muotoa voi tiivistää. Oikean keskustelupalstan olisi hyvä olla kompaktimpi kuin valitsemani ratkaisu. Tässä muodossa kommenttien lukeminen ja sisällön läpikäynti käy lukijalle raskaaksi. Lisäksi lukemista voisi helpottaa erotamalla kommentteja toisistaan, vaikka antamalla joka toiselle kommentille hivenen erilainen taustaväri, jolloin lukijan olisi helpompi seurata kommenttien kulkua. Jos artikkelin kommenttipalstasta haluaisi tehdä suoran keskustelupalstan olisi tarpeen luoda myös vastausominaisuus”reply” viesteille ja antaa

keskusteluketjuille syvennys tai muu erotin jolla ne eristettäisiin omiksi kokonaisuuksikseen.

Lisättäköön vielä, että Facebook /Twitter näppäinten puute on tarkoituksellista.

Mallissa (kuvasarja 2.) näkyy miten kommenttien sisältö tiivistyy ahtaassa näkymässä. Käyttäjän profiilikuva on tälle tärkeä osa identiteettiä, joten poistamisen sijaan se siirretään tilan säästämiseksi kommentin sisälle. Vastaus/raportointi painikkeet pudotetaan viestin alalaitaan, jottei lukijan tarvitse vierittää viestin näkymää päästäkseen niihin käsiksi.

6.5. Galleria

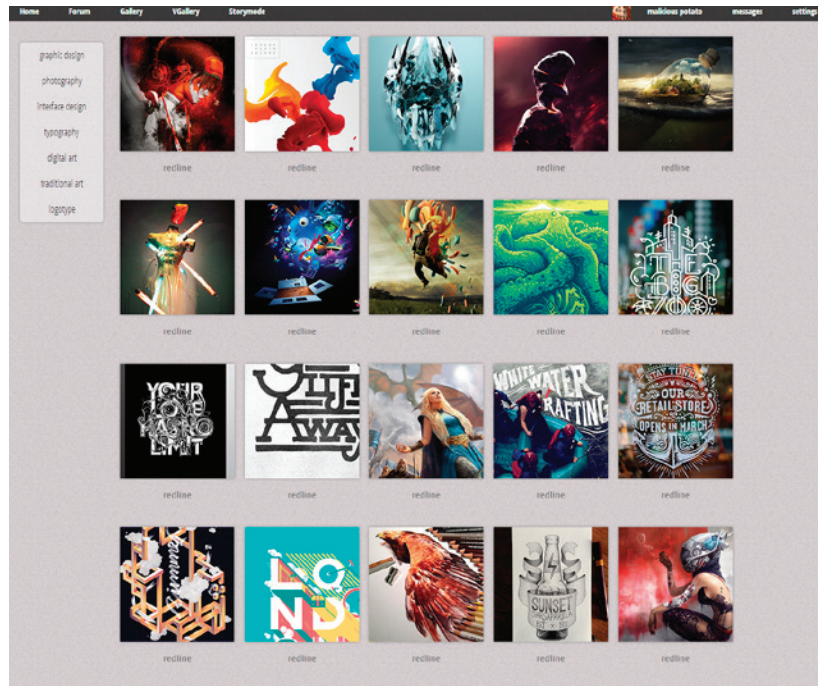
Galleria on funktioltaan erilainen kokonaisuus kuin tekstipainotteiset sivut. Kuvat ovat oleellisin osa sisältöä ja sivuston teeman vuoksi niitä on paljon. Galleria vaatii kuvien lisäksi tavan selata ja järjestellä kuvia. Tätä tarkoitusta varten rakensin uuden kiinteämittaisen valikon sivun vasempaan reunaan. Valikon olisi voinut teknisesti ottaa asettaa suoraan päänavigaation alle, mutta demonstraation nimissä päätin rakentaa perinteisemmän valikkomallin, jota yleisemmin käytetään gallerioissa.

Gallerian kohdalla poistin aikaisemmin asettamani 1050 pikselin leveysrajan käytöstä, sillä sivun sisällöllä ei ole rikkinen riskiä ja näin saatoin hyödyntää koko käytettävissä olevan tilan.

Joustavien kuvien sijaan tein gallerian kuvista vakioittaisia linkkikuvia. Eli niin sanottuja peukalokuvia (**thumbnails**). Peukalokuvien tarkoitus on olla pienempiä nettioptimoituja versioita alkuperäisistä raskaista galleriakuvista, jolloin **käyttäjän selain ei joudu lataamaan raskaita kuvia turhaan**. Halutessaan lukija pääsee käsiksi kuvan raskaaseen alkuperäiseen versioon

painamalla peukalokuvaketta. Demosivuston idea on, että jokaisella kuvalla olisi oma kommentointisivunsa, mutta demon rajoitteiden vuoksi sivu avaa linkkiä painaessa vain korosturuudun, joka on tässä tapauksessa toteutettu lightbox nimistä jQuery liitännäistä käyttäen.

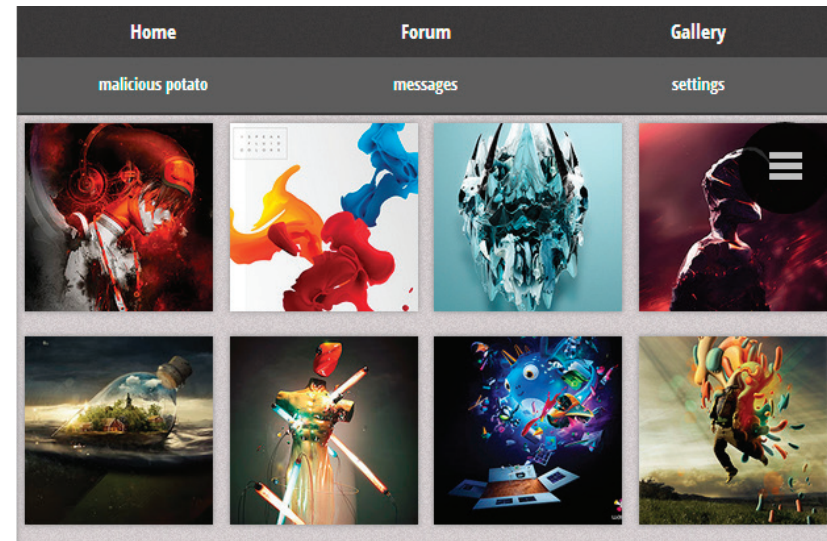
Staattisen kuvakoon valinnalle peukalokuville on omat syynsä. Kuvien liiallinen skaalaaminen voi pilata niiden laadun, erityisesti kun kyse on kevyistä peukalokuvista, joiden laatu liian läheltä tarkasteltaessa ei ole imarteleva. Lisäksi kuvat soljuvat näin niille varatussa tilassa, rivittäen itsensä automaattisesti näkymään sopiviksi, jolloin saamme aikaan helppohoitaisen kuvagallerian.



kuvasarja 3.

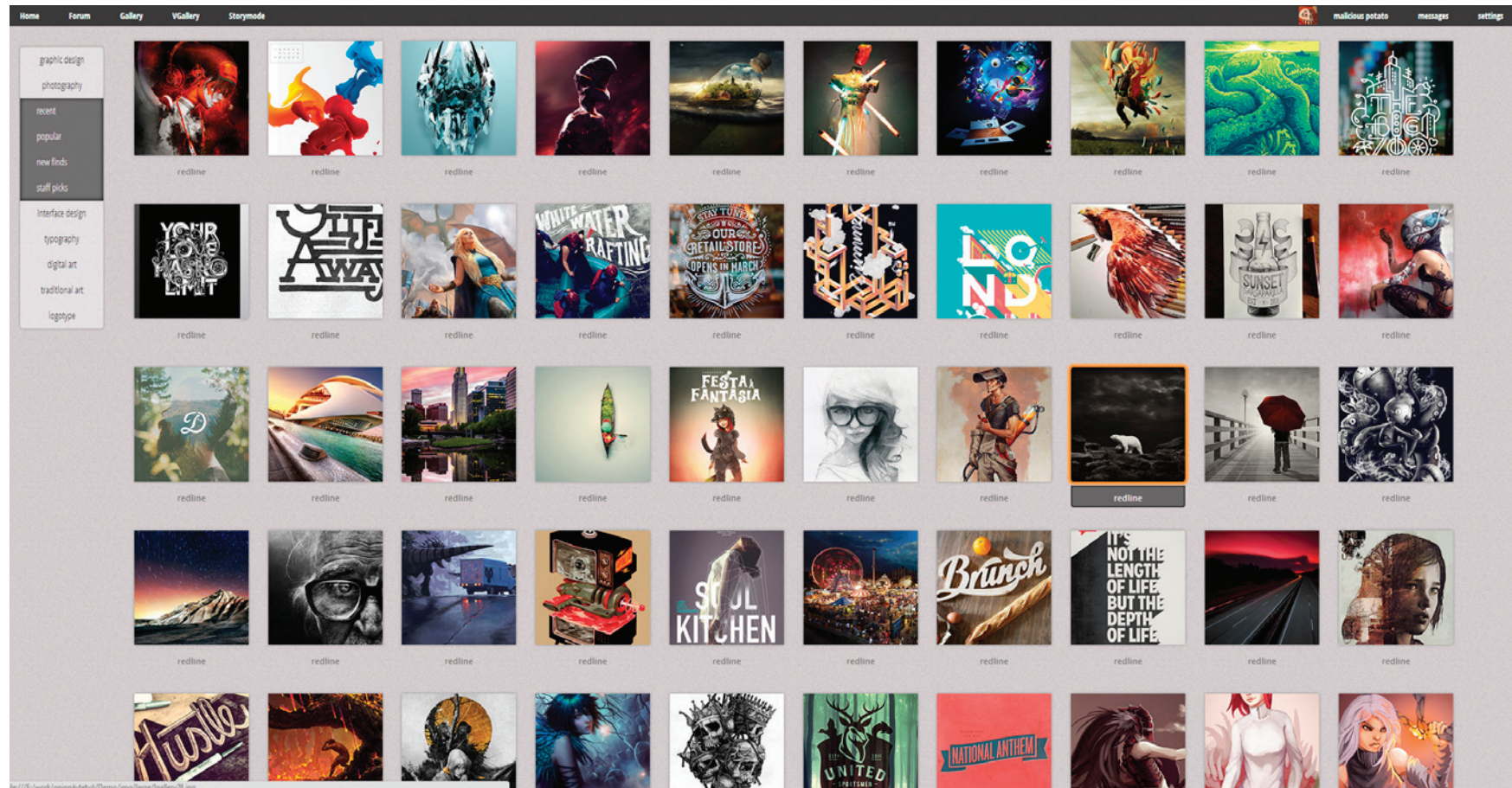
Gallerian ainoa skaalautuva osa täydessä näkymässä on peukalokuvien välinen tyhjä tila, joka kasvaa hivenen mitä enemmän tilaa on käytössä. Muutos on kuitenkin vähäinen eikä merkittävästi muuta kuvien asettelua. Peukalokuvien välinen tila pystysuunnassa on kiinteä, eikä muutu ennen murtumispistettä.

Gallerian asettelu muuttuu merkittävästi tabletkoossa. Kuvien nimet piilotetaan ja tyhjää tilaa vähennetään merkittävästi. Kuvat säilytetään edelleen sivun tärkeimpänä sisältönä.



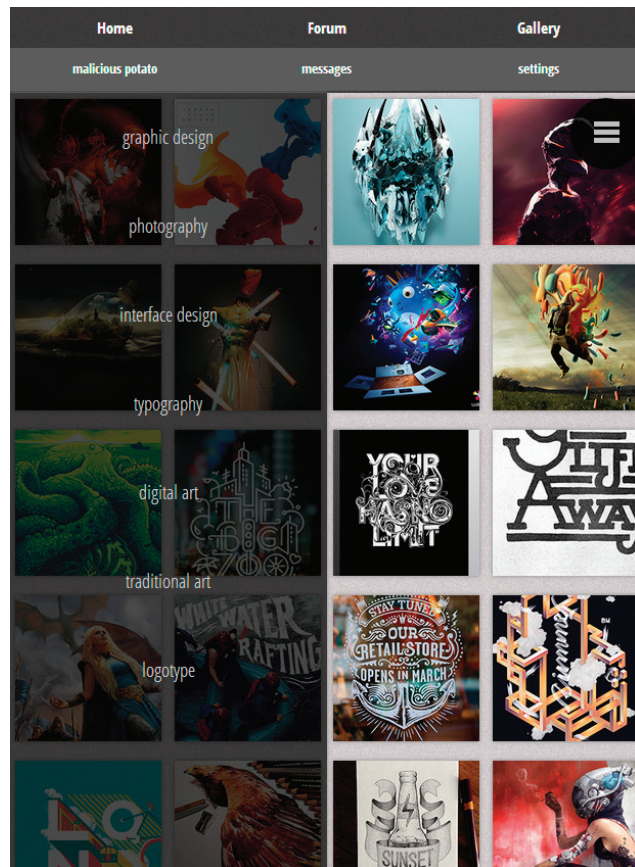
960 pikselin ja 640 pikselin näkymät
(pöytäkone ja tablet)

Vasemman laidan menuvalikko on tarpeellinen, mutta tiellä, joten se piilotetaan uuden näppäimen taakse. Näppäimessä käytetty kolmiraita ikoni on yleistymässä tarkoittamaan valikkoa ja käytänkin sitä kertomaan käyttäjälle mitä painikkeen takaa löytyy.



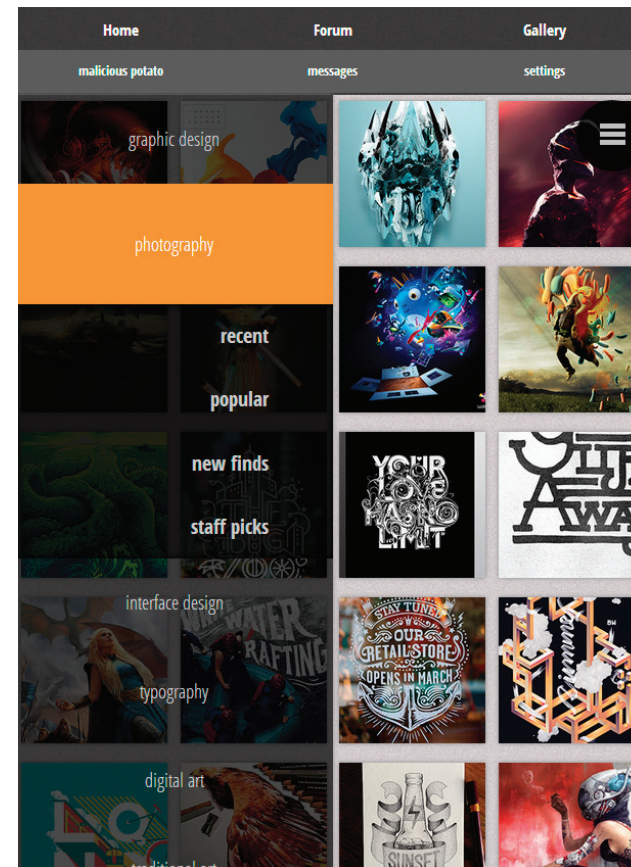
kuva 14.

Galleria, Näkymä 2560 x 1440 näytöllä



kuva 15.

Pienen näkymän valikko
(oikean yläkulman menupainikkeen alla)



kuva 16.

Pienen näkymän valikon lisäpainikkeet
(aukeavat kun genre valitaan)

6.6. Optimointi

Verkkosivujen optimointi on oma taiteenlajinsa. Tyypillinen sivuoptimointi koostuu joukosta useita pieniä askareita ja voi helposti kuullostaa naurettavalta, mutta jo pienen odottelun on todettu voivan karkottaa käyttäjä. Aikaisemmin ajateltiin, että nopeiden laajakaistayhteyksien myötä pääsisimme pikkutarkasta optimaatiosta, mutta mobiililaitteiden myötä optimaatio on edelleen tärkeää eritoten responsiivisille sivuille, joiden odotetaan toimivan kaikenlaisissa laitteissa. Älypuhelin- ja tablettien verkkonopeudethan vaihtelevat suuresti ja erityisesti mobiiliverkon kautta tapahtuva asiointi nopeutuu kun ladattavia tiedostoja on vähemmän. Mobiiliin suunnittelun liittyäisiin useita koodin puolella tehtäviä optimointikeinoja. Helpompia optimoimisen keinoja on muun muassa nämä:

Raskaiden tiedostojen tai kuvien laitto linkkien taakse

Säilyttää useita versioita samoista kuvista ja ladata aina selaimen sopivin versio

Ulkoisesti latautuvien scripttien säästäväinen käyttö, joillain poikkeuksilla, esimerkiksi jQuery:n lataaminen Googlen servereiltä on hyvä idea, sillä Googlen serverit ovat todennäköisesti omaa serveriäsi nopeammat.

Meillä ei ole aikomusta laittaa omaa demoamme internetiin, joten aggressiivinen optimointi ei ole tarpeen, mikäli haluaisimme kuitenkin parantaa sivun optimointia voisimme esimerkiksi tehdä gallerian kuvien peukalokuvista useamman version ja ladata vain tarpeelliset koot. (Calhoun 2011)

6.7. Testaus ja lopputulos

Valmis demosivu on testattu ja todettu toimivaksi usealla pöytäkoneselaimella (kaikki uusimpia versioita 29.09.2013) mukaanlukien: Firefox, Chrome, Opera ja IE 10+. Sivut toimivat toiminnallisuuden kannalta moitteettomasti. Sivulla on kuitenkin käytetty uusia CCS tyylimäärittäjiä kuten pyörityksiä, heittovarjoja ja läpinäkyvyyttä, joiden toistossa IE:lla oli aluksi ongelmia.

Mobiiliselaimista sivua on testattu Operan ja Firefoxin mobiiliversioilla, sekä Dolphin ja Chrome selaimella. Sivun toimii, näissäkin, mutta vaihtelevalla menestyksellä. Sivua ei ole täysin optimoitu ja galleriasivun esitys on välillä hidasta. Lisäksi Firefox-mobiiliselaimella oli pieniä ongelmia läpinäkyvyyden toistossa ja sivun Javascript vaikutti aiheuttavan tässä jonkinasteisia vakausongelmia. Ongelmat ovat kuitenkin koodilähtöisiä eivätkä koske responsiivisuutta.

Testausvaiheessa kävi ilmi, että sivu toimii hyvin pöytäkoneselaimilla ja sen responsiivisuus toimii kuten tarkoitus oli, mobiiliselainten pakko-zoomaus tuotti kuitenkin ongelmia,

kunnes viewport määrittäjä korjasi tämän ongelman mobiiliselainten näkökulmasta. Negatiivisena puolena käyttäjä ei kuitenkaan enää kyennyt itse zoomaamaan sisältöä yhtä vapaasti.

Kaiken kaikkiaan sivu läpäisee kuitenkin kokeet, sillä responsiivisuus toimii kaikilla testatuilla alustoilla. Mikäli demo olisi tulossa oikeaan käyttöön, vaatisi se kuitenkin vielä hiomista.

7. Päätäntä

Loppujen lopuksi koen responsiivisen tekniikan ja responsiivisen suunnittelun mielenkiintoiseksi. Tekniikka on selvästi syntynyt tarpeeseen ja sen opiskelu oli ehdottoman mielekästä. Oppimani asiat tulevat varmasti olemaan hyödyllisiä tulevissa työtehtävissäni. Uskon responsiivisuuden itsessään olevan tärkeä osa alani tulevaisuutta.

Mitä opinnäytteen itsensä toteutukseen tuli suurin osa ongelmistani kohdistui omaan aikataulutukseeni ja lähdemateriaalin kieleen. Tuotti jonkinasteisia ongelmia selittää asioita suomeksi, kun kaikella sanastolla ei ole varmaa suomenkielistä käännöstä eivätkä monet termeistä ole edes yhdenmukaisia. Pahimmassa tapauksessa alkuperäiskielelläkin asioista käytetään termistöä ristiin. Yritin välttää kielen lipumista liian tekniseksi, mutten ole varma, onnistuinko aikeessani, vai sainko vain pois-suljettua oleellista sisältöä.

Lähdemateriaaleissa oli häiritsevän paljon päällekkäisyyksiä ja ongelmia selvyden kanssa kun käytettiin Fluid Ja Flex gridin tapaista sanastoa, eikä tässä erityisesti auttanut se, ettei lähdeaineiston kirjoittajilla tuntunut olevan yhteistä kuvaa termien käytöstä.

Aikataulutuksen ongelmat kestivät koko hankkeen ajan, kun alkuperäinen keväällä laadittu aikataulu ei pitänyt lainkaan ja koko työ siirtyi syksyyn. Kesä meni muiden projektien parissa huolimatta hyvistä aikeista. Aikataulutus on ehdottomasti suurimpia heikkouksiani ammattilaisena. Samoin myös työn temmon säätelyminen. Jumiudun edelleen helposti toimittamaan, jotakin pientä askareta ja hiomaan sitä täydellisyyten, vaikka deadline painaisi jo päälle.

Uskon, että jos olisin osannut aikatauluttaa projektin paremmin, olisin saanut aikaan merkittävästi laajemman ja paremmin hiotun selvityksen demon rakentamisesta. Lisäksi olisin toivonut pystyväni toteuttamaan opinnäytteen visuaalisen asun järkevällä aikataululla. Vaikka se ei suoraan liity aiheeseeni, olisin halunnut toteuttaa opinnäytteen kirjallisen osan visuaalisempana käyttäen enemmän omia kuvituksia tai kuvanmuokkauksia.

Kaiken kaikkiaan olen tyytyväinen lopputulokseen. Sain koottua toimivan ja alkuvaatimukset täyttävän demosivun, jonka käyttöönotto ei olisi turhan hankalaa. Lisäksi olen tutustunut kattavasti kevyeen verkko-ohjelmointiin ja luonut hyvän pohjan jatko-opinnoille, joita aion jatkossa lähteä hakemaan.

Lähteet

Collison, S. Budd, A. & Moll, C. 2007

CSS Tehokas hallinta. Laura Kuisma. 1. painos. Readme.fi Helsinki.

Sachs, T. & McLain, G, Ph.D. 2006

Back to the User - Creating user-focused web sites. New Riders Publishing. United States of America.

Korpela, J.K 2011

HTML5 - Uudet ominaisuudet. WSOYpro Oy Jyväskylä.

Frain, B 2012

Responsive Web Design with HTML5 and CSS3. Packt Publishing. Birmingham Mumbai.

Freeman, E. & Robson. E. 2011

Head First HTML5 Programming. First Edition. O'Reilly Media Inc. Sebastopol, CA.

Leiniö, T. 2012

Mitä on responsiivinen design. Blogikirjoitus. [Luettu 15.07.2013.] Saatavissa: www.sofokus.com/blogi/mita-on-responsiivinen-design

Haaga-Helia ammattikorkeakoulu - verkkojulkaisun tekeminen. verkkosivusto. 2013

Verkkodokumentti. [Luettu 15.08.2013.] Saatavissa: myy.haaga-helia.fi/~vanvu/www/suunnittelu/taittopohja.html

Marcotte, E.

Responsive Web Design. 2010. Verkkootikkeli. [Luettu 06.06.2013.] Saatavissa: alistapart.com/article/responsive-web-design

Fluid Grids. 2009. Verkkootikkeli. [Luettu 15.06.2013.] Saatavissa: <http://alistapart.com/article/fluidgrids>

Colangelo, A. 2013

The Design of Code: Organizing Javascript. Verkkootikkeli. [Luettu 15.08.2013.] Saatavissa: alistapart.com/article/the-design-of-code-organizing-javascript

Hay, S. 2013

Designing for Breakpoints. Verkkootikkeli. [Luettu 15.06.2013.] Saatavissa: alistapart.com/article/designing-for-breakpoints

Greig, S. 2012

Deciding what Responsive Breakpoints to use. Verkkootikkeli. [Luettu. 15.06.2013.] Saatavissa: tangledindesign.com/deciding-what-responsive-breakpoints-to-use

Gemert, V. 2013

Logical Breakpoints For Your Responsive Design. Verkkootikkeli. [Luettu 15.06.2013.] Saatavissa: www.smashingmagazine.com/2013/03/01/logical-breakpoints-responsive-design/

How Much Has Web Really Changed? Verkkootikkeli.

[Luettu 04.08.2013.] Saatavissa: www.smashingmagazine.com/2013/05/06/new-defaults-web-design/

Karukka, M. & Inkilä, T. 2013

Responsiivinen verkkosivujen suunnittelu mukauttaa sisällön eri päätelaitteille. ePooki. [Luettu 20.09.2013.] Oulun seudun ammattikorkeakoulun tutkimus- ja kehitystyön julkaisut 6. Saatavissa: www.oamk.fi/epooki/internet-ja-ict/responsiivinen-verkkosivujen-suunnittelu

Armstrong, C. 2012

The Infinite Grid. Verkkootikkeli. [Luettu 19.07.2013.] Saatavissa: alistapart.com/article/the-infinite-grid

Marquis, M. 2012

Responsive Images and Web Standards at the Turning Point. Verkkootikkeli. [Luettu 01.08.2013.] Saatavissa: alistapart.com/article/responsive-images-and-web-standards-at-the-turning-point

Friedman, V. 2008

Applying Divine Proportion To Your Web Design. Verkkootikkeli. [Luettu 02.10.2013.] Saatavissa: www.smashingmagazine.com/2008/05/29/applying-divine-proportion-to-web-design/

Knight, K. 2009

Fixed vs. Fluid. Elastic Layout : What's The Right One For You?. Verkkootikkeli. [Luettu 02.10.2013.] Saatavissa: coding.smashingmagazine.com/2009/06/02/fixed-vs-fluid-vs-elastic-layout-whats-the-right-one-for-you/

Koch, P.P. [s.a]

A tale of two viewports - part one. verkkootikkeli. [Luettu 15.08.2013.] Saatavissa: www.quirksmode.org/mobile/viewports.html

A tale of two viewports - part two. verkkootikkeli. [Luettu 15.08.2013.] Saatavissa: www.quirksmode.org/mobile/viewports2.html

Yates, I . 2013

Quick Tip: Don't Forget the Viewport Meta Tag. Verkkotutoriaali. [Luettu 06.07.2013.] Saatavissa: webdesign.tutsplus.com/tutorials/htmlcss-tutorials/quick-tip-dont-forget-the-viewport-meta-tag/

Fiedler, D.

Web Developer Basics: Learning About HTML5. Verkkootikkeli. [Luettu 14.06.2013.] Saatavissa: www.htmlgoodies.com/html5/Web-Developer-Basics-Learning-About-HTML5-3915046.htm#fbid=fcF5WRct20w

Wolejko, G. 2013

5 reasons Why HTML5 matters?. Blogikirjoitus. [Luettu 15.06.2013.] Saatavissa: www.cognifide.com/blogs/ux/5-reasons-why-html5-matters/

Sterling, G. 2012

Google Survey Reveals What Users Want From Mobile Sites. Verkkotutkimus. [Luettu 05.07.2013.] Saatavissa: <http://marketingland.com/google-survey-what-users-want-from-mobile-sites-22606>

Keith, J. 2010

A Brief History of Markup. Verkkootikkeli. [Luettu 15.06.2013.] Saatavissa: alistapart.com/article/a-brief-history-of-markup

Nykänen, O. 2003

W3C pähkinänkuoressa. Verkkootikkeli. [Luettu 28.08.2013.] Saatavissa: www.w3c.tut.fi/reports/2003/0113aboutw3c/

Bos, B & Wium Lie, H. 1999

Cascading Style Sheets, designing for web. 2nd edition. Addison-Wesley Professional.

Google developers - sivusto. sivukehittäjille suunnattu ohjeistus

Verkkotutoriaali/ohjeistus. [Luettu 24.06.2013.] Saatavissa: developers.google.com/webmasters/smartphone-sites/details

Ward, D. 2013

3 reasons why you should let Google host jQuery for you. Verkkoartikkeli/kirjoitus. [Luettu 29.08.2013.] Saatavissa: encosia.com/3-reasons-why-you-should-let-google-host-jquery-for-you/

Grace, J. 2013

How to Convince Your Web Development Clients to Go With a Responsive Site When They Just Don't "Get It". Verkkoartikkeli. [Luettu 02.10.2013.] Saatavissa: managewp.com/convince-clients-to-have-responsive-site

Bovens, A. 2012

Viewport meta tags used on the "top 20 responsive sites" Blogikirjoitus. [Luettu 22.09.2013.] Saatavissa: my.opera.com/ODIN/blog/2012/07/17/viewport-meta-tags-used-on-the-top-20-responsive-sites

Cant, E. 2012

CSS @viewport or Meta Tag?. Verkkoartikkeli. [Luettu 01.10.2013.] Saatavissa: <http://menacingcloud.com/?c=cssViewportOrMetaTag>

Powers, D. Expanded and updated from article originally written by Booth, D. 2011

Introduction to media queries - Part 1: What are media queries?. Verkkoohejeistus. [Luettu 10.06.2013.] Saatavissa: www.adobe.com/devnet/dreamweaver/articles/introducing-media-queries.html

Chaeffer, K. 2011

CSS Font-Size: em vs. px. pt. vs. percent. Verkkoartikkeli. [Luettu 03.07.2013.] Saatavissa: <http://kyleschaeffer.com/development/css-font-size-em-vs-px-vs-pt-vs/>

Calhoun, D. 2011

Mobile Performance Manifesto. Blogiartikkeli. [Luettu 04.08.2013.] Saatavissa: davidbcalhoun.com/2011/mobile-performance-manifesto

Kuvalähteet

kuva 1.

Ruudunkaappaus verkkosivusta Chrome selaimessa [Luettu. 04.10.2013] Saatavissa: <https://portal.savonia.fi/amk>

kuva 2.

Ruudunkaappaus verkkosivusta Chrome selaimessa [Luettu. 04.10.2013] Saatavissa: <https://portal.savonia.fi/amk>

kuva 3.

Ruudunkaappaus verkkosivusta Chrome selaimessa [Luettu. 30.09.2013] Saatavissa: <http://www.abookapart.com/products/responsive-web-design>

kuva 4.

Ruudunkaappaus verkkosivusta Chrome selaimessa [Luettu. 30.09.2013] Saatavissa: <http://www.abookapart.com/products/responsive-web-design>

kuva 5.

Ruudunkaappaus verkkosivusta Chrome selaimessa [Luettu. 30.09.2013] Saatavissa: <http://www.abookapart.com/products/responsive-web-design>

kuva 6.

Infografiikka googlen vapaasta käyttäjätutkimuksesta [Luettu. 05.07.2013] Saatavissa: <http://marketingland.com/google-survey-what-users-want-from-mobile-sites-22606>

kuva 7.

HTML5 logo, W3C:n kehitys-sivulta, vapaa lähde [Luettu. 05.10.2013] Saatavissa: http://www.w3.org/html/logo/downloads/HTML5_Logo_512.png

kuva 8.

Infografiikka HTML5-verkkosivulta, joka seuraa selainten HTML5 yhteensopivuutta [Luettu. 05.10.2013] Saatavissa: <http://beta.html5test.com/html>

kuva 9.

Infografiikka HTML5-verkkosivulta, joka seuraa selainten HTML5 yhteensopivuutta [Luettu. 05.10.2013] Saatavissa: <http://beta.html5test.com/html>

kuva 10.

Infografiikka HTML5-verkkosivulta, joka seuraa selainten HTML5 yhteensopivuutta [Luettu. 05.10.2013] Saatavissa: <http://beta.html5test.com/html>

kuva 11.

Infografiikka HTML5-verkkosivulta, joka seuraa selainten HTML5 yhteensopivuutta [Luettu. 05.10.2013] Saatavissa: <http://beta.html5test.com/html>

kuva 12.

Infografiikka HTML5-verkkosivulta, joka seuraa selainten HTML5 yhteensopivuutta [Luettu. 05.10.2013] Saatavissa: <http://beta.html5test.com/html>

kuva 13.

Ruudunkaappaus demosivusta Chrome selaimessa mallitarkoituksessa. Sivun on saatavissa ainoastaan lokaalina.

kuva 14.

Ruudunkaappaus demosivusta Chrome selaimessa mallitarkoituksessa. Sivun on saatavissa ainoastaan lokaalina.

kuva 15.

Ruudunkaappaus demosivusta Chrome selaimessa mallitarkoituksessa. Sivun on saatavissa ainoastaan lokaalina.

kuvasarja 1.

ruudunkaappauksia demoverkkosivusta, otettu Chrome selainta hyödyntämällä havainnoimaan demon yleisilmettä

kuvasarja 2.

ruudunkaappauksia demoverkkosivusta, otettu Chrome selainta hyödyntämällä havainnoimaan kommenttien ulkoasun eroja eri laitteilla luettaessa.

kuvasarja 3.

ruudunkaappauksia demoverkkosivusta, otettu Chrome selainta hyödyntämällä havainnoimaan gallerianäkymän eroja eri laitteilla luettaessa.

Demomallin galleriakuvat

(gallerian kuvat on lainattu esittelytarkoitukseen FromUpNorth-verkkosivun kautta, kuvien oikeudet kuuluvat alkuperäisille tekijöille) Kuvia ei käytetä kuvituksena, vaan ainostaan havainnollistamaan sivun tarkoitettua toimintoa.

Kuvat on luettu 15.09.2013

<http://www.cuded.com/2013/03/digital-art-by-nguyen-thanh-nhan/>

<http://www.typeforyou.org/2013/08/14/i-speak-fluid-colors-digital-art-project-by-maria-gronlund-2/>

<http://maandesign.com/hard-rimme/>

<http://www.pinterest.com/pin/146155950380688073/>

<http://www.dailytypespiration.com/?p=723>

<http://alexnegrea.deviantart.com/art/A-Song-Of-Ice-And-Fire-Mother-of-Dragons-fan-art-399013507>

<http://erikjohanssonphoto.com/work/drifting-away/>

<http://www.fromupnorth.com/digital-art-inspiration-809/>

<http://www.dailyinspiration.nl/anubisath-guardian/>

<http://411posters.com/2013/09/soundgarden-copenhagen-2013-by-dan-mumford/>

<http://www.fromupnorth.com/digital-art-inspiration-793/>

<http://cargocollective.com/lucasyoung>

<http://leejeffries.500px.com/>

<http://designerguy.co.uk/fmp.html>

<http://blog.chrisbresler.de/>

<http://instagram.com/mialynne177>

http://adsoftheworld.com/media/print/baby_beef_jardim_costume_party_little_red_riding_hood