

## **HTML5:n uudet ominaisuudet**

Tatu Jauhiainen

Opinnäytetyö

TIKO-05

16.8.2012



<b>Tekijä tai tekijät</b> Tatu Jauhiainen	<b>Ryhmätunnus tai aloitusvuosi</b> 2010
<b>Raportin nimi</b> HTML5:n uudet ominaisuudet	<b>Sivu- ja liitesivumäärä</b> 27+ 8
<b>Opettajat tai ohjaajat</b> Sirpa Marttila	
<p>Tutkimuksen idea lähti omasta tarpeesta selvittää, mitä käyttäjälle merkittäviä uudistuksia HTML5-spesifikaatio tuo tullessaan.</p> <p>HTML5-spesifikaatio ja myöhemmin –standardi tarjoaa huomattavia uudistuksia web-sivujen sisältöön, joista tutkimuksen kohteena olivat Tietovarastot, Geolocation, CSS3, Web-lomakkeet ja Canvas.</p> <p>HTML5 tarjoaa uuden sukupolven merkkaukielen, jonka avulla selain pyörittää sovelluksia, eikä vain lataa toiminnallisuutta ulkopuolisilta palvelimilta.</p> <p>Tutkimuksen suorittamista varten rakennettiin web-sivusto, jolla testattiin näitä ominaisuuksia. Näin päästiin käytännössä kokeilemaan lähempää tarkastelua varten otettuja ohjelmointirajapintoja.</p> <p>Web-lomakkeessa paino laitettiin luonnollisesti uusille ominaisuuksille.</p> <p>Canvas on hyvin tuettu ja sillä pystyy toteuttamaan monipuolisia asioita.</p> <p>CSS3:n kehitys on vielä kesken, mutta selaimet tukevat jo kuitenkin reilusti yli puolta käytössä olevia ominaisuuksia.</p> <p>Tietovarastojen tuki on vielä aika sekavaa ja Websql datan tuki ja ylläpito on lopetettu W3C:n toimesta, silti se on tuettuna joissain selaimissa. Indexed database näyttää olevan monimuotoisempi tietokanta-alusta tulevaisuudessa.</p> <p>Geolocation on jo hyvin tuettu ja tarjosi tämän tutkimuksen mielenkiintoisimman kohteen.</p>	
<b>Asiasanat</b> HTML5, Geolocation, Tietovarastot, CSS3, Canvas, Web-lomakkeet	

<b>Authors</b> Tatu Jauhiainen	<b>Group or year of entry</b> 2010
<b>The title of thesis</b> HTML5, new features	<b>Number of pages and appendices</b> 27 + 8
<b>Supervisor(s)</b> Sirpa Marttila  <p>HTML5 offers next generation markup language which offers a base for modern browsers to run HTML5 applications and not just load services from outside servers. HTML5 specification and later standard will offer remarkable applications for web pages and this research concentrated especially on databases, Geolocation, CSS3, Web forms and the use of Canvas.</p> <p>The purpose of this thesis was to clarify what significant innovations HTML5 specification contains.</p> <p>In order to accomplish the research, web pages were built for testing. This enabled the testing of the chosen APIs in practice. In Web forms, the focus was on new features.</p> <p>The study indicated that Canvas is well supported by modern browsers and it is very versatile.</p> <p>According to the study, the development of CSS3 is still unfinished but browsers support over half of its features which are in use today. The study showed that the support of data storages is still quite varied and Websql database support and maintenance is terminated by W3C. However, some browsers still support it.</p> <p>The study concludes that indexed databases appear to be the form that multiform databases will take in future. Furthermore, Geolocation is well supported and it was the most interesting subject in this research.</p>	
<b>Key words</b> HTML5, Geolocation, Databases, CSS3, Canvas, Web forms	

## Sisällysluettelo

1 Johdanto .....	1
2 HTML5.....	2
2.1 Web-standardit .....	2
2.2 HTML5.....	3
2.3 HTML5:n uudistuksia.....	6
2.4 Tutkittavat ominaisuudet.....	10
2.5 Tietoturva .....	18
2.6 Selaintuki tutkittaville ominaisuuksille .....	18
3 Testaus ja tulokset.....	19
4 Johtopäätökset ja suositukset .....	22
Lähteet.....	26
Liitteet:.....	28
Testisovellus.....	28
Loppuraportti .....	33

# 1 Johdanto

Tämän opinnäytetyön aihe on HTML5:n uudet ominaisuudet. Tarkoituksena on selvittää muutamien keskeisten HTML5:n tarjoamien ominaisuuksien toimintaa, mistä käyttäjät hyötyvät eniten webin käytössä ja käytännössä testata niitä web-ympäristössä. Nämä ominaisuudet ovat Geolocation, joka tarjoaa paikantamispalveluja, tietovarastot, jotka ensi kertaa sijaitsevat käyttäjän laitteella serverin sijaan, Web-lomakkeet, joihin HTML5 tuo uudistuksia ja CSS3 (Cascading Style Sheets), joka on uusin versio ulkoisesta tyylisivustosta sekä Canvas, joka on erittäin monipuolinen esittämisalusta. Tutkimuksen ohessa tehdään testisivu, millä tutkittavia ominaisuuksia testataan käytännössä. Tutkimuksessa tarkastetaan myös käyttäjämäärältään suurimpien selainten tarjoama tämänhetkinen tuki tutkittaville uudistuksille.

Mikäänhän ei estä kehittäjiä jo käyttämästä uutta kuvauskieltä, käytännössä ongelmaksi muodostuu, saavatko websivujen selaajat ominaisuudet hyödynnettyä. Selaimet tarjoavat jo erittäin hyvän tuen näille ominaisuuksille, nyt on web-sivustojen kehittäjien aika alkaa käyttää uutta kuvauskieltä.

Tutkimusalue on rajattu muutamaaan HTML5:n uuteen ominaisuuteen, mitä ei aikaisemmista HTML:n versioista löydy.

HTML5 on vasta spesifikaatiovaiheessa ja se valmistellaan suositusehdotukseksi vuonna 2012. Lopullinen standardi julkaistaan näillä näkymin vuonna 2022. Vauhti on kuitenkin ollut reipasta, joten standardin julkaisu saattanee hyvinkin nähdä päivänvalon jo aikaisemmin.

## 2 HTML5

### 2.1 Web-standardit

Web-standardit ovat web-teknologioiden teknisiä spesifikaatioita. Suurin osa web-standardeista on W3C:n julkaisemia.

Web-standardit ovat usein de facto (käytännössä) standardeja. Mikään laki ei määrää niitä käytettäväksi, joten useat web-kehittäjät eivät ota niitä huomioon. W3C:n suosituksetkaan eivät aina tuo helpotusta tähän.

Marraskuussa 2010 W3C:sta tuli ISO/IEC JTC 1 PAS (Publicly Available Specification) Submitter, mikä tarkoittaa, että W3C:n tuottamat ”any stable core web technologies” on myös täytettävä ISO:n (The International Standards Organization) kriteerit. Tämä muuttaa W3C:n web-standardien statukseksi de jure (lakiin perustuen). (Sikos 2011, 3-4.)

Tällä hetkellä ei kuitenkaan ole yhtään web-standardia tässä käsittelyssä (Marraskuu 2011).

Standardoinnin päämääränä on toiminnallisuus, yhteentoimivuus sekä selain- ja resoluutioriippumattomuus, jotta käyttäjille voidaan taata pääsy webin sisältöihin sekä valikkojen käytettävyyden ja sivustojen ennakoitavissa olevan käyttäytymisen varmistaminen. (Sikos 2011, 4.)

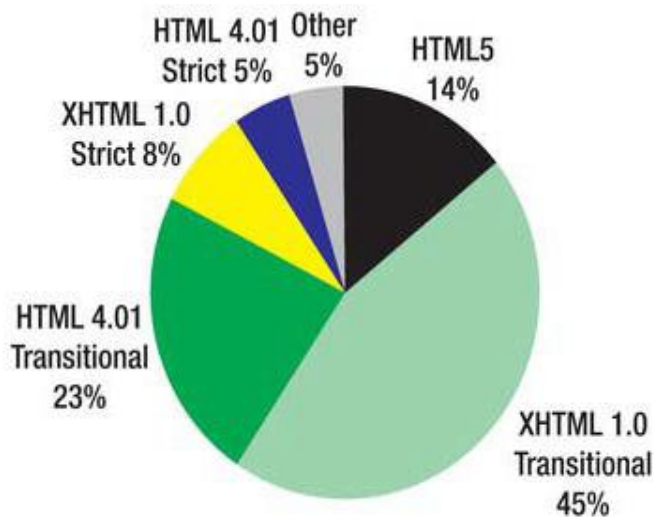
Taulukko 1. Web-standardoinnin vaikutusvaltaiset järjestöt (Sikos 2011, 6)

Organisaatio	Lyhenne	Web-sivut	Tärkeimmät spesifikaatiot ja standardit
Dublin Core Metadata Initiative	DCMI	<a href="http://www.dublincore.org">www.dublincore.org</a>	Dublin Core Metadata
ECMA International (formerly ECMA)	ECMA	<a href="http://www.ecmainternational.org">www.ecmainternational.org</a>	ECMAScript
International Organization for Standardization	ISO	<a href="http://www.iso.org">www.iso.org</a>	Web-sivustojen tekniikka ja muut IT standardit, kuten käyttöliittymästandardit, PNG toiminnallinen spesifikaatio
Internet Assigned Numbers Authority	IANA	<a href="http://www.iana.org">www.iana.org</a>	
Internet Engineering	IETF	<a href="http://www.ietf.org">www.ietf.org</a>	Internet standard (STD)

Task Force			dokumentti , Request for Comments (RFC) dokumentit, esim. HTTP, MIME ja URI:n oikea käyttö
Unicode Consortium	Unicode	www.unicode.org	Unicode Standard, Unicode Technical Reports (UTRs)
Web Hypertext Application Technology Working Group	WHATWG	www.whatwg.org	HTML5, Microdata, Web Applications, Web Forms, Web Workers
World Wide Web Consortium	W3C	www.w3.org	Suosituksia esim. (X)HTML, CSS, DOM, XForms, SVG, RDF, GRDDL, OWL

## 2.2 HTML5

HTML (HyperText Markup Language) on kuvauskieli, jota erityisesti käytetään web-sivujen rakentamiseen. Tällä hetkellä on meneillään version 5.0 kehittäminen, joka aloitettiin jo vuonna 2004. HTML5 spesifikaatio on kirjoitushetkellä luonnosvaiheessa (working draft), 2012 se saavuttaa valmistellaan suositusehdotukseksi-tason (candidate recommendation) ja 2022 on tarkoitus julkaista suositusehdotus (proposed recommendation). (Lubbers, Albers & Salim 2011, 2.)



Kuvio 1. Kuvauskielten osuudet 2011 (Sikos 2011, 21)

HTML sellaisenaan nähtiin riittämättömäksi, joten ryhmä kehittäjiä perusti Web Hypertext Application Working Groupin (WHATWG) vuonna 2004. He loivat HTML5 spesifikaation.

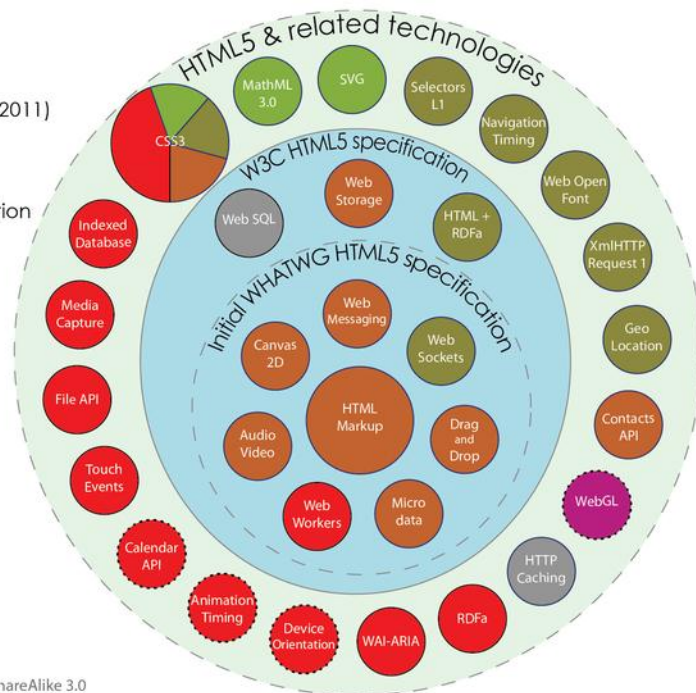
W3C liittyi HTML:n kehitykseen jälleen 2006 ja julkisti ensimmäisen HTML5 luonnoksen vuonna 2008 ja XHTML 2 työryhmä lakkautettiin vuonna 2009.

HTML5 ratkaisee käytännön ongelmia ja selainvalmistajat lisäävät kiitettävästi uusia ominaisuuksia tuotteisiinsa, vaikka HTML5 spesifikaatiota ei vielä olekkaan vahvistettu. Uusien ominaisuuksien käyttö jo näin varhaisessa vaiheessa antaa käyttäjäpalautetta ja näinollen voidaan parantaa HTML5 spesifikaatiota. HTML5 kehittyikin nopeassa tahdissa ja tarjoaa käytännön parannuksia webbiin. (Lubbers ym. 2011, 1.)

# HTML5

Taxonomy & Status (December 2011)

- W3C Recommendation
- Candidate Recommendation
- Last Call
- Working Draft
- Non-W3C Specifications
- Deprecated W3C APIs



By Sergey Mavrody 2011 | CC Attribution-ShareAlike 3.0

Kuvio 2. HTML5 kehittämisen tilanne (Sergey Mavrody, 2011)

Kaikki edeltävät HTML-versiot perustuvat SGML:ään (Standard Generalized Markup Language), mutta HTML5 on ensimmäinen SGML:stä riippumaton versio. (Sikos 2011, 56.)



HTML5-standardi tulee sisältämään n. 40 muuta standardia, näistä tärkeimmät lienevät useat multimediaan liittyvät standardit, CSS3 sekä ECMAScript (paremmin tunnettu nimellä JavaScript), joista osa on jo standardoitu ja osa eri kehitysvaiheissa. HTML5 sisältää uuden syntaksin, uuden doctypen ja merkistökoodauksen.

HTML5 syntaksi noudattaa seuravanlaista kaavaa:

Doctype on yksinkertaisesti muotoa `<!DOCTYPE html>`. Koska HTML5 ei enää periydy SMGL:sta niin se ei tarvitse viittausta DTD:en (Document Type Definition).

Elementit koostuvat seuraavista; Void elements esim. `<area>`, `<met>`, Raw text elements, joka sisältää tagit `<script>` ja `<style>`, RCDATA elements, joka sisältää tagit `<textarea>` ja `<title>`, Foreign elements, joka sisältää MathML ja SVG (Scalable Vector Graphics) nimiavaruuksien elementit ja Normal elements mihin kuuluu kaikki muut sallitut elementit. Void elementseille ei saa laittaa kuin aloitustagin, lopetustagin käyttö on kielletty. Foreign elementsit tarvii lopetustagin tai sitten aloitustagissa kenoviivan käytön. Raw text elementsit, RCDATA elementsit ja Normal elementsit tarvitsevat sekä aloitus- että lopetustagin. Normal elementseissä on tosin poikkeuksia, jotka eivät välttämättä tarvitse lopetustagia. Täydellinen lista poikkeuksista löytyy 3WC:n sivuilta.

Elementin sisältö tulee tagien väliin tai jos lopputagi ei ole pakollinen, niin heti aloitustagin perään. (HTML5 Specification.)

(Vuorio, J 1.3.2011.) Microsoft Oy:ltä sanoo: Merkittävimmät asiat (HTML5:ssa) ovat testipatteristo kaikille selaimille ja tätä kautta pyrkimys standardoituihin selaimiin. HTML5 käsittää 40 eri speksiä, joista tärkeimmistä CSS3 on melkein valmis, JavaScript on valmis, HTML5 itsessään ei ole läheskään valmis, esimerkiksi WebSocketissa on tietoturva-aukkoja. HTML5 tuo yhteiset testausvälineet, joilla selainvalmistajat voivat testata selaimiaan. Kehittäjän kannalta sovellusten kehitystyö tulee olemaan helpompaa.

HTML5 tarjoaa monipuolisempia sovelluksia ja selaaminen nopeutuu. HTML5 tarjoaa alustan universal runtimelle, jossa voidaan jakaa ilmaisia sovelluksia.

Selaaminen helpottuu ja lisäosien määrä vähenee. Päästäänkö lisäosista kokonaan eroon niin luultavasti ei. HTML5 perustuu open source ohjelmille eli niillä ei ole kopiosuojaa. Isojen yhtiöiden maksulliset sovellukset eivät tule jatkossakaan pyörimään HTML5-standardien pohjalla. Esimerkiksi Flash, mediaplayer ja silverlight tulevat säilymään vielä pitkään. Tietokantojen tulo selaimelle parantaa käytettävyyttä huomattavasti ja mahdollistaa offline-tilassa työskentelyn.

Ominaisuuksia tulee ilmanmuuta lisää ennenkuin standardi julkaistaan.

Selaimia voidaan tarkastella ominaisuuksien peruusteella, mutta käyttömieletmykset ja tarkoitus määräävät selainvalinnan. Esimerkiksi IE 6 on vielä käytössä monilla yrityksillä, koska ne eivät vain voi päivittää sitä vanhojen sovellustensa takia.

Flash ja SilverLight eivät katoa myöskään niiden kehitysympäristöjen takia, näillä ympäristöillä on tuhansia käyttäjiä ympäri maailman. HTML5 ja JavaScript ei ole vielä yhtenäinen, erilaisia kirjastoja löytyy paljon. Kunnollista HTML5 kehitysympäristöä ei vielä ole, editointityökalut ovat vasta tulossa.

## 2.3 HTML5:n uudistuksia

**SVG** (Scalable Vector Graphics) on yksi tärkeimmistä standardeista joka tulee sisällymään HTML5 standardiin. Se antaa mahdollisuuden luoda grafiikkaa suoraan web-sivuille kun aikaisemmin muodot on tehty lähinnä Photoshopin avulla ja upotettu jpg- tai gif-formaatissa sivuille. SVG sisältää myös suppeat animaatio-ominaisuudet.

SVG tukee vektorikuvia, rasteri- ja digitaalisia kuvia sekä tekstiä.

SVG säilyttää kuvan tarkkuuden sitä pienennettäessä tai suurennettaessa. SVG-kuva sisältää xml-pohjaiset ohjeet sen piirtämiseen. (Korpela 2011, 314.)

### Listaus 1. SVG esimerkki

```
<svg xmlns="http://www.w3.org/2000/svg">
<circle id="circle"
cx="50" cy="50" r="20"
stroke="#373737" stroke-width="2"
fill="#7d7d7d" />
</svg>
```

Esimerkissä piirretään ympyrä kohtaan  $x=50$ ,  $y=50$ , sen säde on 20 px ja täytetään harmaalla värillä.

**Uusia elementtejä** julkaistaan ja joitakin vanhoja poistuu. Uusista puhutuimmat ovat `<video>`, joka antaa mahdollisuuden katsella videoita HTML5-sivuilla ogg/theora-, H.264- tai WebM- formaatissa (Lawson & Sharp 2011, 98-99.) `<audio>`, jolla äänitiedostoja voi kuunnella ogg/vorbis-, mp3-, wav- ja AAC-formaateissa (Lawson & Sharp 2011, 98-99). Ääni- ja videotiedostojen toistamisessa web-sivuilla on huomattavaa, että ne eivät tarvitse toimiakseen liitännäisiä, kuten Flashplayer, Quickplayer tai Mediaplayer, vain muutamia mainitakseni. `<canvas>` on grafiikka-alusta, josta myöhemmin enemmän luvussa 2.4 Tutkittavat ominaisuudet. Elementeillä `<nav>`, `<header>`, `<article>`, `<section>`, `<aside>` ja `<footer>` rakennetaan HTML5-sivuston ulkoasu, kun ennen yleisenä tapana oli käyttää `<div>` elementtiä kaikkeen sijoitteluun. `<div>` elementin käyttö perustui pitkälti CSS:n kanssa toimimiseen, muuten ei voitu sijoittaa rakennelohkoja CSS:lla sujuvasti, mutta nyt CSS voidaan liittää suoraan rakenne-elementeille. `<embed>` elementillä luodaan 'säiliö' ulkopuoliselle sovellukselle. HTML5 ei tue enää seuraavia elementtejä: `<acronym>`, `<applet>`, `<big>` jne. Täydellinen ja viimeisin lista uusista ja poistuvista elementeistä löytyy W3C:n sivuilta. (HTML5 Specification.)

Kun aikaisemmin selainvalmistajat kehittivät **JavaScriptiä** vain omille selaimilleen niin nyt JavaScript API dokumentoidaan ensi kertaa HTML5:n toimesta, joka tuo sen käyttöön parempaa yhteensopivuutta. (Keith 2011, 21.)

Suuri osa HTML5:n mukana tulevista uusista sovelluksista käyttää JavaScriptiä ohjelmointikielenään.

Käytännössä scripti kirjoitetaan `<script>` `</script>` tagien väliin ja sijoitetaan joko web-sivun `<head>`- tai `<body>`-osioon. Tällöin elementtiin on lisättävä type-attribuutti `<script type="text/javascript">`. Toinen vaihtoehto on käyttää ulkoista javascript tiedostoa. Ulkoisen tiedoston sijainti kerrotaan selaimelle src-attribuutilla

<script src="myJavascript.js"></script>-elementillä. Edellisessä ulkoinen javascript tiedosto sijaitsee samassa kansiossa kuin html tiedostokin.

(Web-julkaisemisen opas, luku 3.)

## Listaus 2. Javascript esimerkki.

```
<!DOCTYPE html>
<html>
<body>
<h1>My First Web Page</h1>
<script type="text/javascript">
document.write("<p>My First JavaScript</p>");
</script>
</body>
</html>
```

**Drag and Drop** APIssa luodaan vähintään kaksi 'kenttää'. Toinen sisältää raahattavat objektit mistä niitä voidaan noukkia ja toinen toimii pudotusalueena mihin raahattavat objektit tiputetaan. Yleensä operaatio alkaa mousedown-toiminnolla ja objekti raahataan pudotusalueelle hiiren nappi alaspainettuna. Hiiren vapauttaminen tiputtaa objektin pudotusalueelle. Tapahtuman kuuntelija (event listener) havaitsee operaation alkamisen ja tapahtumankäsittelijä (event handler) hoitaa toiminnallisuuden. (HTML5 Specification.)

## Listaus 3. Drag and Drop esimerkki, aloituskenttä

```
<p>What fruits do you like?</p>
<ol ondragstart="dragStartHandler(event)">
<li draggable="true" data-value="fruit-apple">Apples</li>
<li draggable="true" data-value="fruit-orange">Oranges</li>
<li draggable="true" data-value="fruit-pear">Pears</li>
</ol>
<script>
var internalDNDType = 'text/x-example'; // set this to something specific to your site
function dragStartHandler(event) {
  if (event.target instanceof HTMLLIElement) {
    // use the element's data-value="" attribute as the value to be moving:
    event.dataTransfer.setData(internalDNDType, event.target.dataset.value);
    event.dataTransfer.effectAllowed = 'move'; // only allow moves
  } else {
    event.preventDefault(); // don't allow selection to be dragged
  }
}
</script>
```

## Listaus 4. Drag and Drop esimerkki, pudotuskenttä

<p>Drop your favorite fruits below:</p>

```

<ol dropzone="move string:text/x-example" ondrop="dropHandler(event)">
  <-- don't forget to change the "text/x-example" type to something
  specific to your site -->
</ol>
<script>
  var internalDNDType = 'text/x-example'; // set this to something specific to your site
  function dropHandler(event) {
    var li = document.createElement('li');
    var data = event.dataTransfer.getData(internalDNDType);
    if (data == 'fruit-apple') {
      li.textContent = 'Apples';
    } else if (data == 'fruit-orange') {
      li.textContent = 'Oranges';
    } else if (data == 'fruit-pear') {
      li.textContent = 'Pears';
    } else {
      li.textContent = 'Unknown Fruit';
    }
    event.target.appendChild(li);
  }
</script>

```

**Web workersit** tarjoavat standardoidun tavan selaimille ajaa JavaScriptia sovellusten taustalla. Web workerseilla voidaan luoda useita ‘säikeitä’, joita ajetaan enemmän tai vähemmän samanaikaisesti. Nämä taustasäikeet voivat laskea monimutkaisia matemaattisia toimituksia, tehdä verkkopyyntöjä tai luoda yhteyden local storageen samalla kun sivusto reagoi käyttäjän rullauksiin, klikkauksiin ja tekstisyytteisiin. (Pilgrim 2010, 23.)

**Offline** staattisilla sivuilla on toiminut jo kauan, riittää kun sivun saa ladattua, niin sitä voi lukea ilman internetyhteyttä vaivatta. Mutta nyt HTML5 tarjoaa mahdollisuuden esim. Gmailille toimia offline-tilassa. Kun avataan sivusto, joka tukee offline ominaisuutta, niin ensin web-palvelin kertoo selaimelle, mitä tiedostoja se tarvii toimiakseen offline-tilassa, tekemällä ns. manifest filen. Nämä voivat käsittää oikeastaan mitä tahansa tiedostoja, JavaScriptia, kuvia tai videoita. Kun selain on ladannut nämä tiedostot, sivusto voidaan renederoida yhteydettömässä tilassa, koska DOM (Document Object Model) kertoo selaimelle, että toimitaan offline-tilassa. Kun Internet-yhteys luodaaan uudestaan, mahdolliset muutokset, jotka on tehty offline-tilassa, voidaan ladata web-palvelimelle. Jotta selain tietää ladata offline-tiedostoja muistiin, pitää sille kertoa siitä HTML-dokumentissa.

## Listaus 5. Manifest attribuutin käyttö

```
<!DOCTYPE HTML>
<html manifest="/cache.manifest">
<body>
...
</body>
</html>
```

Manifest file voi sijaita missä tahansa web-palvelimella.

(Pilgrim 2010, 23, 137-138.)

## 2.4 Tutkittavat ominaisuudet

### Geolocation

Geolocation kertoo käyttäjän laitteen fyysisen paikan kartalla. IP-osoite, laitteen sisäinen GPS, Wi-Fi, MAC-osoite ja GSM ID ovat keinoja paikantamiseen. Myös käyttäjä itse voi määrittää sijaintinsa.

IP-osoiteella saa paikannuksen sen mukaan. Jos esim. IP-osoitteen tarjoaa ulkopuolinen Internet-palveluntarjoaja, paikannus tapahtuu tämän sijainnin mukaan. GPS-paikannuksella laite ottaa yhteyden GPS-satelliittiin. Paikannus on erittäin tarkka, mutta yhteyden muodostaminen kestää usein kauan.

Wi-Fi kertoo laitteen paikan lähimmän langattoman tukiaseman tarkkuudella eli on suhteellisen tarkka paikannusmenetelmä, mutta vain kaupungeissa.

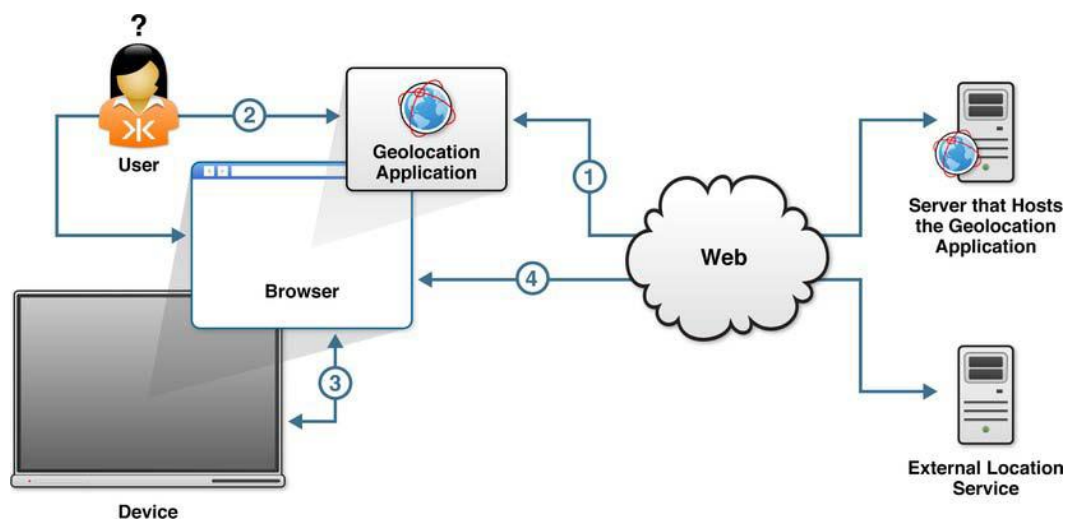
GSM-paikannus tarjoaa paikannuksen muutaman matkapuhelintukiaseman mukaan ja on suhteellisen tarkka ja nopea.

Käyttäjän määrittelemässä paikannuksessa sovellus pyytää osoitetta tai postinumeroa ja tarjoaa paikannuksen sen mukaan. Paikannus on tarkkaa ja nopeaa.

Geolocation API ei määritä mitä teknologiaa paikantamiseen käyttäjän laitteen tulisi käyttää. Se vain paljastaa API:lle noudettavan paikan informaation. Miten tarkka tämä on riippuu täysin paikasta, ei laitteesta josta pyyntö lähetettiin. Laitteen tarkkaa sijaintia ei voida varmuudella kertoa. (Lubbers ym. 2011, 108.)

Geolocationin käytön yksityisyys varmistetaan kysymällä käyttäjältä lupa paikannuksen suorittamiseen. Koko prosessi etenee seuraavasti:

Käyttäjä menee sivustolle, joka käyttää Geolocationia. Sovellus käynnistää Geolocation funktiopyynnön ja pyytää paikkakoordinaatteja selaimelta. Selain pysäyttää tämän ja kysyy käyttäjän hyväksyntää. Luvan saatuaan selain saa koordinaatit käytössä olevan teknologian perusteella. Tämä on selaimen sisäinen toiminto. Selain lähettää koordinaatit hyväksytylle ulkopuoliselle paikannuspalvelulle ja tämän hyväksyessä palauttaa ne, jolloin ne voidaan lähettää sovelluksen Geolocation sovellusserverille.



Kuvio 3. Geolocation yksityisyyskäytäntö (Lubbers ym. 2011, 112.)

## Tietovarastot

Web Storage on uusissa selaimissa jo laajasti käytössä. Web-muisti koostuu kahdesta oliosta, sessionStorage ja localStorage. sessionStorage on tarkoitettu istuntoa varten eli se tyhjentyy kun sivusto tai selain suljetaan, kun taas localStorage on pysyvämpi. LocalStorageen muistiin tallennetaan avain – arvo pareja, joita luodaan ja otetaan käyttöön getItem() ja setItem() metodien avulla. Lisäksi päämetodeihin kuuluvat removeItem(), jolla poistetaan avain – arvo pari ja clear(), jolla koko muisti tyhjennetään. Avaimen ja arvon tiedostomuoto on merkkijono (string). (Korpela 2011, 232-233.)

Tiedon varastointi tapahtuu selaimella eli näinollen paljon parjatut keksit (cookies) poistuvat käytöstä, kuinka nopeasti ja päästäänkö niistä kokonaan eroon on

mahdoton sanoa tässä vaiheessa. HTML5 sallii tiedon tallentamisen lokaalisti samaan tapaan kuin evästeillä. Erona on se, että nyt tietoa voi tallentaa huomattavasti enemmän ja käyttömahdollisuudet ovat monipuolisemmat. (TeroAuralinnablogi.)

Toisin kuin cookieissa selain ei palauta tietoa palvelimelle, ellei tätä tehdä manuaalisesti. Local storage on natiivisti selaimella, joten se on käytettävissä vaikka kolmannen osapuolen tarjoamat selaimen lisäosat eivät olisikaan. Viiden MB:n tallennustila selaimilla on yllättävän hyvin tuettu, vaikka se on vain suositus HTML5:n Storage spesifikaatiossa. Jos local storage täyttyy, tällä hetkellä ei kehittäjillä ole mahdollisuutta lisätä sitä selaimilla. Opera tarjoaa käyttäjälle mahdollisuuden hallita eri sivustojen käyttämää dataa, mutta se on täysin vain käyttäjän hallittavissa eli kehittäjä ei voi siihen mitenkään vaikuttaa. (Pilgrim 2010, 129, 132.)

HTML5 mahdollistaa web-sovelluksille sovellusvälimuistin (application cache). on tietovarasto sovelluksen käyttöä varten esim. tilanteessa kun Internet-yhteys katkeaa ennalta arvaamattomasti. Tätä ei pidä sekoittaa offline-tilaan, joka tarkoittaa selaimen omaan välimuistiin tallennetun datan käyttöä yhteydettömässä tilassa. (Korpela 2011, 268-270.)

Selainten täytyy pystyä varastoimaan suuria määriä tietoa pystyäkseen tarjoamaan kunnolliset offline ominaisuudet web-sovelluksille. Indexed Database on vastaus tähän tarpeeseen. Sillä pystytään varastoimaan dataa selaimeen ja suorittamaan indeksoituja tai avainperustaisia hakuja tietokannasta. Indeksit luodaan käyttäen B-puu (B-tree) data rakennetta. (Indexed Database API.)

Websql database ei ole enää tuettu W3C:n toimesta, joten sitä ei tässä tutkimuksessa käsitellä.



## Web-lomakkeet

HTML5 tarjoaa web-lomakkeille uusia input-elementin type-määrittelyn arvoja. Jo käytössä olevien lisäksi tulee mm. color, date, url jne. Jotkut uusista arvoista kuten type="url" toimivat myös vanhoissa selaimissa. (Korpela 2011, 144.)

Placeholder Text asettaa syöttökenttään vihjetekstin, mitä sillä voidaan tehdä, esim. 'Etsi sivustolta'. Kun focus siirretään kyseiseen kenttään, teksti katoaa.

Javascriptilla koodattu autofocus saattaa aiheuttaa ongelmia sivuston rullauksen kanssa, kun focus onkin jo jossakin input-kentässä, mitä käyttäjä ei havaitse tai jos sivuston lataus on kesken kun käyttäjä valitsee toisen kentän mihin kirjoittaa, niin latauksen valmistuttua scripti siirtää fokuksen takaisin haluaamansa input-kenttään. HTML5:n autofocus attribuutti taas on merkkikieltä (markup) eikä scriptiä, joten sen toiminta on yhdenmukaista kaikissa sivustoissa.

Email on tarkoitettu sähköpostiosoitteelle. Sitä voidaan käyttää kaikissa kaavakkeissa vaikka selain ei sitä tukisikaan. Selain tällöin käsittelee sen muodossa type='text'.

Url on tarkoitettu Internet osoitteille ja tässäkin selain, joka ei tue uutta tyyppiä käsittelee sen muodossa type='text'.

Numberiin laitetaan numeroarvoja ja attribuutit helpottavat sen käyttöä kehittäjän näkökulmasta. Min, max, step ja value määrittävät mitä kenttään voi syöttää.

Numerosyötön voi myös esittää liukusäätimeinä.

HTML5 tarjoaa natiivin Date picker kontrollin ilman skripteja. Niitä on kuusi erilaista aikamäärettä, date, month, week, time, date + time ja date + time – timezone.

Search on tarkoitettu hakujen tekoon kun aikaisemmin nämäkin olivat input type='text'-muotoa.

Color tarjoaa perinteisen värivalitsin-kentän natiivina.

Ikävä kyllä selainten tarjoama tuki näille uusille tyypeille on hyvin vaihtelevaa ja vajavaista. Parhaiten uudistuksia tällä hetkellä tukee Opera. (Pilgrim 2010, 23, 147-160.)

Uudistuksen tarkoituksena on saada yksi tieto yhteen kenttään. Esim. nykyään syntymäaikaa kysytään yleensä kolmella eri kentällä, päivä, kk, vuosi kun `type="date"` määritteellä voidaan kirjoittaa yhteen kenttään muodossa "1.1.2012". (Korpela 2011, 153.)

HTML5 tarjoaa sovellusliittymän, Constraint validation API, joka on alkeellinen lomakkeiden syötteiden virheenkäsittelyyn tarkoitettu sovellus. Kuitenkin edelleen on parempi tehdä virheenkäsittely JavaScriptillä. (Korpela 2011, 170.)

### CSS3

Cascading Style Sheets (CSS) on 'tietokonekieli', jolla määritetään rakenteisten dokumenttien semantiikka. (Sikos 2011, 161.)

CSS3 ei ole HTML5 riippuvainen vaan sitä voidaan käyttää minkä tahansa HTML-version kanssa.

CSS3:n kehitys aloitettiin vuonna 2005. Suurimpana muutoksena aikaisempiin versioihin nähden on, että CSS3 on modulisoitu. Tämä tarkoittaa, että erilliset kokonaisuudet on jaettu moduuleihin. (Sikos 2011, 163.) Muita muutoksia ovat pyöreät kulmat, varjostus ja kuvista muodostetut rajat, uusia valitsimia (puu-rakenne), esitystasot, SVG yhdistyy tiukemmin CSS:ään, matemaattisten merkintöjen muotoilu, lisäominaisuuksia tekstin muotoilemiseen, nimiavaruuslaajennukset, lisää ominaisuuksia kirjasinten ulkoasun muotoiluun, ääniefektit ja speech-mediatyyppi uusitaan. (Opetusteknologia | Luennot.)

Taulukko 2. CSS3:n moduulit (Web-sivujen tekeminen ja Web muutoinkin.)

moduli	Modulin tila	Modulin sisältö
CSS Syntax Level 3	pahoin vanhentunut (2003)	yleiset muutosäännöt
CSS Values and Units Level 3	viimeisteltävä työversio (LC)	arvot ja mittayksiköt
CSS Image Values and Replaced Content Level 3	valmistellaan ehdotusluonnokseksi (CR 2012)	kuviin viittaaminen URLeilla
CSS Style Attributes	valmistellaan suositusehdotukseksi	CSS-sääntöjen esittäminen style-määritteissä
CSS Namespaces	suositus (REC)	nimiavaruuksien käyttö CSS:ssä
Selectors Level 3	suositus (REC)	selektorit (joilla valitaan, mitä elementtejä sääntö koskee)
Selectors Level 4	työversio	runsaasti laajennuksia,

moduli	Modulin tila	Modulin sisältö
		esim. :nth-child()
CSS Cascading and Inheritance Level 3	epäaktiivinen (2005)	kaskadi- ja periytyvyysäännöt
Media Queries	suositus (REC)	mediakyselyt, @media {...}
CSS Conditional Rules Level 3	työversio	laajennuksia ehdollisuuteen, mm. selaimen ominaisuuksien ja sivun URLin mukaan
CSS Basic Box Model Level 3	pahoin vanhentunut (2007)	laatikkomalli, täyte, reunaviiva ja reunus
CSS Line Layout Level 3	pahoin vanhentunut (2002)	tekstirivien muotoilu
CSS Positioned Layout Level 3	työversio	asemointi, leveydet ja korkeudet
CSS Flexible Box Layout	viimeisteltävä työversio (LC)	joustavat laatikot, joissa sisällön sijoittelua voidaan ohjailla
CSS Multi-column Layout	valmistellaan suositusehdotukseksi (CR)	tekstin juokseva palstoitus
CSS Regions	työversio	muotoilu, jossa käytetään muitakin kuin suorakulmaisia alueita
CSS Transitions	työversio	ajastetut muutokset, joissa esim. väri vaihtuu toiseksi
CSS Animations	työversio	animaatiot
Compositing and Blending	editorin luonnos	värien ja elementtien sekoittaminen
Filter Effects	editorin luonnos	suodattimien asettaminen
CSS Generated Content Level 3	pahoin vanhentunut (2003)	generoituva sisältö, laskurit ym.
CSS Lists and Counters Level 3	työversio (2011)	luettelmat ja laskurit
CSS Paged Media Level 3	epäaktiivinen (2006)	sivunvaihtojen ohjaus yms.
CSS Fragmentation Level 3	työversio	sivunvaihdon käsitteen yleistys
CSS Generated Content for Paged Media	työversio	sivujen otsakkeet, alaviitteet yms.
CSS Color Level 3	suositus (REC)	värit, myös läpinäkyvyys
CSS Backgrounds and Borders Level 3	valmistellaan suositusehdotukseksi (CR)	taustat ja reunaviivat
CSS Fonts Level 3	työversio	fonttiasetukset, myös latautuvat fontit
CSS Text Level 3	työversio	Mm. rivinvaihdot, välilyönnit, tavutus, tasaus
CSS Marquee	valmistellaan suositusehdotukseksi (CR, 2008)	rullaava teksti
CSS Writing Modes Level 3	työversio	vaaka- ja pystysuora kirjoitussuunta

<b>moduli</b>	<b>Modulin tila</b>	<b>Modulin sisältö</b>
CSS Ruby	työversio (2011), vanhentunut	ruby-esitys (selitystekstit, lähinnä itäaasialaisille kielille)
CSS Line Grid	editorin luonnos	rivirekisteri
Tables Level 3	vanhentunut	taulukot
CSS Grid Layout	työversio	hilat, joilla voidaan määritellä taulukkomainen taitto
CSS Template Layout	työversio	taittomallit
CSS Basic User Interface Level 3	työversio	vuorovaikutteisuuden liittyviä ominaisuuksia
CSS Object Model		sovellusliittymiä (API), joilla käsitellään mediakyselyjä, selektoreita ja CSS:ää muutoin
CSSOM View	työversio	sovellusliittymiä, joilla käsitellään dokumentin esityksen ominaisuuksia
CSS Speech	valmistellaan suositusehdotukseksi (CR)	ääniesitys
CSS Device Adaptation	työversio	esitysnäkymän (viewport) perusominaisuudet
CSS Transforms	työversio	siirrot, kierrot ym.
CSS Exclusions and Shapes	työversio	tekstin juoksutus erimuotoisten alueiden ohi
CSS Presentation Levels	epäaktiivinen (2003)	sisällön vaiheittainen esittäminen
Behavioral Extensions to CSS	hylätty, mutta käsitteistöä käytetään HTML5-luonnoksissa	elementtien toiminnallisuuden muuttaminen

W3C:n käyttämät termit määrittelyjen tilalle:

WD Working Draft	työversio; voi olla epävakaa ja sisältää aukkoja
LC Last Call	viimeisteltävä työversio
CR Candidate Recommendation	valmistellaan suositusehdotukseksi
PR Proposed Recommendation	suositusehdotus
REC Recommendation	suositus, ”W3C:n standardi”

Työversiota edeltävä taso on editorin luonnos (Editor’s Draft).

---

Julkaistu 6.6.2012. Muokattu viimeksi 26.6.2012.

## Canvas

Canvas on alunperin Applen kehittämä alusta esittämään grafiikkaa ja aluksi sitä tuki vain Safari-selain. Myöhemmin kaikki muut suuret selainvalmistajat ottivat sen myös käyttöön. Canvaksen mahdollistaa grafiikan piirtämisen HTML5-pohjaisille sivuille Javascriptin avulla. Canvas vaatii tunnistus-, pituus- ja leveys-attribuutit (id, height, width) toimiakseen. (Rowell 2011, 8-9.)

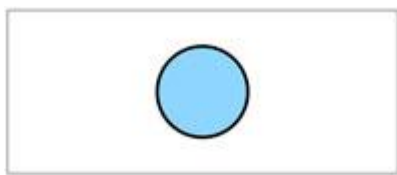
Elementti `<canvas>` on säiliö 2D-tiedostoille, joita web-sivuilla halutaan esittää, piirroksia, kuvat, pelit, taustakuvat, animaatiot jne. nämä kaikki voidaan laittaa `<canvas>`-elementin sisään. (Lawson & Sharp 2011, 115-117.)

### Listaus 6. Yksinkertainen Canvas esimerkki

```
window.onload = function(){
var canvas = document.getElementById("myCanvas");
var context = canvas.getContext("2d");

context.arc(canvas.width / 2, canvas.height / 2, 70, 0, 2 *
Math.PI, false);
context.fillStyle = "#8ED6FF";
context.fill();
context.lineWidth = 5;
context.strokeStyle = "black";
context.stroke();
};
```

Esimerkki koodi piirtää html5-dokumentin body-elementissä olevaan canvas-elementtiin ympyrän määritettyine viivanpaksuuksineen, väreineen ja täytteineen.



Kuvio 4. Canvas esimerkki

Canvaksen eri toimintamahdollisuuksien hyödyntämisessä on eräs huono puoli ajatellen sitä HTML5:en ideaan tuoda toiminnallisuutta mahdollisimman paljon asiakaspuolelle, esim. kuvien manipuloinnissa koodaus joka sisältää `getImageData()`-metodin vaatii turvallisuusrajoitusten vuoksi server-ympäristön. (Rowell 2011, 90.)

## 2.5 Tietoturva

Kehitteillä oleva uusi html5-standardi tarjoaa uusien ominaisuuksiensa lisäksi myös uudenlaisia tietoturva-uhkia. Nettilain alkaa sen myötä käsitellä pelkkien verkkosivujen sijasta kokonaisia sovelluksia.

Hyökkääjä voi saada urkittua käyttäjän kotikoneelle varastoituja tietoja tai lisätä niiden joukkoon huomaamattomasti jotakin. Paikkatiedon avulla onnistuu puolestaan sijainnin selvittäminen. Css-tyylitiedostojen uusi versio tarjoaa hyökkääjälle mahdollisuuden hallita, mitä elementtejä sivulla on näkyvissä.

Html5:n websocket-teknologia antaa selaimelle verkkotietoliikenteen pinon, jota hyökkääjä voi käyttää salaisen backdoor-verkkoyhteyden käyttämiseen.

Selainvalmistajat eivät kuitenkaan ole tietämättömiä uhkista. Uusien ominaisuuksien lisäämisen myötä pyritään samalla kehittämään keinoja estää niiden väärinkäyttöä. (TIVI.FI.)

## 2.6 Selaintuki tutkittaville ominaisuuksille

Testiselaimiksi valittiin IE9, Mozilla Firefox 14 ja Opera 12. Tällä hetkellä kaikki selaimet läpäisivät acid3-testin, joka on selainten renderointi- ja nopeustesti sekä lisäksi sillä testataan seuraavia osa-alueita:

- DOM2 Core
- DOM2 Events
- DOM2 HTML
- DOM2 Range
- DOM2 Style (getComputedStyle, ...)
- DOM2 Traversal (NodeIterator, TreeWalker)
- DOM2 Views (defaultView)
- ECMAScript
- HTML4 (<object>, <iframe>, ...)
- HTTP (Content-Type, 404, ...)
- Media Queries
- Selectors (:lang, :nth-child(), combinators, dynamic changes, ...)
- XHTML 1.0
- CSS2 (@font-face)

- CSS2.1 ('inline-block', 'pre-wrap', parsing...)
- CSS3 Color (rgba(), hsla(), ...)
- CSS3 UI ('cursor')
- data: URIs
- SVG (SVG Animation, SVG Fonts, ...)

(The Web Standards Project.)

Taulukko 3. HTML5 tuetut ominaisuudet (When can I use....)

	Opera	Firefox	IE
Local storage	Kyllä	Kyllä	Kyllä
Session storage	Kyllä	Kyllä	Kyllä
Database storage	Kyllä	Ei	Ei
WebSQL DB	Kyllä	Ei	Ei
IndexedDB	Ei	Kyllä	Ei
Web - lomakkeet	Kyllä	Kyllä	Kyllä
CSS3	Osittain	Osittain	Osittain
Geolocation	Kyllä	Kyllä	Kyllä
Canvas	Kyllä	Kyllä	Kyllä

### 3 Testaus ja tulokset

Tutkittavina kohteina ovat Canvas, Tietokannat, Geolocation ja web-lomakkeet.

CSS oli tutkimuksessa mukana testisivuston tyylisivustona.

Testauksessa käytettiin Windows7 -käyttöjärjestelmää. Testisivusto validoidaan ennen testausta W3C:n validointisivustolla <http://validator.w3.org>. Tutkimus suoritetaan vasta kun validointi on suoritettu hyväksytysti.

Ennen testisivun koodauksen aloittamista testattiin selainten tarjoama tuki testattaville ominaisuuksille. Testiselaimina toimivat Mozilla FireFox 13.0.1, Internet Explorer 9.0.7 ja Opera 11.62. Koska testiselaimet läpäisivät jo aikaisemmin acid3-

testin voidaan todeta, että riittävä CSS3 tuki on jo olemassa ja tätä ei erikseen enää testata.

Testiselaimista ainoastaan Opera tuki kaikkia kolmea tietovarastotyyppiä; Session Storage, Local Storage ja Database Storage. Tulos vastaa teoriapohjan tietoja.

#### Listaus 7, selaimen tietokantatuken testaus

```
<script language="javascript">
function RunTest() {
for (var mydata in window)
{
switch (mydata) {
case "sessionStorage":
document.getElementById("session").innerHTML = "supported";
break;
case "localStorage":
document.getElementById("local").innerHTML = "supported";
break;
case "openDatabase":
document.getElementById("db").innerHTML = "supported";
break;
} }}
</script>
```

Muut selaimet kuin Opera eivät läpäisseet Database Storagen mahdollisuutta.

Local Storagea testattiin yksinkertaisella JavaScript sovelluksella, joka input-ruudusta annetulla syötteellä tallentaa syötteen lokaaliin tietovarastoon. Sovellukseen liitettiin funktio, jolla muistin voi tyhjentää. Teastaus toimi kaikilla selaimilla samalla tavalla. Muistiin laitettut avainparit tallentuivat ja ne voitiin poistaa kuten pitikin. Session storagea ei testattu, koska se on palvelinpuolen ominaisuus. Samoin Database Storage on palvelinpuolella ja tällä hetkellä vain Operan tukema.

Kaikki kolme testiselainta läpäisivät Geolocationin tuen testin. Tulos vastaa teoriapohjan tietoja.

#### Listaus 8. Selaimen Geolocationin tuen testaus

```
<script language="javascript">
function RunTest() {
if(navigator.geolocation) {
document.getElementById("support").innerHTML = "Geolocation supported.";
} else {
```



```
document.getElementById("support").innerHTML = "Geolocation is not supported in  
your browser."  
}}  
</script>
```

Geolocation-sovellus rakennettiin lisäämällä ensin funktio, joka ilmoittaa asiakaskoneen paikan leveys- ja pituusasteiden mukaan. Kovakoodaamalla lisättiin lopputyöntekijän olinpaikan koordinaatit koodiin. Lisättiin funktio, joka laskee olinpaikan ja asiakaskoneen välisen etäisyyden. Tämän jälkeen lisättiin vielä Google Maps sovellus graafiseksi pohjaksi.

Sovellus näytti pöytäkoneen olinpaikaksi Internet-yhteyden tarjoajan DHCP-serverin sijoituspaikan niinkuin pitääkin kaikilla selaimilla. Tosin IE:n paikannus heitti kahdesta muusta selaimesta poiketen muutamalla sadalla metrillä.

Langattomassa verkossa testi antoi tulokseksi hyvinkin tarkan sijaintipaikan kaikilla selaimilla, heittoa ei todellisen sijainnin kanssa ollut ollenkaan tai enintäänkin muutamia metrejä.

Myös Canvas-testin läpäisi kaikki kolme testiselainta. Tulos vastaa teoriapohjan tietoja.

Lista 9, selaimen Canvas tuen testaus.

```
<script language="javascript">  
function RunTest() {  
var test_canvas = document.createElement("canvas")  
var canvascheck=(test_canvas.getContext)? true : false  
alert(canvascheck)  
}  
</script>
```

Canvaksen testaukseen tehtiin neljä eri kokeilua, ensimmäinen piirtää geometrisiä kuvioita, tässä tapauksessa kolmioita, eri värityksillä. Toinen piirtää satunnaisia kuvioita (kukkia) eri paikkoihin canvakselle hyväksikäyttäen random()-funktioita. Kolmas testi piirtää pata-logon keskelle canvasta ja kopioi logosta puolikkaat molemmille puolille alkuperäistä logoa. Kaikki piirrostestit toteutettiin SVG-tekniikalla ja JavaScript skriptauksella. Neljäs testi on mp4-video esitettynä canvaksen sisällä.

Kaikki selaimet toimivat grafiikkatesteissä identtisesti, mutta ainoastaan IE suostui esittämään videon.

Web Formsien testaus suoritettiin kaavakkeella, johon laitettiin HTML5:n mukanaan tuomia uudistuksia. Kokeiltiin Constraint validation API:n toimintaa, placeholder-attribuuttia, number ja date input typeja sekä Regular Expressionsia. Tässä hajonta oli suurinta selainten kesken ja ainoastaan Opera toimi niin kuin piti. Firefoxilla ja IE:llä on vielä paljon tehtävää Formsien suhteen. Tulos ei vastannut kuin Operan osalta teoriapohjan tietoja.

CSS3:a ei siis testattu muuten kuin, että se on osana testisivustoa ulkoisena tiedostona html5.css. Kaikissa selaimissa ulkoasu näytti silmälle yhtenäiseltä.

## **4 Johtopäätökset ja suositukset**

Tutkimuksen tarkoituksena oli selvittää, mitä HTML5:n tarjoamat uudet ominaisuudet ovat ja miten toimivat tällä hetkellä. Lopputyötä tehtäessä kuva HTML5:n tilasta selkeni. Sitä rummutetaan jo uutena versiona, mutta todellisuudessa se on pahasti keskeneräinen. HTML5 formseja ja isoja tietokantoja en rupeaisi tekemään mihinkään projektiin ennenkuin selaintuki näille ominaisuuksille varmistuu ja vahvistuu. Tutkimuksessa oli otoksena vain kolme selainta ja niidenkin välillä oli senverran hajontaa, joten voidaan todeta, että selainmarkkinat on edelleen hyvin hajallaan. Tähän tietenkin vaikuttaa myös se, että selainvalmistajat eivät tiedä mitä tarkalleen lopulliseen standardiin sisältyy. Jo nyt W3C on lopettanut joidenkin osioiden tukemisen ja kehityksen, mitkä olivat mukana aikaisemmassa vaiheessa.

Kun HTML5 saadaan toimimaan kaikilla osa-alueilla tulevat web-sovellukset valtaamaan osuuksia työpöytäsovelluksilta. Tällöin esim. perinteinen Office-paketti tuskin sijaitsee koneen kovalevyllä, vaan selain tarjoaa sen tiettyä käyttömaksua vastaan. Tämä varmaan tulee vähentämään myös piratismia, joka nyt on iso

ongelma. Tähän Microsoft on jo pitkään pyrkinyt ja HTML5 tuo siihen mahdollisuuden. Ihmetyttää ainoastaan IE:n tuki HTML5:een, kun yhtiöllä olisi selkeä business agenda tällä alueella.

Lomakkeiden uudistusten tarkoituksena lienee helpottaa lomakkeiden tekoa ja toimintaa. Tutkimustulosten perusteella kuitenkin tutkimuksessa mukana ollut käyttäjämäärältään pienin selain Opera, ainoana vastaa lomakkeiden uudistuksiin tällä hetkellä. Ennenkuin selaimet vastaavat HTML5-spesifikaatioon lomakkeiden osalta, niiden tekeminen lienee turhaa, koska 98% Internetin käyttäjistä käyttää jotain muuta selainta kuin Operaa, joka ainoana tällä hetkellä tukee siedettävästi HTML5 web formseja graafisesti.

Tietovarastot tuovat kipeästi tarvittavaa säilytystilaa sovelluksille ja ennenpitkää 'keksit' tulevat häviämään kokonaan. Niiden neljän kilobitin tallennuskapasiteetti on auttamattomasti liian pieni nykyisten sovellusten tiedonsiirtoon ja näin hidastavat sivustojen toimivuutta. Local storage ja Session storage tulevat olemaan huomattavasti parempia kuin edeltäjänsä. Suurempaa tietokantaa vaativat sivustot ja sovellukset tulevat tarvitsemaan yhden kaikkien selainvalmistajien tuen tietokannalle ja tällä hetkellä se näyttää olevan Indexed Database. Tässä tuessa on vielä ikävä kyllä valitettavaa hajontaa, joten saamme nähdä edelleen perinteisiä sovelluksia muutaman vuoden ennenkuin tietokannat toden teolla siirtyy selaimelle.

Geolocation on ilman muuta yksi mielenkiintoisimmista ohjelmointirajapinnoista, jonka HTML5 tuo tullessaan. Jo nyt monet kantavat 'taskussaan' laitetta, jolla tätä voidaan hyödyntää. Tekniikka on valmis tehdä mitä tahansa sovelluksia, jossa paikannus helpottaa asiakaspuolta. Facebook, Twitter yms. sovellukset tarjoavat käyttäjille mahdollisuuden kertoa mitä minä teen, tein tai tulen tekemään.

Geolocation tarjoaa mahdollisuuden kertoa missä teen tai olen. Tulevaisuus näyttää haluavatko käyttäjät jakaa tämän tiedon.

Kaupunkien keskustoissa toteutetut ilmaiset langattomat verkot eivät ole olleet oletetun suosittuja ja syy tähän on kännyköiden Internet-yhteydet. On helpompaa

kantaa 50 gramman puhelinta kuin kannettavaa ja yhteydenotto ei vaadi erillisiä 'tikkuja' ja yhteydenmuodostuksia. Ilmainen langaton verkko koko kaupungin alueelle olisi seuraava askel, jolla Geolocation saataisiin riittävän tarkaksi kaikille sitä käyttäville. Mediamaksu olisi yksi mahdollisuus rahoittaa koko kaupungin kattava verkko.

HTML5:n kiinnostavin ominaisuus on sen tarjoamat ohjelmointirajapinnat, mitkä saattavat tehdä selainten lisäosat tarpeettomiksi ja varmasti tulevat tekemään näin osalle niistä. Tähän kuitenkin päästäneen vasta vuosien kuluttua ja muutamien lisäosien häviämien näyttää epätodennäköiseltä. Ainakin esimerkiksi Adobe Flash ja Microsoft SilverLight tulevat pitämään osuuksistaan kiinni loppuun asti. Toki kyseisillä toimijoilla on huomattavasti kokemusta alastaan ja parempi tietoturva kuin HTML5 pystyy vielä tarjoamaan. Tosin Google on jo luopunut jostain lisäosistaan todettuaan, että HTML5 tekee ne tarpeettomaksi. Google lieneekin suurista IT-yhtiöistä eniten HTML5:n takana kokonaisuudessaan. Googlen suuri vika on, että se 'nuuskii' käyttäjiään ja tekee omat sovelluksensa vaikeasti muokattaviksi. Siksi Google Chrome-selain ei kuulunut tämän tutkimuksen piiriin.

Yksi tärkeä asia mihin tämä tutkimus ei ole ottanut juurikaan kantaa, eikä siitä löydy vielä suuremmin aineistoakaan, on tietoturvallisuus. HTML5 on askel avoimemman Internetin suuntaan ja se asettaa suuret vaatimukset tietoturvan suhteen.

Lähitulevaisuudessa tulemme varmuudella näkemään keskustelun siirtyvän enemmän ja enemmän myös tälle alueelle. Toisaalta jos toiminnallisuuksia siirretään takaisin serverien hoidettavaksi tietoturvallisuussyistä, HTML5:en perusajatus kärsii siitä.

Selainpuolella Operalla tuntuu olevan pieni etumatka kilpailijoihinsa nähden. Asiaa jonkun verran seuranneena voi todeta, että se saattaa muuttua jo lähitulevaisuudessa kun tulee selainten seuraavat versiot. Versioita e tosin pidä sekoittaa päivityksiin, jotka vaan lähinnä korjaavat bugeja.

Kaiken kaikkiaan HTML5 on tervetullut standardisoimaan Internetissä nykyään vallitsevat villit markkinat. Jos mihinkin tarpeeseen tarvii nykyään asentaa oma

lisäosansa, niin tulevaisuudessa uskon pystyväni saamaan tarvitsemani palvelut käyttöön ilman erillisiä asennuksia selaimen ja HTML5:n yhteistyön ansiosta.

## Lähteet

HTML5 Specification.

Luettavissa:<http://dev.w3.org/html5/spec/single-page.html#introduction>. Luettu 7.8.2012.

Indexed Database API.

Luettavissa:<http://www.w3.org/TR/IndexedDB/>. Luettu 24.7.2012.

Lubbers, P. Albers, B. Salim F. 2011. Pro HTML5 Programming, Second Edition. Apress. New York.

Opetusteknologia | Luennot

Luettavissa:<http://appro.mit.jyu.fi/opetusteknologia/luennot/luento4/#TOC9>.

Luettu 11.7.2012

Pilgrim, M. 2010. HTML5 up and running. O'Reilly. Sebastopol.

Rowell, E. 2011. HTML5 Canvas Cookbook. Packt Publishing. Birmingham.

Sikos, L F. 2011. Web Standards, Mastering HTML5, CSS3, and XML. Apress. New York.

TeroAuralinnablogi.

Luettavissa:<http://www.auralinna.fi/2009/11/15/html5/>. Luettu 13.5.2012.

The Web Standards Project.

Luettavissa: <http://www.webstandards.org/action/acid3/>. Luettu 4.8.2012.

TIVI.FI

Luettavissa:[http://www.tietoviikko.fi/kaikki\\_uutiset/html5+ndash+ei+vain+parempia+verkkosivuja+vaan+myos+pelottavia+turvauhkia/a490862](http://www.tietoviikko.fi/kaikki_uutiset/html5+ndash+ei+vain+parempia+verkkosivuja+vaan+myos+pelottavia+turvauhkia/a490862). Luettu 24.6.2012

Web-julkaisemisen opas, luku 3

Luettavissa: <http://www.cs.tut.fi/~jkorpela/webjulk/3.2.html>. Luettu 27.7.2012

Web-sivujen tekeminen ja Web muutoinkin.

Luettavissa:<http://www.cs.tut.fi/~jkorpela/css/modulit.html>. Luettu 10.7.2012

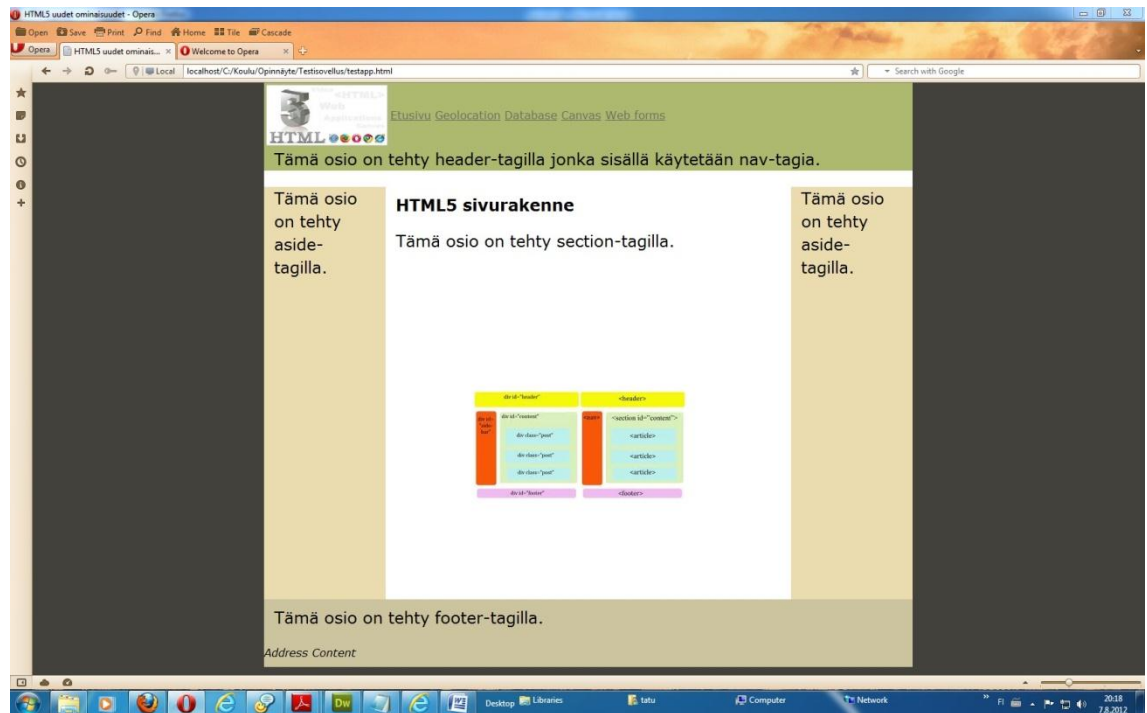
When can I use...

Luettavissa:<http://caniuse.com>. Luettu 1.8.2012

## Liitteet:

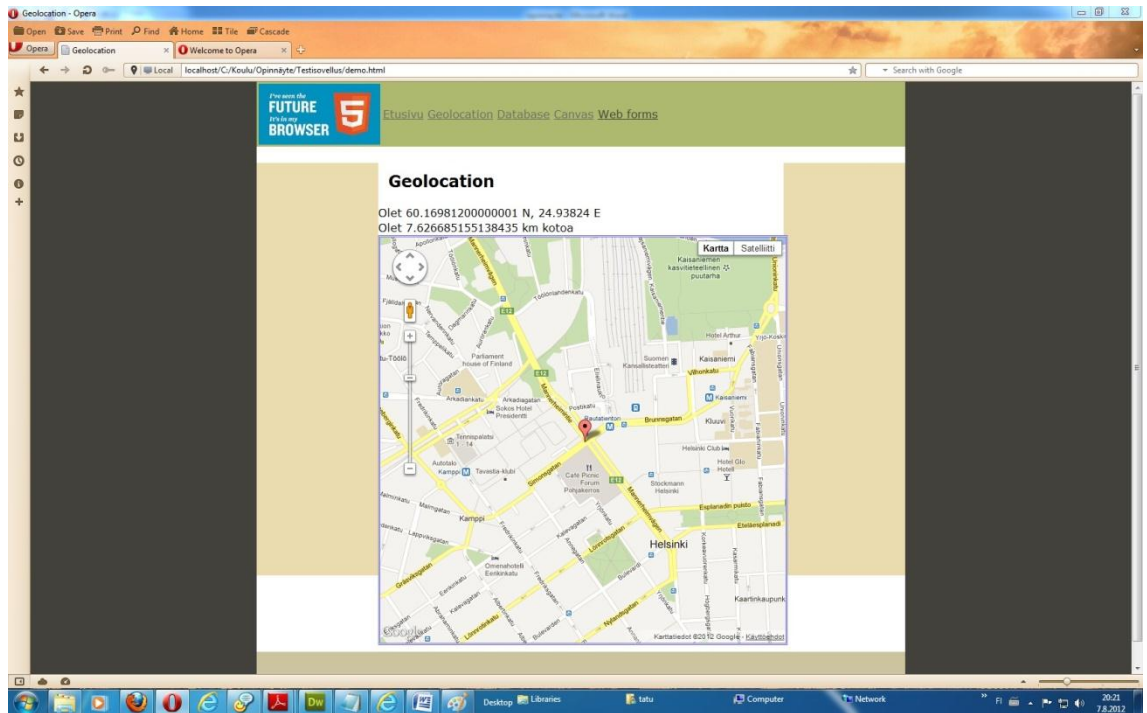
## Testisovellus

Testisovelluksen etusivu esittelee lähinnä HTML5:n uutta tapaa rakentaa sivustoja.

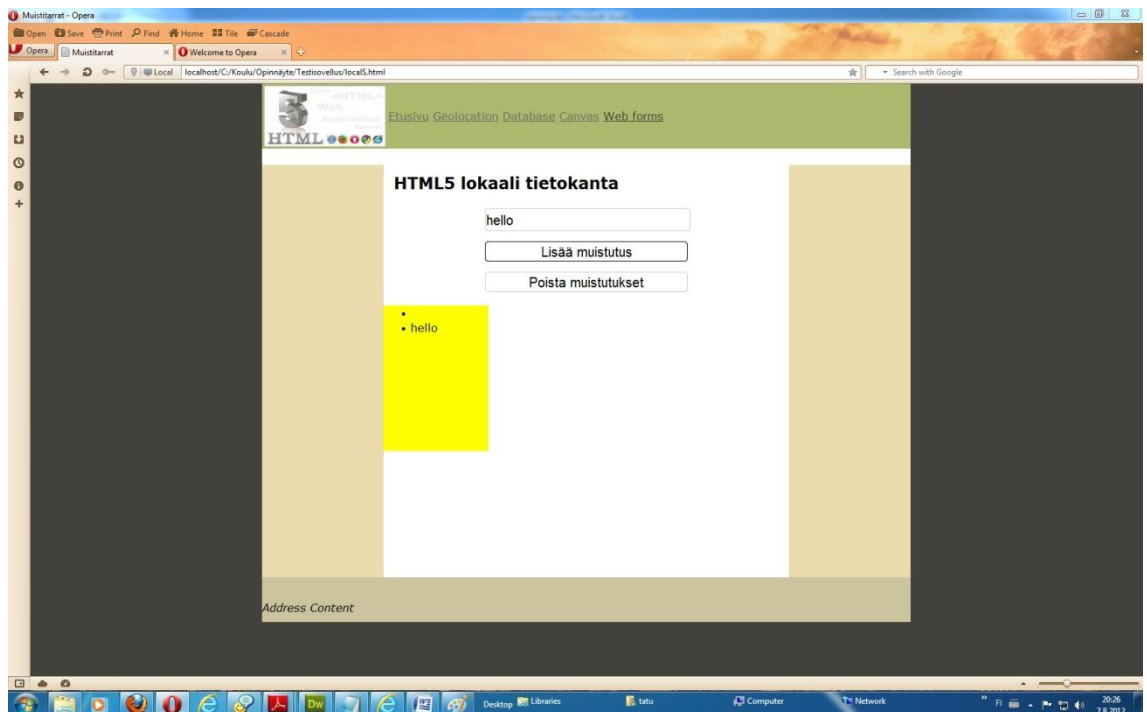


Testisovelluksen Geolocation sivu

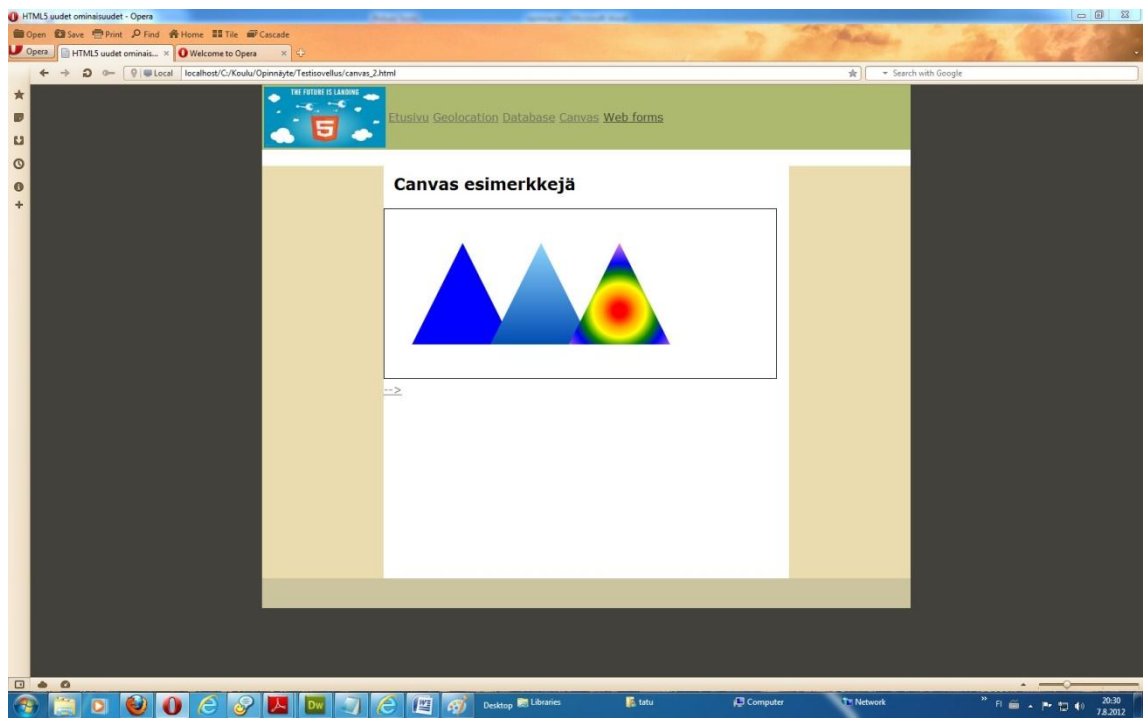
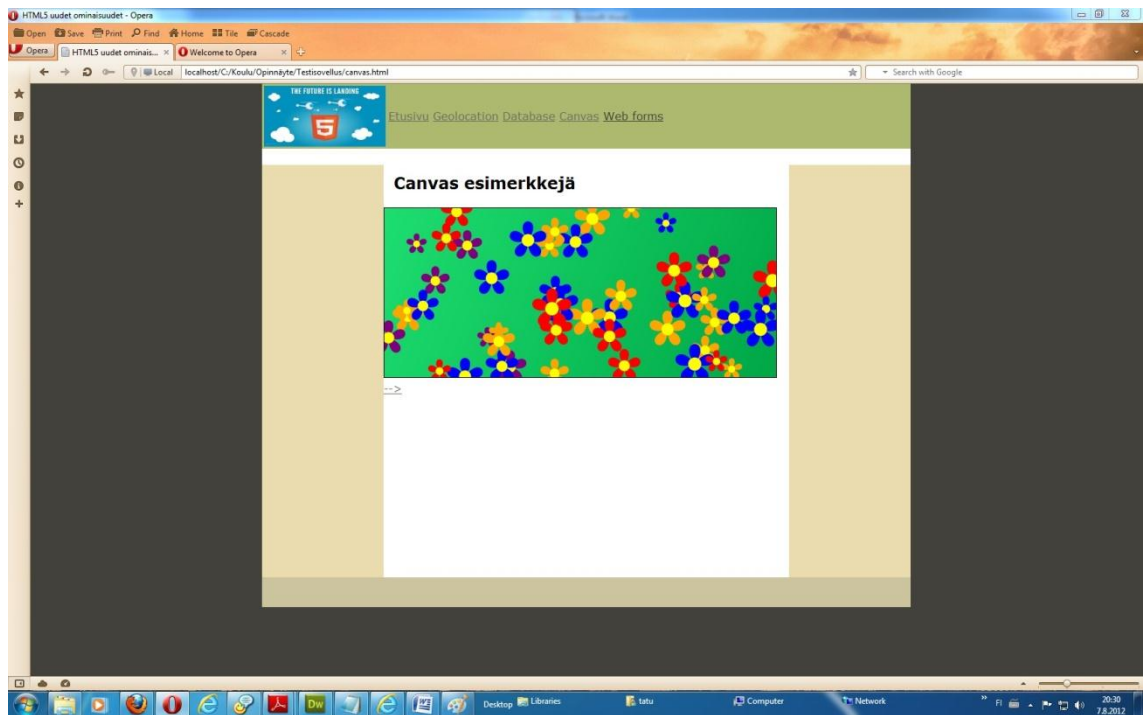


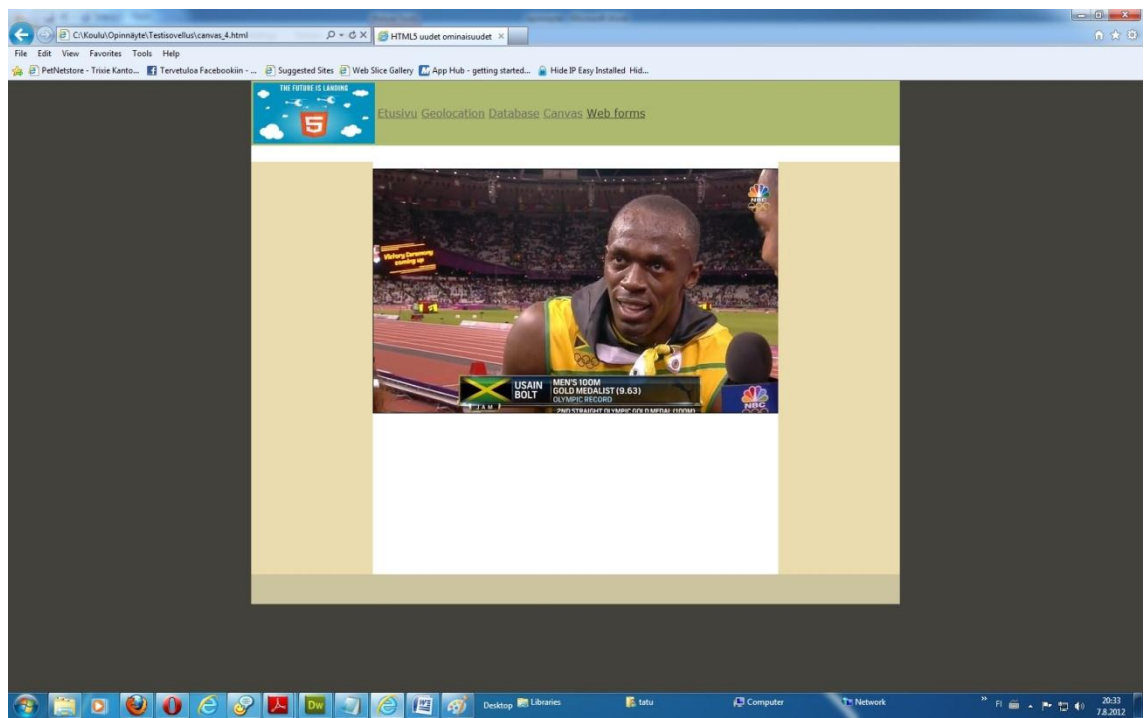
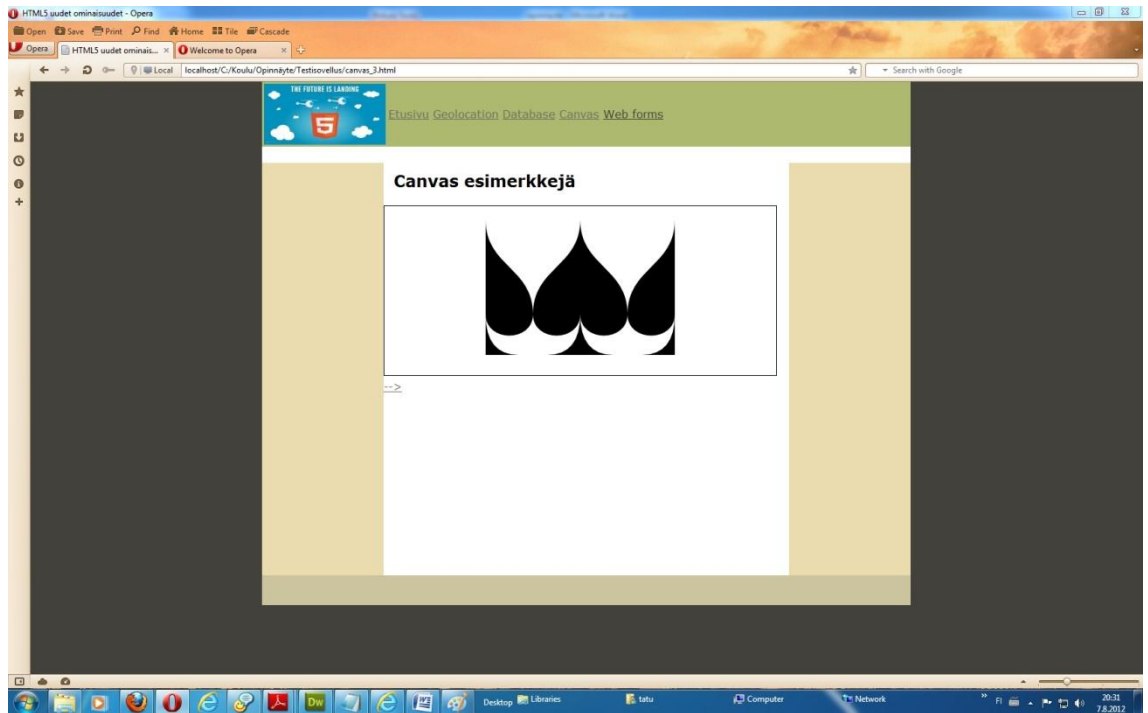


Testisovelluksen Database sivu

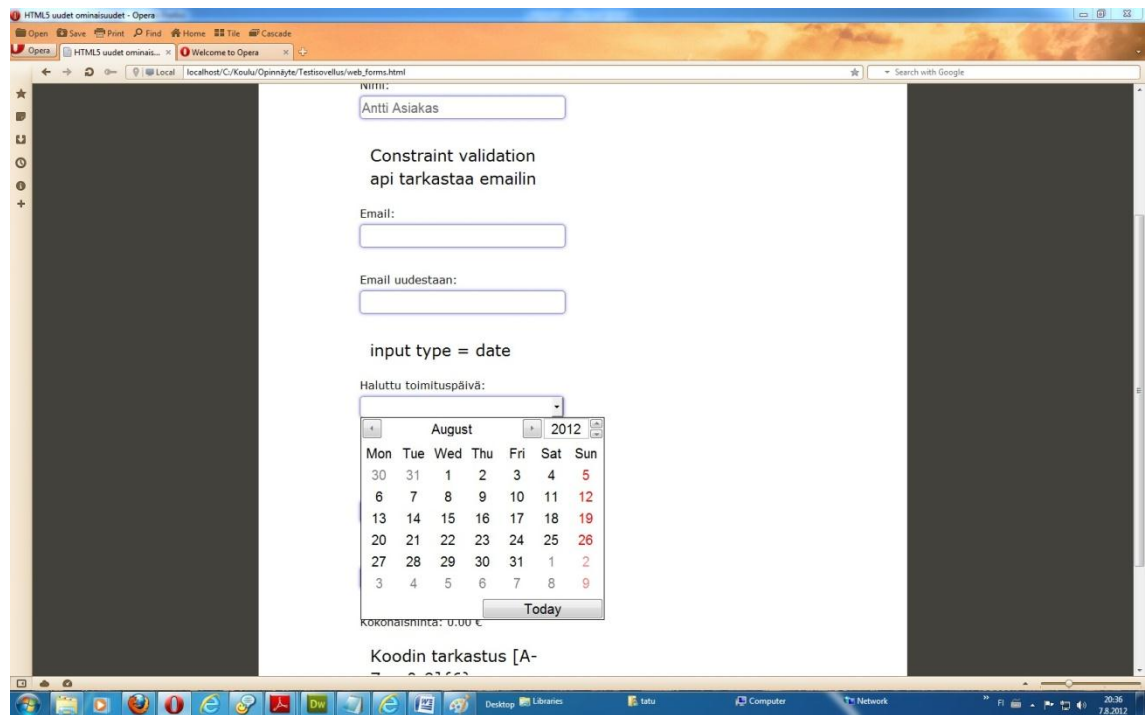


Testisovelluksen Canvas sivut





Testisovelluksen Web Forms sivu



## Loppuraportti

Tutkimuksen aihe oli tutkimuksen tekijän oma ehdotus, joka muotoutui lopulliseen ulkoasuun tutkimuksen edetessä. Alunperin oli tarkoitus tehdä server to client- tason toteutus, mutta se typistettiin vain koskemaan client-puolta liian laajan tutkimusongelman vuoksi. Ratkaisu oli hyvä, koska muuten työtunnit olisivat menneet yli lopputyön vaatimuksista. Aiheen valintaan vaikutti ratkaisevasti omat opinnot web-tekniikoiden parissa.

Tutkimuksen tavoitteena oli tutustua muutamaiin tärkeimpiin HTML5-ohjelmointirajapintoihin ja testata miten ne toimivat uusimmissa selaimissa. Tutkimusympäristön koodaus oli suhteellisen helppoa ja hieman yllätti miten yksinkertaista joidenkin ominaisuuksia on testata. Kaikkia uusia ominaisuuksia ei toki edes yritetty 'väkisin' upottaa testisivustoon. Projektin edetessä tutkittavaksi suunniteltu Offline-ominaisuus vaihdettiin Canvasiin, koska offlinen tutkimus pelkällä client-puolen sovelluksella olisi ollut liian vähäistä. Tutkimuksen aikana selkeni, että HTML5 tulee olemaan standardi miten web-sivuja tehdään jo nyt ja tulevaisuudessa. Selainvalmistajat ovat tässä avainasemassa, mitä uudistuksia he tukevat. W3C tekee arvokasta työtä ja toivottavasti näemme pian ehjän HTML5 standardin, mikä on tuettu kaikkien selainvalmistajien toimesta.

Tekijän omana tavoittena oli sisäistää HTML5:n syntaksi, oppia scriptien käyttö HTML-dokumentin sisällä ja oppia hyödyntämään paikallisia tietovarastoja, jotka sallivat offline-tilassa työskentelyn sekä ymmärtää Geolocationin tarjoamat mahdollisuudet ja rajoitteet. Tässä jäätin puolitiehen session storagen käytön suhteen, koska se olisi vaatinut sivuston, joka sisältää palvelinpuolen koodia. Tutkimuksen tavoitteet täytettiin kuitenkin hyvin ja kuva siitä 'missä mennään' selkeni. HTML5 ei ole läheskään valmis, paljon on työtä tekemättä. Muutamat seikat yllätti tutkimusta tehtäessä, varsinkin se kuinka paljon eroja selainten välillä on. Normaalikäytössä niitä ei huomaa, mutta kun testaa niiden toimivuutta eri sovelluksille, niin eroja alkaa syntyä. Askel paremman ja standardoitumman

Internetin suuntaan on kuitenkin jo otettu, mutta 'villistä netistä' ei päästäne koskaan eroon ja se on tavallaan myös sen rikkaus, jokainen joka osaa edes hiukan html:ää tai käyttää julkaisuohjelmia, voi tehdä omat sivut world wide webbiin.

Testauksessa käytettiin kolmea eri valmistajan selainta, niiden uusimpia markkinoilla olevia versioita; IE, Firefox ja Opera. Tämäkin otos jo riitti todistamaan, ettei mitään yhtenäisyyttä eli standardia ole saavutettu selainvalmistajien kesken.

Itse työn lähdeaineistoon tutustuminen ja opinnäytteen kirjoittaminen vei suurimman osan ajasta. Suurin osa aineistosta oli englanninkielistä ja huolimatta tutkijan suhteellisen hyvästä kielitaidosta paljon aikaa kului varmistuksiin, ettei ole sattunut väärinymmärryksiä. Aineisto sisälsi kuitenkin hyvin paljon teknistä sanastoa ja välillä vaikeasti ymmärrettäviä sana- ja kirjoitusmuotoja.

Tämän työn resurssit liittyivät lähinnä tutkijan omaan ajankäyttöön. Ohessa liite ajankäytöstä. Testisivuston tekemiseen meni huomattavasti vähemmän tunteja kuin oli arvioitu, mutta aineiston läpikäymiseen kului enemmän kuin arvioitiin. Kustannuksia ei työstä syntynyt.

Lopputyö	tehtävä	lopputulos	aloituskriteeri		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
					Viikot																	
	<b>1 Projektin käynnistys</b>	<b>hyväksytty projektisuunnitelma</b>	<b>hyväksytty aihe-ehdotus</b>	<b>20</b>																		
	1.1 Projektisuunnitelman teko	Projektisuunnitelma		18	18																	
	1.2 Aloituskokous	käynnistetty projekti	p-suunnitelma valmis	2	2																	
	<b>2 Aineisto</b>	<b>Dokumentaatio</b>	<b>hyväksytty projektisuunnitelma</b>	<b>275</b>																		
	2.1 Aineiston keruu ja analysointi	Riittävä aineisto koossa		167		20	20	20	20	15	8	8	8	8	loma	loma	loma	8	8	8	8	8
	2.2 Lopputyön kirjoittaminen	Lopputyöraportti		108		2	2	2	2	2	2	2	8	8				8	10	20	20	20
	<b>3 Suunnittelu</b>	<b>kaaviot yms.</b>	<b>hyväksytty projektisuunnitelma</b>	<b>8</b>																		
	3.3 Web-sivun suunnittelu	tarvittavat scriptit, asetelu, ulkonäkö yms.		8				0	0	0	8											
	<b>4 Toteutus</b>	<b>web-sivuston 1. versio</b>	<b>Suunnittelu valmis</b>	<b>60</b>																		
	4.3 Web-sivun toteutus	web-sivu		10								10	0	0				0	0			
	4.4 Geolocation	web-sivu		9									9									
	4.6 Lomakkeet	web-sivu		6										6				0	0			
	4.7 CSS3	web-sivu		5								5	0	0								
	4.8 Testaus	testaustulokset		30																	30	
	<b>5 Projektin ohjaus</b>	<b>hallittu edistyminen</b>	<b>projekti on käynnistetty</b>	<b>25</b>																		
	5.1 Edistymisen raportointi ja seuranta	edistymisraportti	seurantajakso umpeutuu	6	2							2							2			
	5.2 Kokoukset	materiaalin kokoaminen	sovittu kokousaika	6	2							2										2
	5.3 Pöytäkirjan laatiminen	pöytäkirja	kokous on pidetty	3	1							1										1
	5.4 Loppuraportti	Loppuraportti		10																		10
				388	25	22	22	22	22	17	18	30	25	22	0	0	0	16	20	28	58	41

Lopputyön toteutuneet tunnit