



SELAINKÄYTTÖISTEN SOVELLUSTEN KÄYTTÖLIITTYMÄSUUNNITTELU JA KÄYTETTÄVYYS

Tampereen ammattikorkeakoulu
Viestinnän koulutusohjelman opinnäytetyö
Vuorovaikutteisuuden suunnittelun suuntautumisvaihtoehto
Kevät 2009
Krista Jouhtimäki

OPINNÄYTTEEN TIIVISTELMÄ

Krista Jouhtimäki

Selainkäyttöisten sovellusten käyttöliittymäsuunnittelu ja käytettävyys

Huhtikuu 2009

45 sivua

Tampereen ammattikorkeakoulu

Viestinnän koulutusohjelma

Vuorovaikutteisuuden suunnittelu

Lopputyön muoto: kirjallinen

Lopputyön ohjaaja: Ari Närhi

Avainsanat: käyttöliittymät, käytettävyys

Selainkäyttöiset sovellukset ovat yleistyneet nopeasti, koska tekniikka ja selaimet ovat kehittyneet ja toteutuksesta on tullut helpompaa. Vaikka toteutus ei edelleenkään ole erityisen halpaa tai nopeaa, on sovellusten hyödyllisyys asiakastyytyväisyyden ja liiketoiminnan kannalta otettava huomioon.

Web-sovelluksista voidaan uusilla tekniikoilla tehdä aidosti vuorovaikutteisia ja nopeita käyttää, eli ne muistuttavat yhä enemmän työpöytäsovelluksia. Vaikka RIA-sovelluksia on tehty aiemminkin, on esimerkiksi Ajax muuttanut toteutusmallia enemmän web-sivun kaltaiseksi, jolloin perinteiset konventiot eivät enää päde. Selaimessa käytettävä sovellus poikkeaa perinteisistä työpöytäsovelluksista ja web-sivuista sekä käyttöliittymäsuunnittelun että käytettävyyden kannalta, joten suunnittelijat saavat sekä uusia mahdollisuuksia toteuttaa sovelluksia että uudenlaisia suunnitteluhaasteita.

Koska aihe on uusi, on kattavaa tutkimusta toistaiseksi vaikea löytää ja parhaat käytännöt hakevat vielä muotoaan. Tutkimuksessa selvitetään miten RIA-sovellukset poikkeavat työpöytäsovelluksista ja web-sivuista ja mitä niiden suunnittelussa tulee siksi ottaa huomioon. Apuna käytetään olemassa olevia RIA-toteutuksia, joiden ominaisuuksia esitellään käytettävyyden ja käyttöliittymäsuunnittelun näkökulmasta.

THESIS SUMMARY

Krista Jouhtimäki

Rich Internet Applications, UI design and usability

April 2009

45 pages

TAMK University of Applied Sciences

Media Programme

Area of specialisation: Interactivity Design

Type of Final Project: Written

Thesis supervisor: Ari Närhi

Keywords: user interfaces, usability

Abstract:

Rich web applications have recently become extremely popular, mainly because technology and browsers have improved and applications are easier to develop. Though not the cheapest or quickest to develop, rich web applications make a great improvement in customer satisfaction and business. Today using a web application can be as fast and interactive as using a desktop application. Technologies like Ajax make web applications seem more like web pages, so well known conventions used in web pages and desktop applications no longer apply. This makes designers face new challenges, but gives new opportunities too.

This thesis is written because studies of the subject are so far hard to find and best practices are still under development. The main goal is to find out how rich Internet applications differ from desktop applications and web pages, and how that affects design. As a reference, a few today's popular RIAs are evaluated from the usability and UI design point of view.

Sisällys

Johdanto	6
1. Terminologia.....	6
1.1 Lyhenteet	6
1.2 Termit	6
1.2.1 Työpöytäsovellus/työpöytäohjelmisto.....	6
1.2.2 Web-sovellus/RIA-sovellus	7
1.2.3 Web-sivu	7
2. RIA-sovellukset.....	7
2.1 Määritelmä	7
2.2 Edut ja hyödyt	8
2.3 Huonot puolet	9
2.4 Esimerkkejä RIA-sovelluksista.....	10
2.4.1 Weebly	12
2.4.2 Grokker	13
2.4.3 Wordpress	14
2.4.4 Flickr	15
2.4.5 Flex ja Silverlight.....	15
2.5 Yhteenveto	17
3. Käyttöliittymä	18
3.1 Käyttöliittymäsuunnittelun perusteita.....	18
3.2 RIA-käyttöliittymän erityispiirteet	19
3.3 Käyttöliittymän dokumentointi.....	20
3.3.1 Konseptitaulut.....	20
3.3.2 Flowframes	20
3.3.3 Prototyypit	22
3.4 Käyttöliittymän elementit.....	22
3.5 Tila ja tilan vaihtuminen	24
3.6 Sivut	25
3.7 Yhteenveto	26
4. Käytettävyys	27
4.1 Mitä on käytettävyys	27

4.2 Käytettävyyden arvioiminen	27
4.3 Heuristinen malli	27
4.3.1 Tilan näkyminen käyttäjälle	28
4.3.2 Vastaavuus tosimaailman kanssa	28
4.3.3 Käyttäjän kontrolli ja vapaudet	28
4.3.4 Yhdenmukaisuus ja standardit	29
4.3.5 Virhetilanteiden välttäminen	29
4.3.6 Tunnistaminen, ei muistaminen	30
4.3.7 Joustavuus ja käytön tehokkuus	30
4.3.8 Esteettinen ja minimalistinen design	32
4.3.9 Auta käyttäjää tunnistamaan virheet ja toipumaan niistä.....	33
4.3.10 Opastus ja ohjeistus	34
4.4 Käyttäjättestaus	35
4.5 Esimerkki: Heuristinen analyysi	36
4.5.1 Flickr	36
4.6 Yhteenveto	41
5 Yhteenveto	42
Lähteet	43
Internet-lähteet	43
Esimerkit.....	44

Johdanto

Tutkimus käsittelee selainkäyttöisiä sovelluksia (Rich Internet Applications, RIA) ja niiden erityispiirteitä käytettävyyden ja käyttöliittymäsuunnittelun kannalta. Useimmat käyttöliittymäsuunnittelun ja käytettävyyden periaatteet pätevät myös suunniteltaessa selainkäyttöisiä sovelluksia, mutta tiettyjä erityispiirteitä on otettava huomioon suunnittelussa. Tutkimus luotaa perinteisen käyttöliittymäsuunnittelun ja käytettävyydetutkimuksen metodeja, mutta ottaa niihin uudenlaisen näkökulman.

Selainkäyttöiset sovellukset ovat Internet-sivun ja työpöytäohjelmiston välimaastossa ja saavat molempien piirteitä käyttötapauksesta riippuen. Lisäksi myös selaimen ominaisuudet vaikuttavat suunnitteluun. Selaimessa toimivat sovellukset ovat myös laajentaneet käyttöliittymäsuunnittelijoiden kenttää ja tuoneet uudenlaisia haasteita, sillä alueella ei ole vielä paljonkaan vakiintuneita toimintamalleja, toisin kuin työpöytäsovelluksissa ja tavallisissa Internet-sivuissa.

1. Terminologia

1.1 Lyhenteet

Ajax Asynchronous JavaScript and XML

RIA Rich Internet Applications

XML eXtensible Markup Language

1.2 Termit

1.2.1 Työpöytäsovellus/työpöytäohjelmisto

Termiä työpöytäsovellus käytetään tekstissä yleistyksenä asennettavista ja ajettavista sovelluksista (engl. executables).

1.2.2 Web-sovellus/RIA-sovellus

Web-sovelluksella tai RIA-sovelluksella tarkoitetaan tekstissä samaa asiaa, selaimella käytettävää sovellusta, jota ei tarvitse erikseen ladata ja suorittaa paikallisesti. Tarkka määrittely kappaleessa 2: RIA-sovellukset.

1.2.3 Web-sivu

Web-sivulla tarkoitetaan tekstissä perinteistä Internet-sivua, jolla ei ole korkean tason vuorovaikutteisia elementtejä, laskenta suoritetaan palvelinpuolella ja sivu ladataan kokonaisuudessaan uudestaan kun sisältö päivittyy.

2. RIA-sovellukset

2.1 Määritelmä

RIA-sovellus on selaimella käytettävä toiminnallisuudeltaan rikas sovellus, jossa on yleensä hyödynnetty korkean tason vuorovaikutustekniikoita. RIA-sovellukset voidaan toteuttaa esimerkiksi Javascriptillä, avuksi löytyy useita valmiita Javascript-kirjastoja. Javascriptillä rakennettu sovellus ei tarvitse erillisiä lisäosia toimiakseen, vaan se toimii suoraan selaimessa. RIA-sovellusten kehittämisen Javascriptillä mahdollistaa Ajax-tekniikka, joka yhdistää useita tuttuja web-tekniikoita (kuten XHTML ja CSS) saman nimityksen alle.

Termin Ajax keksi Jesse James Garrett vuonna 2005 merkitsemään tekniikoita, jotka mahdollistavat selaimen kommunikoimisen palvelimen kanssa ilman koko sivun uudelleenlatausta (Wikipedia: Ajax 2008). Tällöin sivusta ladataan uudestaan vain tarvittava osa, jolloin palvelimen ei tarvitse lähettää kaikkea sivun sisältämää tietoa jokaisella latauskerralla. Tämä vähentää tarvittavan liikenteen määrää, jolloin lataukset voidaan suorittaa nopeammin. Lisäksi suuri osa toiminnoista voidaan suorittaa ottamatta yhteyttä palvelimeen. Yleensä RIA-sovelluksissa tarvittavat laskennat tehdään asiakaspuolella (engl. client side), jolloin palvelimeen tarvitsee ottaa yhteyttä vain haettaessa ja tallennettaessa dataa. (Wikipedia: Ajax 2008, Vesa 2008)

Toinen vaihtoehto RIA-sovelluksen tekemiseen on käyttää rich web client -sovelluksia kuten Adobe Flex, Adobe Flash tai Microsoftin Silverlight. Tällöin tarvitaan selaimen asennettava lisäosa (engl. plugin) ja kyseessä on tarkkaan ottaen työpöytäsovellus, joka asennetaan ja käynnistetään verkon yli. Sovellukset ovat tulossa myös mobiililaitteisiin, esimerkiksi Silverlight-ajoympäristöstä on jo olemassa mobiiliversio. (Wikipedia: Rich Internet applications 2008, Vesa 2008)

Myös työpöytäsovellusten suunnittelijoille on tarjolla uusia haasteita, sillä niissä voidaan käyttää myös Internet-sisältöä. .NET tai Java/Swing –tekniikoilla rakennetut työpöytäsovellukset voivat käyttää verkkosisältöä hyödykseen. Työpöytäsovelluksen etuna on mahdollisuus toteuttaa paljon hienostuneempi käyttöliittymä kuin selaimen on nykytekniikoilla mahdollista toteuttaa.

Joskus ladattavan sovelluksen tarjoaminen web-sovelluksen rinnalla on hyödyllistä, sillä se mahdollistaa paremman käyttökokemuksen sellaisilla päätelaitteilla, joihin ohjelma on mahdollista asentaa. Esimerkiksi Flickr.com tarjoaa web-käyttöliittymän lisäksi myös työpöytäsovelluksen, jolla voi hallita verkossa näytettävää sisältöä. (Cooper, Reimann & Cronin 2007, sivut 178-182)

2.2 Edut ja hyödyt

Perinteiset web-sovellukset nojaavat pitkälti sivun käsitteeseen. Tämä siksi, että aina Ajaxin syntyyn asti on koko sivu täytynyt ladata palvelimelta minkä tahansa sisällön vaihtuessa. Tämä muistuttaa tavallisen kirjan sivun kääntämistä: on odotettava hetki ennen sisällön vaihtumista. Internetin alkuaikoina, ja usein vielä nykyisinkin, myös sisältö muistutti kirjaa tai lehtistä. Työpöytäsovelluksissa tämä metafora ei ole koskaan ollut käytännöllinen, eikä sitä ole ollut syytä käyttää. (Heller 2005)

Selainkäyttöisten sovellusten yleistyessä syntyi tarve uudelle tekniikalle. Sovelluksissa useimmiten lähetetään käyttäjän antamaa tietoa palvelimelle ja näytetään tarvittaessa käyttäjälle uutta sisältöä, jolloin palvelimelta ladataan kokonainen sivu. Aikaisemmin sovellusten käyttäminen on ollut hidasta, pääosin siksi että jokaisen toiminnon yhteydessä kaikki sivun sisältämä data lähetetään joko palvelimelle tai käyttäjän päätelaitteeseen. Uudet tekniikat mahdollistavat sivun osittaisen lähettämisen ja uudelleenlataamisen, jolloin selainkäyttöiset sovellukset voidaan saada toimimaan lähes

yhtä nopeasti kuin työpöytäsovellukset. (Wikipedia: Rich Internet applications 2008, Perfetti, Heller 2005)

RIA-sovellukset tuovat selkeitä etuja työpöytäsovelluksiin verrattuna. Ensinnäkin niitä voi käyttää miltä tahansa koneelta, jossa on Internet-yhteys ja jokin selain asennettuna. Erillistä ohjelmaa ei usein tarvitse asentaa lainkaan, tai selaimen lisäosan asentaminen käy helposti (mikäli käyttäjällä on oikeudet siihen). Esimerkiksi Flashin näyttämiseen tarvittava lisäosa löytyy jo useimmista selaimista. RIA-sovellukset ovat riippumattomia käyttöjärjestelmästä ja turvallisempia kuin ajettavat ohjelmat (engl. executables). RIA-sovelluksia voidaan myös päivittää vaatimatta käyttäjältä erillistä päivityksen asennusta, eli päivitykset ovat käyttäjälle pääosin huomaamattomia. RIA-sovelluksiin voidaan myös tarvittaessa toteuttaa mahdollisuus offline-käyttöön. (Wikipedia: Rich Internet applications 2008)

RIA-sovellukset keventävät palvelinkuormaa, sillä liikennettä on vähemmän ja palvelimelta vaaditaan vähemmän suorituskykyä kun suuri osa laskennasta suoritetaan asiakkaan resursseilla. Asiakas voi myös kommunikoida palvelimen kanssa itsenäisesti ja esimerkiksi hakea dataa valmiiksi muistiin, jolloin käyttäjälle näkyviä vasteaikoja saadaan lyhennettyä. (Wikipedia: Rich Internet applications 2008)

2.3 Huonot puolet

RIA-sovellus ei automaattisesti ole parempi ratkaisu kuin työpöytäohjelmisto. Ennen tuotekehityksen aloittamista on pohdittava mitä etuja saavutetaan toteuttamalla sovellus selainkäyttöiseksi. RIA-sovelluksen toteuttaminen ei ole helpompaa tai nopeampaa kuin vastaavan työpöytäsovelluksen kehittäminen. Lisäksi suunnittelussa on huomioitava selaimen rajoitteet ja ominaisuudet. (Cooper, Reimann & Cronin 2007, sivut 178-182)

RIA-sovellukset toimivat hiekkalaatikossa (engl. sandbox), joka tekee niistä melko turvallisia käyttäjälle. Sovellukset eivät pääse käsiksi järjestelmän resursseihin, kuten esimerkiksi tiedostoihin. Ohjelman toiminta saattaa häiriintyä kriittisesti jos kehittäjät arvioivat väärin sovelluksen oikeudet järjestelmässä. Lisäksi skriptien suorittamisen selaimessa voi estää, jolloin RIA-sovelluksen toiminta estyy.

Asiakaspuolen prosessointinopeus tai yhteyden hitaus saattaa myös rajoittaa sovelluksen toimintaa, etenkin kannettavien päätelaitteiden ollessa kyseessä. Lisäksi

RIA-sovellukset ovat pääosin riippuvaisia Internet-yhteydestä. Kehitystä vaikeuttaa myös se, että selaimet kehittyvät ja muuttuvat huomattavasti nopeammin kuin työpöytäohjelmiston alusta (Perfetti).

Yksi suuri ongelma on myös sisällön saaminen hakukoneiden saataville, koska hakukoneet eivät pysty indeksoimaan dynaamista tekstisisältöä.

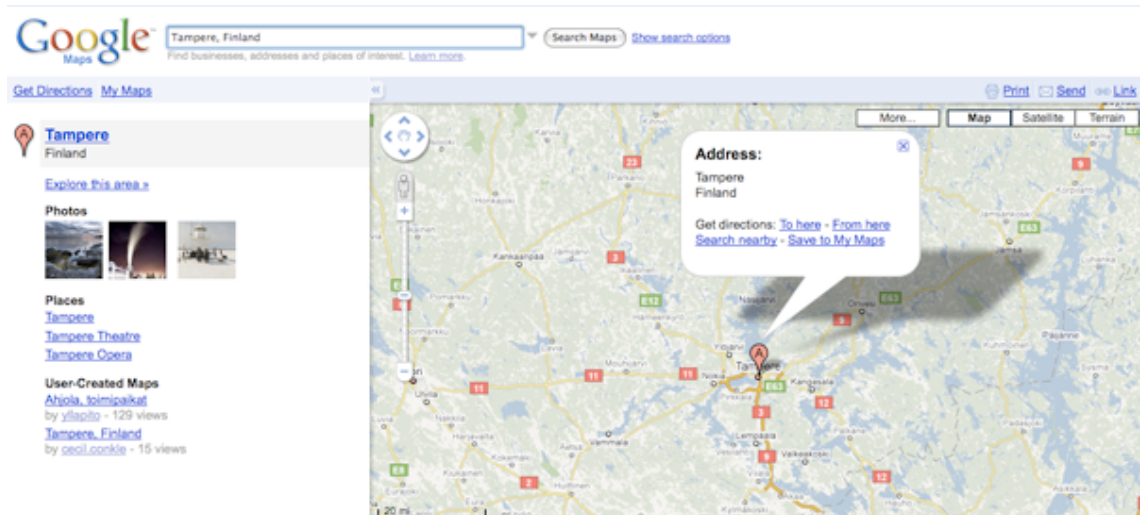
(Wikipedia: Rich Internet applications 2008)

RIA-sovellukset ovat vielä tätä kirjoitettaessa kehityksen alkupäässä eivätkä kovin tuttuja käyttäjille. Siihen miksi sovellukset eivät toistaiseksi ole eduistaan huolimatta yleistyneet vaikuttaa esimerkiksi se, että monet RIA-sovellukset vaativat modernin selaimen ja tehokkaan Javascript-moottorin toimiakseen, ja vanhoja selaimia käytetään edelleen huomattavan paljon. Lisäksi kaikki selaimet eivät tue web-standardeja kunnolla, joten RIA-sovellusten kehittäminen useille selaimille on vaikeaa. Myös selaimen valmiit toiminnot vaikeuttavat suunnittelua ja käyttöä.

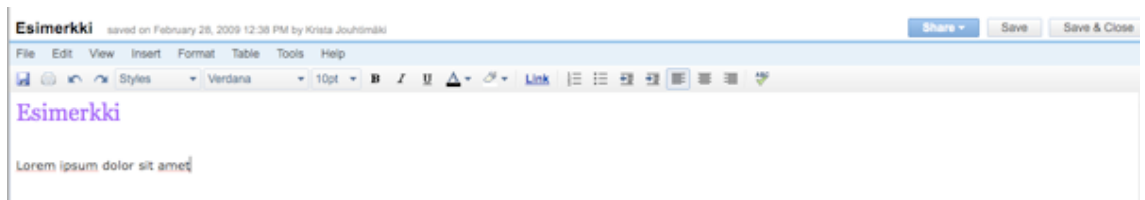
Edistystä kuitenkin tapahtuu nopeasti ainakin tekniikan puolella, esimerkiksi Googlen uusi selain ottaa kantaa RIA-sovellusten kehitykseen, sillä siinä on uusi V8 Javascript-moottori, jonka luvataan nopeuttavan Javascriptin suoritusta ja mahdollistavan täysin uudenlaisia web-sovelluksia. (Wikipedia: Rich Internet applications 2008)

2.4 Esimerkkejä RIA-sovelluksista

RIA-sovelluksia on ainakin kahdenlaisia, sekä itsenäisiä että web-sivuston osaksi rakennettuja sovelluksia. Googlen sovellukset, kuten Gmail (<http://mail.google.com>) ja Google Maps (<http://maps.google.com>) ovat ehkä tunnetuimpia RIA-tekniikoilla toteutettuja itsenäisiä web-sovelluksia. Esimerkiksi Google Docs (<http://docs.google.com>) muistuttaa huomattavan paljon tavallista tekstinkäsittelyohjelmaa. Tutkimuksen pääpaino on Ajax-sovelluksissa, sillä ne eivät vaadi erillistä ohjelmaa tai lisäosaa toimiakseen, mikä tekee niistä aitoja RIA-sovelluksia.

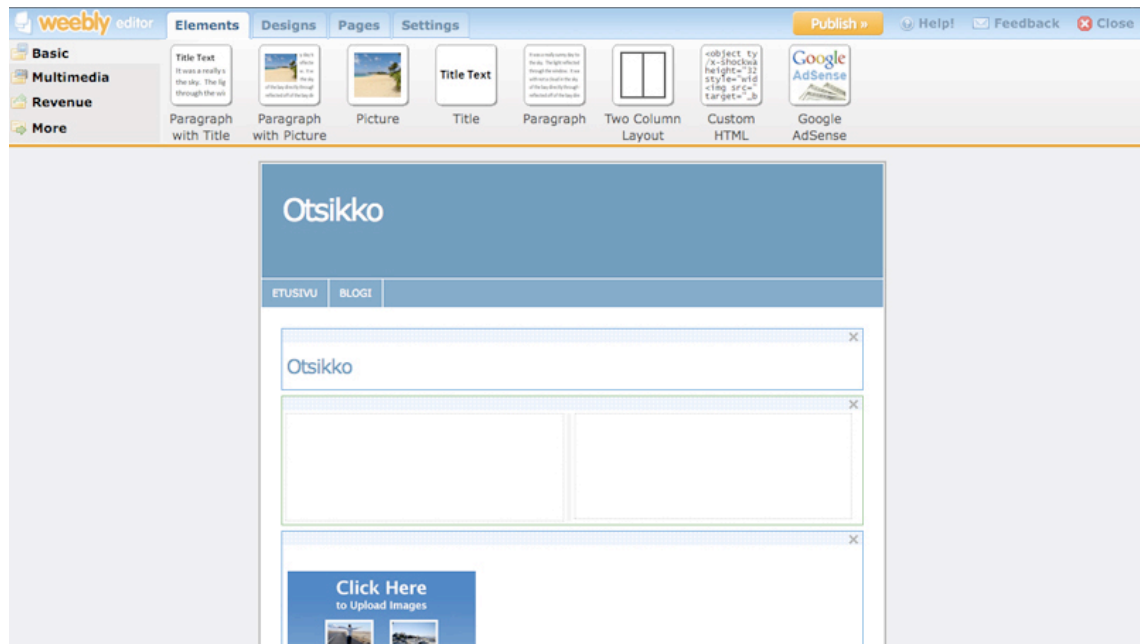


Kuva 1 Google maps –sovellus



Kuva 2 Google docs - tekstinkäsittelyohjelma

2.4.1 Weebly



Kuva 1 Weebly - sivun ylläpito

Weebly (<http://www.weebly.com>) on Javascriptin avulla toteutettu RIA-sovellus, jolla voi ylläpitää omaa web-sivustoa tai blogia. Weeblyssä käytetään hyväksi paljon kehittyneitä vuorovaikutustekniikoita: elementtejä voi lisätä ylläpidettävälle sivulle raahaamalla, tyyliä vaihtaessa kursorin vieminen uuden tyylin kuvakkeen päälle vaihtaa ylläpidettävän sivun tyylin dynaamisesti. Weebly on ulkoasultaan ja toiminnoiltaan viimeistellyn oloinen ja käytettävyys on selkeästi otettu huomioon suunnittelussa.

2.4.2 Grokker

The screenshot displays the Grokker search engine interface. At the top left is the Grokker logo. Below it, a search bar contains the text 'cat'. To the right of the search bar are buttons for 'GROK' and 'Search Options'. Below the search bar, there are tabs for 'Outline View' and 'Map View', with '250 total results' displayed. The main area is a circular map view showing various search results as icons. A tooltip is visible over one of the icons, showing the following information:

Title	Black Cat
Date	Jan 7, 2009
Rank	43
Source	Yahoo!

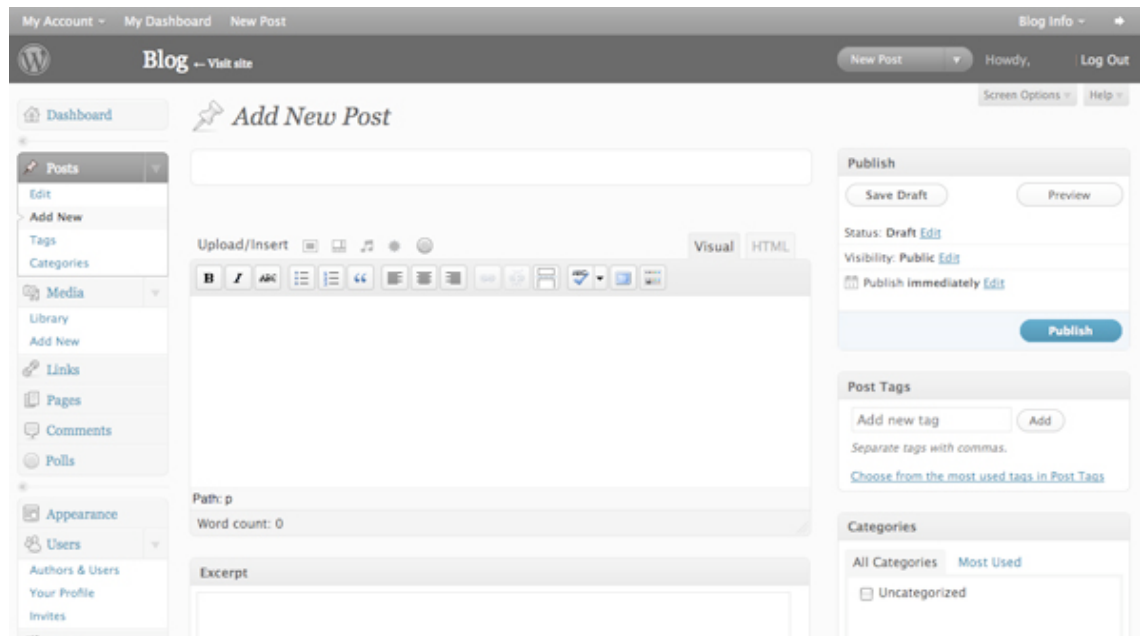
Other visible results on the map include 'Sweetness', 'iCat', 'Arctic Cat, Inc.', 'Cat Dynamics - Cat...', and 'CatGenie - The Wor...'. On the right side, there is a 'Detail' pane showing the selected result: 'iCat'. The detail pane includes links for 'Add to Working List', 'Post to del.icio.us', 'Bookmark', and 'Email'. The description for 'iCat' reads: 'Providing free software solutions for developing commerce-enabled online stores. http://www.icat.com/ - 7k - 01/12/2009 Source: Yahoo!'. Below this, other results like 'Arctic Cat, Inc.', 'Cat Dynamics - Cat Technology Specialists', and 'CatGenie - The World's Only Self-Flushing, Self Washing Cat Box' are visible.

©2006 Groxis Inc., All Rights Reserved.

Kuva 2 Grokker

Grokker (<http://www.grokker.com>) on Ajax-tekniikoilla toteutettu innovatiivinen hakusovellus. Hakusanan perusteella luodaan hakusanakartta, jonka avulla voi tarkentaa hakuaan. Vasemman reunan lomakkeella voi vaihtaa tai tarkentaa hakuetojaan, haun tulokset päivittyvät dynaamisesti oikean reunan kolumniin. Hakusanakartassa voi hiiren avulla tarkentaa tai laajentaa näkymää siten, että lähemmäs liikuttaessa hakusanat tarkentuvat.

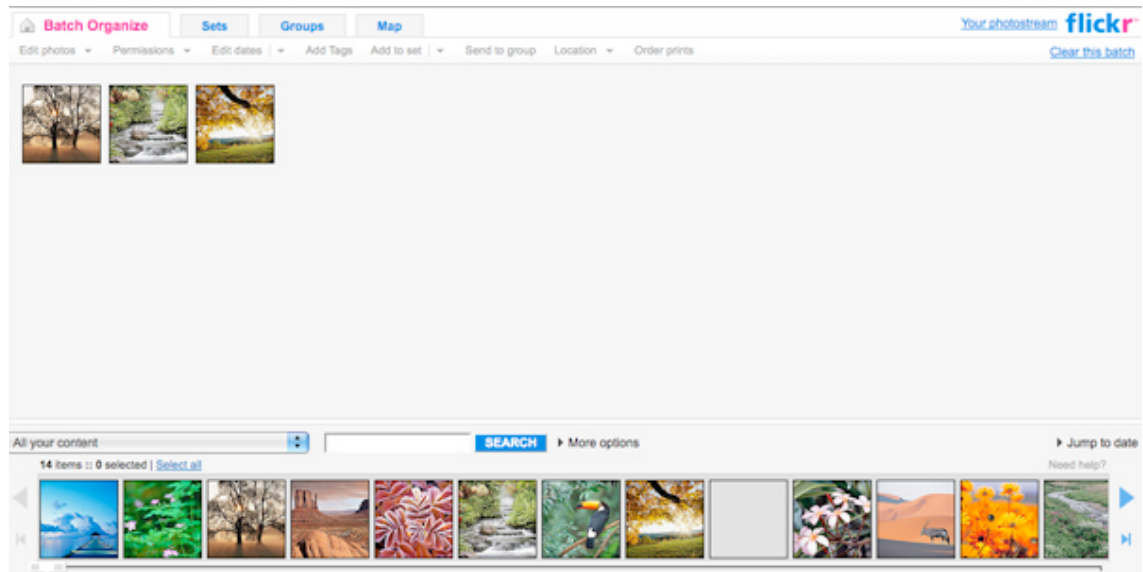
2.4.3 Wordpress



Kuva 3 Wordpress

Wordpress (<http://www.wordpress.com>) on blogien ja sivustojen luomiseen ja julkaisuun tarkoitettu web-sovellus. Wordpress tarjoaa paljon ominaisuuksia esimerkiksi yksityisyyteen liittyen, kuten sivujen suojaamisen salasanalla tai julkaisun vain valittujen henkilöiden luettavaksi. Lisäksi sovelluksessa on mahdollista tallentaa tiedostoja kirjastoon, lisätä posteihin videota tai ääntä ja muokata ulkoasua haluamallaan tavalla. Wordpressissä on hyvin toteutettu tekstinkäsittelyalue, jossa voi käyttää apuna pikanäppäimiä.

2.4.4 Flickr



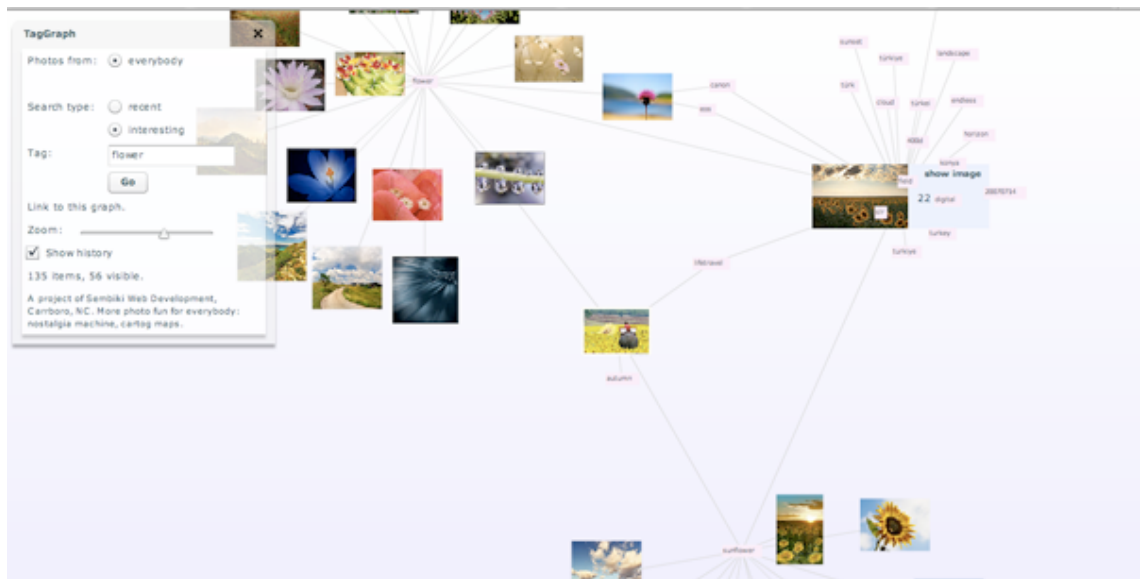
Kuva 4 Flickr

Flickr (<http://www.flickr.com>) on kuvien jakamiseen tarkoitettu sovellus. Kuvia voi lisätä, poistaa, järjestää erilaisiksi kokonaisuuksiksi ja julkaista haluamallaan suojauksilla. Flickrin tutustutaan tarkemmin osiossa 4.5.1 Heuristinen analyysi: Flickr.

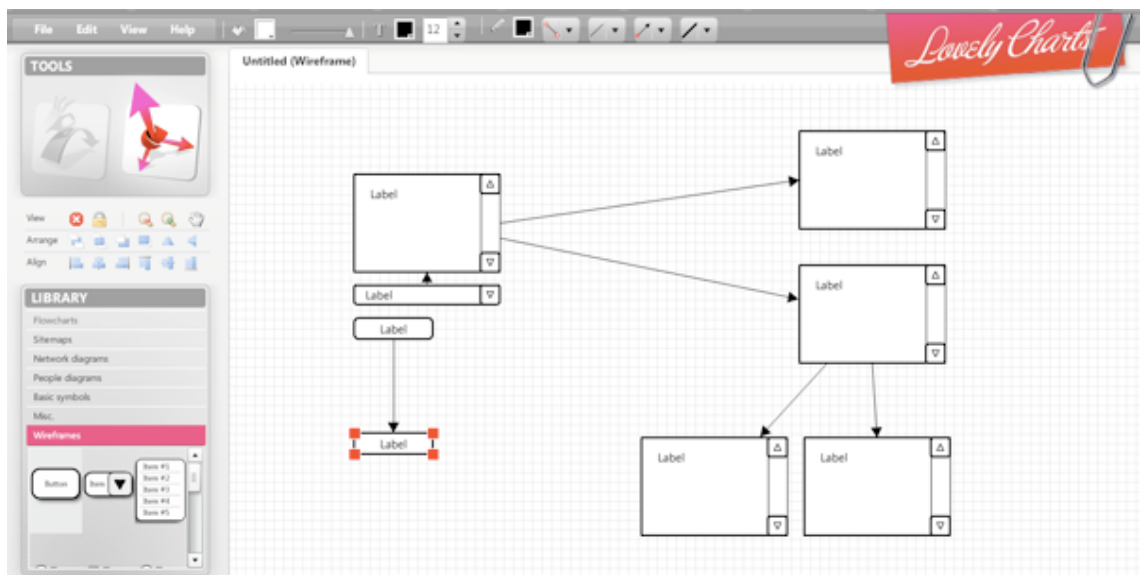
2.4.5 Flex ja Silverlight

Selainkäyttöisiä sovelluksia voi toteuttaa myös erillisen lisäosan vaativilla tekniikoilla kuten Adobe Flex ja Flash tai Microsoftin Silverlight. Flex ja Flash vaativat toimiakseen Adobe Flash Player –lisäosan, joka asennetaan selaimeen, Silverlight vaatii oman sovelluksensa. Adoben Flash Player löytyy jo useimmista selaimista, mutta uudempi Silverlight ei vielä ole saavuttanut samanlaista tukea. Palveluntarjoajan täytyykin ottaa huomioon, että kaikki eivät halua tai voi asentaa sovellusta voidakseen käyttää tarjottua sovellusta.

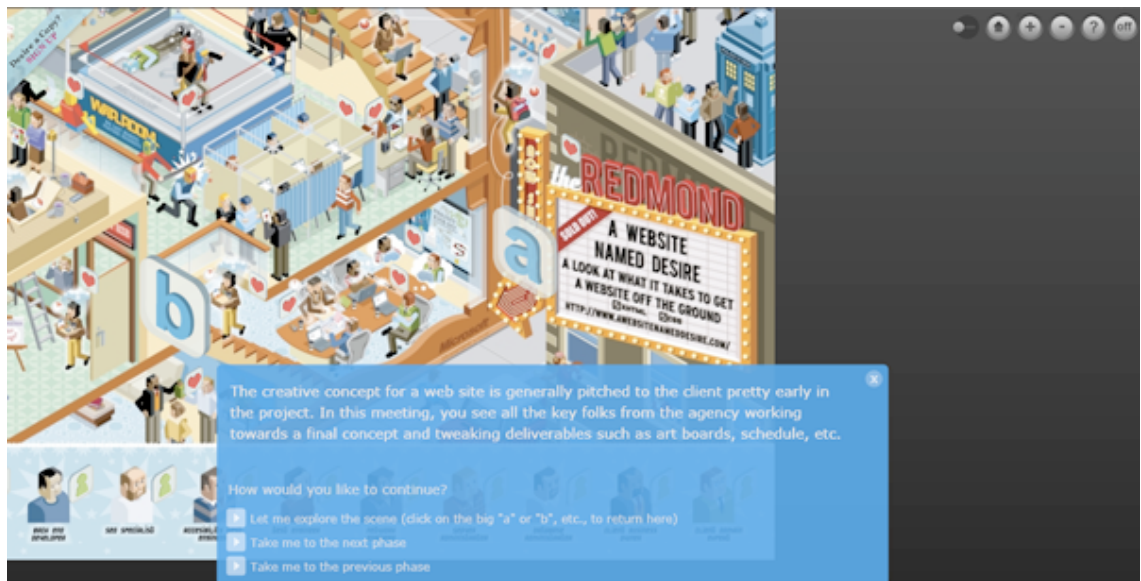
Adoben ja Microsoftin tekniikoiden etuna on parempi kontrolli käyttöliittymän elementteihin ja visuaalisuuteen. Mikäli sovelluksen tarkoitus on olla visuaalisesti näyttävä ja käyttökokemuksen samanlainen kaikissa selaimissa, kannattaa Ajaxin sijaan valita yksi näistä tekniikoista.



Kuva 3 Adobe Flexillä toteutettu sovellus, jossa voi selata kuvia niihin liitettyjen tagien perusteella. Ryhmät liikkuvat ruudulla animoidusti ja toisiinsa liittyvät elementit yhdistyvät viivoilla toisiinsa. (<http://taggraph.com>)



Kuva 4 Lovely Charts on Flexillä toteutettu sovellus, jolla voi luoda erilaisia kaavioita. Ulkoasultaan kaunis sovellus on myös helpokäyttöinen ja ymmärrettävä. (<http://my.lovelycharts.com>)



Kuva 5 A Website Named Desire kuvaa web-sivun suunnitteluprosessia visuaalisen esityksen avulla. Hiirellä voi raahata kuvaa ja tarkentaa haluamiinsa kohtiin, inforuutujen kohdalla aukeaa tekstilaatikoita, joissa on selitetty termejä ja työvaiheita. Sovellus on toteutettu Silverlightilla. (<http://www.visitmix.com/labs/descri/awebsitenameddesire>)

(Microsoft, Adobe Flex, Adobe Flash Player Statistics)

2.5 Yhteenveto

RIA-tekniikoilla voidaan toteuttaa vuorovaikutteisia ja rikkaita käyttöliittymiä, joihin voidaan toteuttaa korkean tason vuorovaikutustekniikoita. RIA-sovellukset saadaan

vastaamaan nopeasti, koska laskentaa voidaan suorittaa asiakaspuolella.

Parhaimmillaan RIA-sovellusta on vaikea erottaa työpöytäsovelluksesta. RIA-sovellusten edut ollaan kuitenkin vasta löytämässä, tekniikka yleistyy hitaasti ja sovellukset eivät ole vielä kaikkien saatavilla. Lisäksi pitää ratkaista erilaisia ongelmia, kuten kuinka hakukoneet löytävät dynaamisen tekstin ja kuinka sisällön saavutettavuus taataan.

3. Käyttöliittymä

3.1 Käyttöliittymäsuunnittelun perusteita

Käyttöliittymän suunnittelussa on hyvä noudattaa joitakin hyväksi todettuja periaatteita, jotka pätevät sekä web-sivustojen ja -sovellusten että työpöytäsovellusten kohdalla.

Silverin mukaan ensimmäinen sääntö on, että muoto seuraa funktiota. Vaikka hyvän suunnittelun tulos on ulkoasultaan kaunis, tulee muodon palvella ensisijaisesti tarkoitusta. Käyttöliittymän elementtien täytyy olla helposti löydettävissä, loogisessa järjestyksessä ja niiden ulkoasun tulee kertoa kuinka ne toimivat. On hyvä myös pitää mielessä, että kaikki mikä ei ole tarpeellista voidaan jättää pois.

Hyvin suunniteltu käyttöliittymä seuraa käyttäjän mentaalisia malleja. Tietyt käyttöliittymät sisäistetään jo lapsuudessa. Jokainen oppii nopeasti että nappuloita painetaan, katkaisinta käännetään ja kahvaa väännetään. Käyttäjät odottavat, että opituilta malleilta näyttävät toiminnot myöskin toimivat kuten esikuvansa.

Myös kontrollien sijainnilla on merkitystä. Hiirtä käyttäessä käyttäjän on helpompi ja nopeampi osua ruudun reunoilla kuin keskellä sijaitseviin objekteihin. Tämä perustuu siihen, että käyttäjä voi siirtää kursoria nopeasti tietäen että hiiren liike pysähtyy näyttölaitteen reunoihin. Keskellä sijaitsevia objekteja lähestyttäessä on liikettä hidastettava, jotta kursori ei liikkuisi kohteen ohi. Suuriin objekteihin on ymmärrettävästi helpompi osua kuin pieniin.

Käyttöliittymässä on tärkeää säilyttää yhtenäisyys. Samat tehtävät täytyy voida suorittaa

samalla tavalla kaikkialla sovelluksessa, ja elementtien ulkoasun täytyy pysyä samana. Lisäksi sovellus ei saisi antaa käyttäjän tehdä vakavia virheitä, varsinkaan siten että niitä ei ole mahdollista korjata. (Silver 2005, sivut 29-44)

3.2 RIA-käyttöliittymän erityispiirteet

Koska useimmat RIA-sovellukset ovat hybridejä web-sivuston ja työpöytäsovelluksen välillä, on suunnitteluprosessissa yhdistettävä sekä informaatioarkkitehtuuria että toimintojen käyttöliittymäsuunnittelua.

Perinteisillä hypertekstin esittämiseen tarkoitetuilla web-sivustoilla käytettävyyšnäkökulma on informaatioarkkitehtuurissa. Suunnittelijan täytyy miettiä mitä tietoa käyttäjät hakevat, miten se järjestetään ja miten se on helpointa löytää.

Web-sivustoilla vuorovaikutuselementit rajoittuvat linkkeihin ja lomakkeisiin. Käyttökokemus perustuu sivutukseen ja sivuilla liikkumiseen. Käyttäjät liikkuvat sivustolla linkkien ja selaimen Taakse- ja Eteen-painikkeiden avulla ja käyttävät mahdollisesti hakua löytääkseen tietoa.

RIA-sovelluksessa käytetään aivan erilaista vuorovaikutteisuutta ja käyttäjä voi suorittaa hyvinkin monimutkaisia toimintoja. Käyttäjä voi usein syöttää, järjestää ja tallentaa tietoa ja liikkua edestakaisin sovelluksen sisällä poistumatta käyttämältään sivulta. (Spool 2007, Perfetti)

RIA-sovelluksissa vasteajat ovat erilaisia kuin työpöytäsovelluksissa. Koska sovellukset toimivat verkon yli, vaikuttaa siirtonopeus suoraan sovelluksen nopeuteen. Mikäli tietoa täytyy hakea tai tallentaa palvelimelle, on odotusaika yleensä pidempi kuin paikallisesti ajettavassa sovelluksessa. Nopeilla yhteyksillä vasteajat saattavat olla huomaamattoman lyhyitä, mutta suunnittelussa on otettava huomioon myös hitaat yhteydet. Erityisesti kannettavat päätelaitteet, esimerkiksi kämmentietokoneet tai matkapuhelimet käyttävät useimmiten langatonta verkkoa jonka siirtokapasiteetti ei ole kovin suuri.

Työpöytäohjelmiston siirtämisessä Internetiin pitää ottaa huomioon selainkäyttöisen sovelluksen erityispiirteet, suoraan samanlaisen ohjelman tekeminen ei onnistu. Web-sovelluksen suunnittelu edellyttää aivan uudenlaista näkökulmaa suunnittelijalta. Lisäksi monet web-sovellukset on rakennettu osaksi web-sivustoa, jolloin suunnittelu

poikkeaa huomattavasti työpöytäsovelluksen periaatteista. (Perfetti)

3.3 Käyttöliittymän dokumentointi

Suunnittelussa käytetään apuna erilaisia kuvaustekniikoita, joilla esitellään sovelluksen toimintaa. RIA-sovellusten kuvaamisessa on kuitenkin erilaiset lähtökohdat kuin web-sivujen tai työpöytäsovellusten. RIA-sovelluksessa elementeillä saattaa olla useampia erilaisia tiloja, lisäksi linkit ja painikkeet voivat aiheuttaa joko komponentin uudelleenlatauksen, koko sivun lataamisen uudestaan tai ladata kokonaan eri sivun. (Cecil 2007)

3.3.1 Konseptitaulut

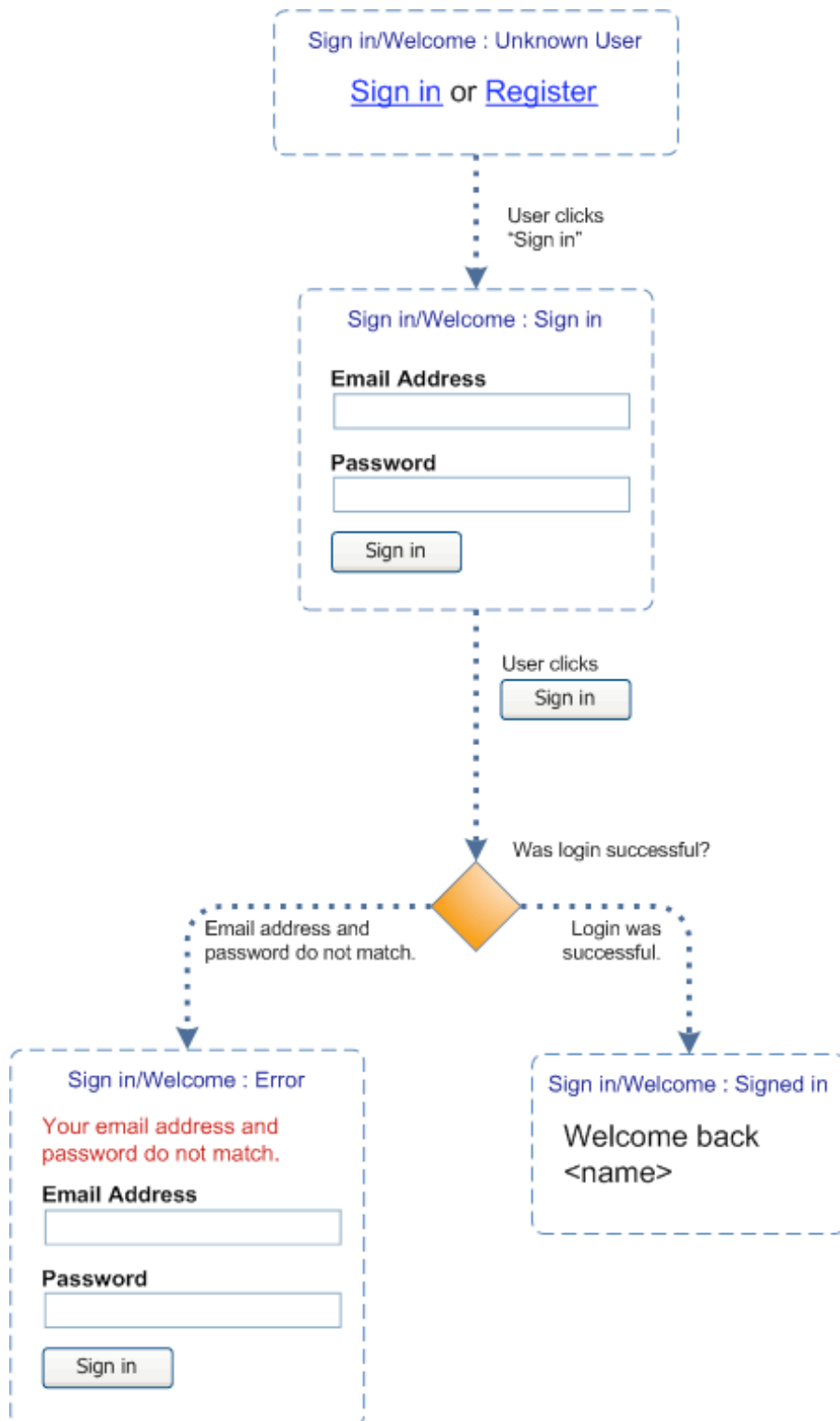
Konseptitaulut (engl. concept boards) eli tarinataulut (engl. storyboards) esittelevät web-sivuston sivujen sisällön. Niissä kuvataan sivun elementit ja niiden asettelu sivulla. Konseptitaulut voivat olla karkeasti käsin piirrettyjä luonnoksia tai kuvata tarkasti sivun visuaalisia elementtejä. Suunnittelu etenee luonnoksista yksityiskohtaisiin rautalankamalleihin ja visuaalisiin konseptitauluihin. RIA-sovellusten suunnittelussa konseptitauluja voi käyttää kuvaamaan eri sivujen sisältöä mikäli sivuja on useita, mutta eri elementtien eri tilojen ja niiden ehtojen kuvaaminen saattaa osoittautua mahdottomaksi. (Cecil 2007, Jadav 2003)

3.3.2 Flowframes

Koska perinteisillä esitysmalleilla on vaikea kuvata RIA-sovelluksen komponenttien toimintaa, Richard F. Cecil esittelee kirjoituksessaan monipuolisemman tavan dokumentoida RIA-sovelluksen toimintoja. Hänen mallissaan näytetään erillisen komponentin eri tilat rautalankamallina vuokaavion sisällä (kuva 1).

Komponentti esitetään rajatun ja nimetyn alueen sisällä, nimi koostuu komponentin nimestä ja tilasta. Käyttäjän toiminnot ja ehdot esitetään esitetään haaroina, joissa näytetään komponentin uusi tila. Toiminnot, jotka lataavat ainoastaan komponentin uudestaan yhdistetään katkoviivan ja nuolen avulla, koko sivun laataamista vaativat toiminnot yhdistetään yhtenäisellä nuolivivalla. Mallia voidaan käyttää myös ilman rautalankamallia kuvaamalla komponenttia erityisellä symbolilla, johon on merkitty

komponentin nimi ja tila. Tällöin voidaan piirtää monimutkaisempia vuorovaikutuskaaviota. (Cecil 2007)



Kuva 6 R. Cecil: wireflow (Cecil 2007)

3.3.3 Prototyypit

Prototyyppi (engl. prototype, mock-up) on web-suunnittelussa toiminnallinen esimerkki koko sivustosta. Prototyypissä on toteutettuna sivuston kaikki erilaiset sivupohjat, grafiikat ja navigaatio sekä tärkeimmät toiminnallisuudet. Prototyypivaiheessa kannattaa myös tehdä ensimmäisiä käytettävyydestestauksia. RIA-sovellusten kohdalla toiminnallisen prototyypin toteuttaminen vaatii yleensä teknistä osaamista: RIA-sovelluksen prototyyppi saattaa olla vaikea toteuttaa web-sivuston prototyypin tekemiseen tarkoitettulla työkalulla, koska siinä on paljon toiminnallisuutta ja mahdollisesti vähän tai ei lainkaan eri sivuja.

(Cecil 2007, Jadav 2003)

3.4 Käyttöliittymän elementit

Käyttöliittymissä käytetään yleensä enimmäkseen vakiintuneita vuorovaikutuselementtejä, kuten esimerkiksi pudotusvalikkoja, painikkeita, valintaruutuja ja tekstilaatikoita. Kauan käytössä olleilla elementeillä on ominaispiirteensä, joita tulisi aina noudattaa. Esimerkiksi tekstilaatikossa ei koskaan saa näyttää tekstiä, joka ei ole muokattavissa.

Usein myös törmää tilanteeseen, jossa valintapainikkeita on käytetty väärin, esimerkiksi laitettu käyttöliittymään yksi radiopainike tilanteessa, jossa käyttäjä voi valita joko kyllä tai ei. Radiopainikkeita käytetään kuitenkin vain silloin, kun ryhmästä valintoja on valittava yksi. Valintaruutu on oikea elementti silloin kun pienestä ryhmästä valintoja voidaan valita useita tai ei yhtään, tai yksi valinta voidaan asettaa joko päälle tai pois päältä.

Lisäksi on paljon suosituksia, joita voi käyttää apuna suunnittelussa. Esimerkiksi pudotusvalikoissa Microsoft suosittaa käyttämään tiettyjä sanaluokkia: jos valikon otsikko on substantiivi, tulisi vaihtoehtojen olla verbejä ja toisinpäin. Oletetaan että otsikkona on verbi Lisää, jolloin vaihtoehtoina käytetään substantiiveja, esimerkiksi Taulukko, Kenttä tai Symboli. (Silver 2005, sivut 115-136)

Web-sovellukset ovat myös kontrollien osalta hankalia, koska ne ovat luonteeltaan sekä web-sivuja että sovelluksia. Selainkäyttöiset sovellukset ovat olleet muodissa vasta

vähän aikaa ja käyttömahdollisuuksia on monia, joten yleisesti käytössä olevia konventioita ei toistaiseksi ole muodostunut. Esimerkiksi painikkeita suunnitelmassa pitää päättää ovatko ne työpöytäsovelluksen tyyllisiä komentopainikkeita vai web-sivuilta tuttuja linkkejä.

Työpöytäohjelmistoissa käytössä ovat lähes aina samanlaiset harmaat ikonit ja menut, joita käyttäjä osaa odottaa ja olettaa niiden myös toimivan tietyllä tavalla. Web-sovelluksessa suunnittelijalla on enemmän vapauksia päättää miltä sovellus näyttää, ja vaikutteita voi ottaa sekä web-sivuista että työpöytäsovelluksista. (Spool 2007, Perfetti)

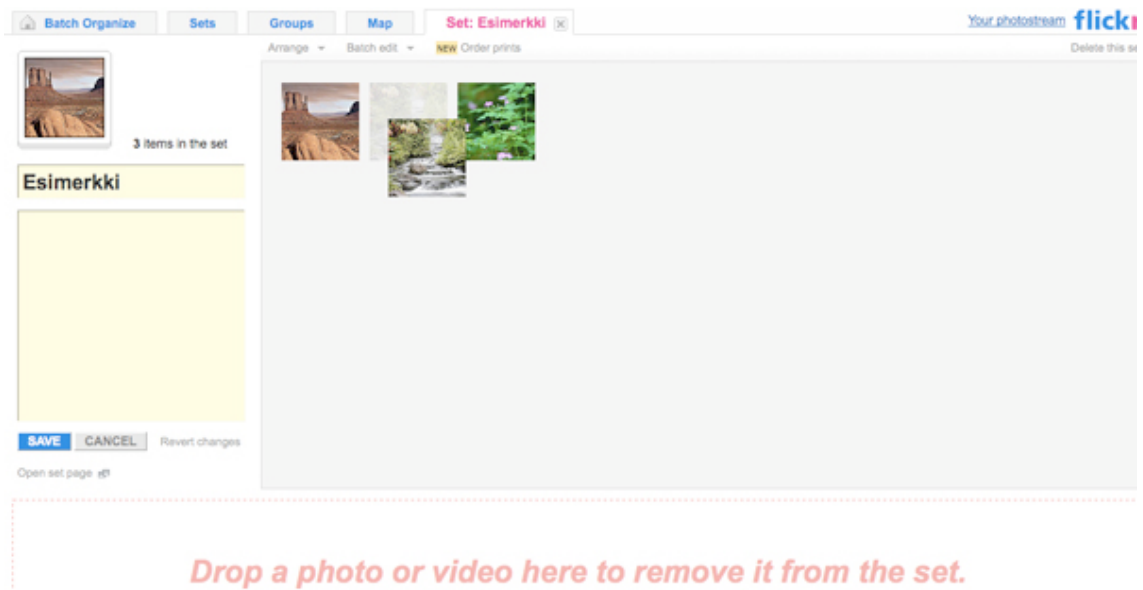
Monet työpöytäsovelluksissa vakiintuneet toiminnot ovat käytännöllisiä tai välttämättömiä myös RIA-sovelluksissa. Koska käyttäjät eivät aina mene tarkoitettua polkua pitkin tai anna oikeata dataa, pitäisi tarjota mahdollisuus kulkea taaksepäin. Mahdollisuus peruuttaa juuri suoritettu toiminto on yleensä tarpeellinen ominaisuus myös web-sovelluksessa, mutta sen toteuttaminen saattaa olla vaikeaa. Miten toimia selaimen peruuta-painikkeen kanssa on haaste sekä suunnittelijalle että tekniselle toteutukselle. (Spool 2007)

Tallennusmahdollisuutta ei web-sovelluksissa usein ole. Käyttäjä saattaa kuitenkin haluta keskeyttää sovelluksen käyttämisen jatkaakseen myöhemmin, joten sovelluksen tilan tallentamismahdollisuus saattaa parantaa käytettävyyttä huomattavasti. Myös hakujen ja useimmin suoritettujen toimintojen tallentaminen voi tehdä sovelluksen käyttämisestä tehokkaampaa ja nopeampaa. (Willemsen 2006)

Kuten muissakin sovelluksissa, toimintojen tulisi olla tunnistettavia ja sovelluksen antaa asianmukaista palautetta käyttäjän toimista. Yksi suunnitteluhaasteista on se, että käyttäjät ovat tottuneet siihen miten web-sivustot käyttäytyvät ja saattavat hämmentyä sovelluksen toimiessa eri tavalla kuin he odottavat.

Monet hienostuneemmat työpöytäsovelluksista tutut vuorovaikutustekniikat, kuten raahaus, vierittäminen vetämällä tai monivalintatoiminnot eivät ole web-sovelluksessa odotettavissa, joten käyttäjät eivät välttämättä huomaa niitä ellei heille tarjota selkeitä opasteita. Hienovaraisena vihjeenä voi toimia esimerkiksi kursorin muutos, mutta yksinään se ei useimmiten vielä riitä. On myös oleellista että käyttäjä huomaa kun jotain tapahtuu, eli sovelluksen antama palaute käyttäjän toimiin on tarpeeksi selkeää ja ymmärrettävää.

(Porter, Spool)



Kuva 7 Flickr.com:issa kuvia lisätään, poistetaan ja järjestetään raahaamalla

Sovelluksen toiminnoissa on hyvä käyttää hyväksi myös todellisen maailman esimerkkejä, mutta unohtamatta sitä, että interaktiivinen ympäristö tarjoaa tiettyjä etuja verrattuna käsinkosketeltaviin esineisiin.

Menestyvässä sovelluksessa täytyy ottaa huomioon myös käyttäjien aikaisemmat kokemukset. Mikäli on oletettavaa että käyttäjät ovat tottuneet käyttämään jotain tiettyä ohjelmistoa, on järkevää käyttää samankaltaisia toimintoja joihin käyttäjät ovat tottuneet. (Spool)

3.5 Tila ja tilan vaihtuminen

Koska RIA-sovellukset eivät usein perustu sivuihin, saattaa elementeillä tai koko sivulla olla erilaisia tiloja (engl. state).

Elementin tila vaihtuu käyttäjän toimien seurauksena, jolloin elementti saa eri sisällön, eri toimintoja tai erilaisen ulkoasun. Yksinkertainen perinteinen esimerkki elementin tilasta on esimerkiksi navigaation linkki, joka näyttää erilaiselta mikäli käyttäjä on kyseisellä sivulla. RIA-sovelluksen komponentit voivat olla monimutkaisia, joten kaikkia toimintoja ei välttämättä voida näyttää kerralla, vaan ne tuodaan esiin tarvittaessa. (Cecil 2007)

Tilan vaihtuessa käyttäjälle on annettava selkeä visuaalinen palaute siitä että jotain juuri tapahtui. Koska RIA-sovelluksissa voidaan tehdä hyvin huomaamattomia osittaisia uudelleenlatauksia, ne jäävät helposti käyttäjältä huomaamatta. Taukoa käytetään yleisesti tilan vaihtumisen indikaattorina, käyttöliittymässä tämä tehokeino tapahtuu animaation välityksellä. Elementti voidaan esimerkiksi häivyttää näkyvistä ja hivuttaa uusi sisältö näkyviin.

Äkillinen näkymän vaihtuminen pakottaa käyttäjän orientoitumaan uuteen näkymään, mikä saattaa pidemmän päälle olla rasittavaa. Animoitu näkymän vaihto auttaa käyttäjää ymmärtämään edellisen ja nykyisen näkymän suhteen toisiinsa ja pehmeä siirtyminen ei häiritse käyttäjää yhtä paljon kuin näkymän äkillinen vaihtuminen. Lisäksi animaatio auttaa käyttäjää pysymään keskittyneenä suoritettavaan tehtävään sekä pitää katseen fokuksen oikeassa paikassa, jotta käyttäjän ei tarvitse etsiä oikeaa kohtaa uudelleen.

Vaikka animaatio voi oikein käytettynä parantaa käyttökokemusta ja tehdä käyttäjän olon miellyttäväksi, väärin käytettynä se voi pilata kaiken. Liian hidaskäyttö tai tarpeeton animaatio ärsyttää käyttäjiä ja pahimmassa tapauksessa ajaa heidät kokonaan pois. (Follett 2007)

3.6 Sivut

Työpöytäsovellukset ja suurelta osin myös web-sovellukset perustuvat aivan erilaiseen toimintamalliin kuin web-sivustot. Käyttökokemus ei perustu sivutukseen vaan sivun ja sen elementtien erilaisiin tiloihin. Toisaalta sivumallin käyttö web-sovelluksessa on mahdollista, ja selaimen toimintojen takia se on yleensä ainakin otettava huomioon. Yksisivuiset RIA-sovellukset muistuttavat käyttökokemukseltaan enemmän työpöytäohjelmistoa, kun taas useita sivuja käyttävät sovellukset saavat web-sivuston piirteitä. (Spool 2007, Perfetti)

Yksisivuisessa sovelluksessa käyttäjälle näytetään kerrallaan vain ne toiminnot ja informaatio mitä hän sillä hetkellä tarvitsee, ja pyydetty uusi informaatio ladataan osittaisena päivityksenä sivulle. Siinä missä hypertekstin eli perinteisten web-sivujen lukeminen tapahtuu lineaarisesti liikkumalla sivulta toiselle ja takaisin, on yksisivuisen sovelluksen käyttö useimpien työpöytäsovellusten tapaan epälineaarista. Esimerkiksi

artikkelien lukeminen lineaariseen tapaan tuntuu loogiselta, mutta mikäli informaatiota on paljon ja toimintoja erilaisia, alkaa lineaarinen lähestymistapa saavuttaa rajojaan. (Willemsen 2006)

Esimerkiksi monimutkaisia toistuvia hakuja sisältävä sivusto hyötyy epälineaarisuudesta. Joka kerta hakuehtoja vaihtaessa käyttäjän täytyisi odottaa koko sivun latautuminen uudestaan. Lisäksi useiden hakujen tapauksessa kulkeminen edestakaisin hakusivun ja tulosten välillä muuttuu äkkiä turhauttavaksi, varsinkin jos kaikki ehdot joutuu joka kerta täyttämään uudestaan.

Yksisivuisessa sovelluksessa käyttäjä näkee hakuehdot koko ajan ruudulla, ja koska sivun vaihtuminen ei häiritse vuota, on helpompi nähdä miten hakuehtojen muutos vaikuttaa hakutuloksiin. Koska immersio säilyy, käyttäjä pysyy myös varmemmin keskittyneenä juuri suoritettavaan tehtävään. Myös tulosten vertailu on helpompaa mikäli yksityiskohdat näytetään suoraan hakutuloksessa esimerkiksi hiiren painalluksesta avautuvana osiona eikä omana sivunaan. Virheilmoitukset ovat myös vähemmän häiritseviä, kun annetaan välitön palaute viemättä käyttäjää pois varsinaiselta sivulta. (Willemsen 2006)

3.7 Yhteenveto

Suunnittelun alussa täytyy päättää tehdäänkö sovelluksesta yksisivuinen työpöytäsovellusta muistuttava web-sovellus vai sisältääkö sivusto useampia sivuja joilla käytetään RIA-komponentteja. Yksisivuiset sovellukset palvelevat parhaiten monimutkaisia käyttötapauksia kun taas monisivuiset sovellukset soveltuvat mainiosti alkuperäiseen käyttötarkoitukseensa eli hypertekstin esittämiseen.

Jotta sivusto todella hyötyisi RIA-komponenteista, on ne suunniteltava niin, että käyttäjät tunnistavat niiden käyttötarkoituksen ja huomaavat kun jokin osa sivusta päivittyy. On myös pidettävä mielessä, että mitä monimutkaisempi sovellus, sitä vaikeampi sitä on käyttää. Niinpä hienojen ominaisuuksien lisäämisen tulee olla perusteltua, ei itsetarkoitus. Hyvin suunniteltu sovellus lisää immersiota ja helpottaa käyttäjän tehtävien suorittamista.

4. Käytettävyys

4.1 Mitä on käytettävyys

Käytettävyydeltään hyvä sovellus on käyttöliittymältään läpinäkyvä eli käyttäjä pystyy keskittymään päämääräänsä sen sijaan että joutuisi miettimään miten sen voi saavuttaa, Huolellisesti suunnitellussa sovelluksessa tehtävien suorittaminen tuntuu luontevalta ja helpolta. RIA-tekniikoiden avulla voidaan toteuttaa älykkäitä käyttöliittymiä sekä saumatonta ja välitöntä vuorovaikutteisuutta, jolloin käyttäjien tarpeita voidaan paremmin huomioida ja tehdä käyttökokemuksesta miellyttävämpi. (Mullet 2003)

Tunnettu käytettävyyden tutkija Jakob Nielsen jakaa käytettävyyden viiteen eri osa-alueeseen: käytettävyydeltään hyvä sovellus on helppo oppia, tehokas, helposti muistettava, mahdollisimman virheetön sekä miellyttävä käyttää. (Nielsen 1993, sivut 27-33)

4.2 Käytettävyyden arvioiminen

Käytettävyyttä mitataan yleensä tiettyihin käyttäjiin ja tehtäviin kohdistetulla käyttäjätutkimuksella. Käyttäjätutkimukseen valitaan mukaan sovelluksen oletettuja tulevia käyttäjiä erilaisista käyttäjäryhmistä. Käyttäjille annetaan erilaisia tehtäviä ja katsotaan kuinka helposti ja tehokkaasti he pystyvät ne suorittamaan.

Käytettävyyden mittaamista varten on olemassa myös erilaisia heuristisia malleja, joiden avulla käytettävyyttä voidaan arvioida. Heuristisessa arvioinnissa käydään sovelluksen käyttöliittymä läpi käyttäen apuna listaa käytettävyyden periaatteista ja tarkistetaan, että käyttöliittymä noudattaa niitä. (Nielsen 1993, sivut 115-117)

4.3 Heuristinen malli

Heuristinen malli on kokoelma periaatteita, joiden tulisi vähintäänkin toteutua käyttöliittymässä, jotta sitä olisi helppo ja miellyttävä käyttää. Tässä esitellään Nielsenin heuristinen malli ja miten sitä voidaan soveltaa RIA-sovelluksiin. (Wikipedia: Heuristic Evaluation 2009, Nielsen, sivut 115-155)

4.3.1 Tilan näkyminen käyttäjälle

Sovelluksen tulisi antaa käyttäjälle jatkuvaa palautetta tilastaan, jotta käyttäjä tietää mitä tapahtuu.

RIA-sovelluksissa voidaan helposti toteuttaa latausanimaatioita, jotka kertovat käyttäjälle, että se prosessoi tietoa tai hakee dataa. Mikäli jonkin toiminnon suorittaminen vaatii useita askeleita, pitäisi käyttäjälle kertoa missä kohtaa hän tehtävän suorituksessa on ja paljonko on jäljellä. Näin käyttäjä voi helposti arvioida kuinka paljon työtä tehtävän suorittaminen vaatii. RIA-sovelluksessa voidaan tallentaa dataa asiakaspuolelle ja sallia vaadittavien askelten suorittaminen missä järjestyksessä tahansa. Kaikissa tapauksissa näytettävän tilan tulisi liittyä käyttäjän suorittamaan tehtävään, ei sovelluksen tekniseen tilaan.

4.3.2 Vastaavuus tosimaailman kanssa

Sovelluksessa tulisi käyttää käyttäjälle tuttua termejä ja käsitteitä. Käyttäjät eivät välttämättä tiedä mitä esimerkiksi raahaaminen tarkoittaa, saati sitten ymmärrä tekniikkaan liittyviä termejä kuten tietokanta. Tällaisten termien näyttämistä käyttäjälle tulisi välttää.

Tieto tulisi esittää luonnollisessa ja loogisessa järjestyksessä. Vaikka RIA-tekniikoilla voidaan toteuttaa helposti metaforia, täytyy olla varma, että ne parantavat käytettävyyttä. Mikäli metafora ei täysin vastaa tosimaailman vastinettaan, se on hyödytön. Sekä tieto että toiminnot tulisi järjestää niin että ne vastaavat käyttäjän päämääriä ja tehtäviä.

4.3.3 Käyttäjän kontrolli ja vapaudet

Käyttäjät tekevät usein vääriä painalluksia, joten uudesta tilasta on aina oltava selkeä poistumistie tai mahdollisuus peruuttaa suoritettu toiminto. Käyttäjät ovat tottuneet käyttämään selaimen Takaisin-painiketta päästäkseen takaisin edelliseen tilanteeseen ja olettavat sen toimivan myös RIA-sovelluksessa. Sovellus on usein mahdollista toteuttaa niin, että Takaisin-painike toimii loogisesti. Mikäli sovellus sisältää monimutkaisia toimintoja ja painikkeen käyttäminen voi esimerkiksi hävittää syötettyjä tietoja, kannattaa avata sovellus uuteen ikkunaan, josta on piilotettu selaimen toiminnot. Tällöin

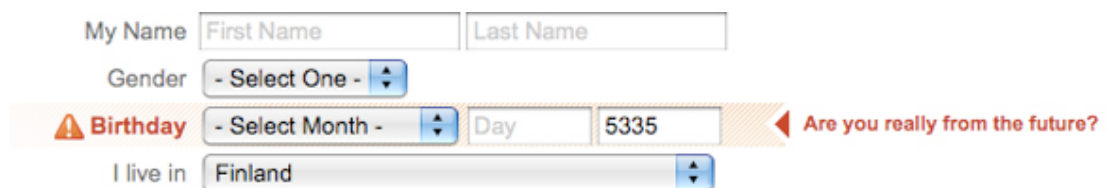
pitää kuitenkin tarjota käyttäjälle myös paluutie takaisin selaimen.

4.3.4 Yhdenmukaisuus ja standardit

Komentojen ja toimintojen pitäisi olla yhdenmukaisia koko sovelluksessa ja seurata myös vakiintuneita konventioita. Tämä koskee terminologiaa, asetelua, värejä ja käyttäytymistä. Koska RIA-sovellukset ovat vielä suhteellisen uusia, konventiot ja parhaat toimintatavat ovat vielä kehitysvaiheessa.

4.3.5 Virhetilanteiden välttäminen

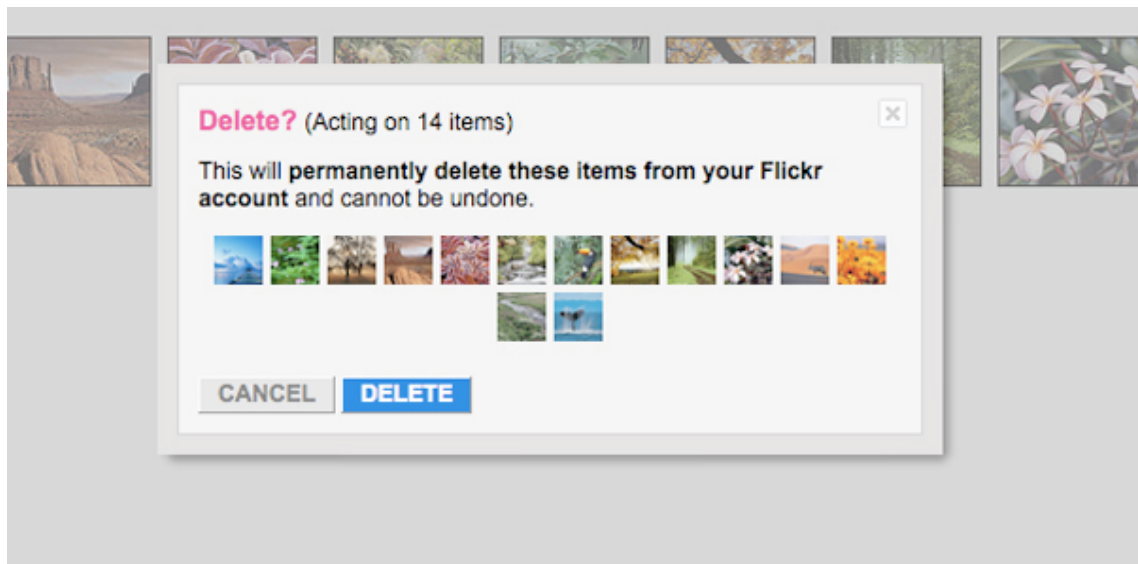
Huolellinen suunnittelu estää virhetilanteiden syntymisen, mutta virheen tapahtuessa virheilmoituksen täytyy olla myös käyttäjäystävällinen. RIA-sovelluksissa yleisiä käyttöliittymäelementtejä ovat esimerkiksi automaattisesti täydentyvät kentät, jotka vähentävät virhetilanteita. Syötekentissä pitäisi myös hyväksyä erimuotoisia syötteitä.



The image shows a registration form with the following fields: "My Name" (split into "First Name" and "Last Name"), "Gender" (dropdown menu), "Birthday" (dropdown for month, text input for day, and a text input containing "5335"), and "I live in" (dropdown menu showing "Finland"). A red error message "Are you really from the future?" is displayed next to the "Day" input field, which is highlighted with a red border. A warning icon is also present next to the "Birthday" label.

Kuva 8 Yahoo-tunnuksen rekisteröintilomake hyväksyy vuosiluvun syötteeksi myös yksi- tai kaksinumeroisia lukuja, jotka se korjaa 1900-luvun vuosiksi, esimerkiksi 85 muuttuu vuodeksi 1985. Täysin mahdoton syöte aiheuttaa välittömän virheilmoituksen.

Toimintoja, jotka saattavat vahingossa käytettynä aiheuttaa todella ikäviä tilanteita, kuten "Poista kaikki", pitäisi välttää tai ainakin sijoittaa ne erilleen yleisesti käytetyistä toiminnoista ja mahdollisesti vain järjestelmänvalvojen näkyville. RIA-sovelluksissa virhetilanteen tiedot voi helposti lähettää ja tallentaa, ja päivitysten tekeminen on helppoa ja näkymätöntä käyttäjälle, joten toistuvat virheet on helppo korjata aiheuttamatta ylimääräistä harmia.



Kuva 9 Flickr.comissa poistotoiminto löytyy alavetovalikosta, joten se täytyy erikseen etsiä.

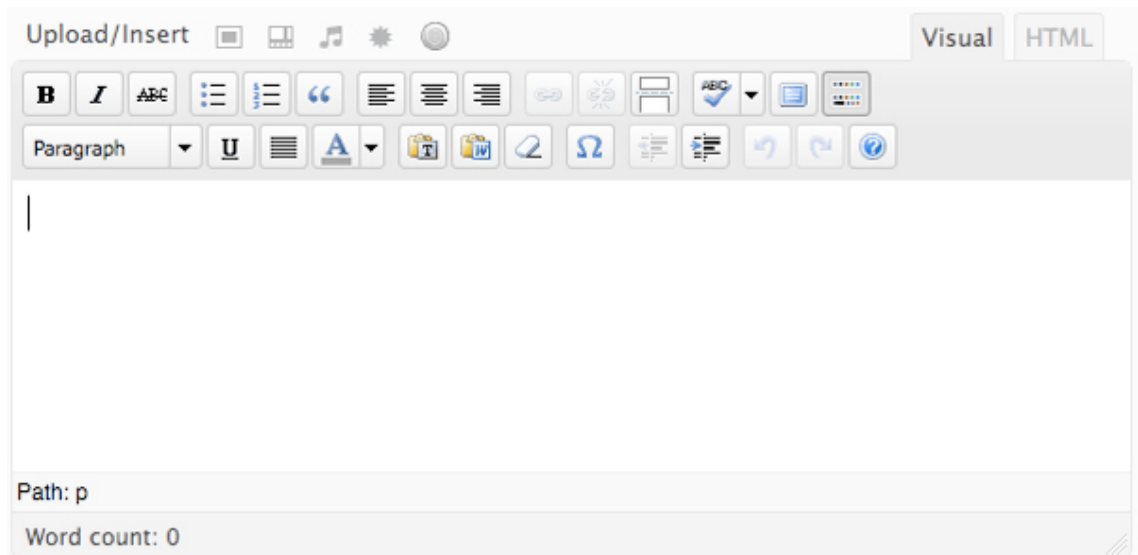
Vahinkotilanteiden varalta sovellus vielä varmistaa että toiminto on tarkoituksellinen ennen sen suorittamista.

4.3.6 Tunnistaminen, ei muistaminen

Ideaalitilanteessa käyttäjän ei tarvitse muistaa tietoa sovelluksen tilasta toiseen. Toimintojen ja vaihtoehtojen tulisi olla näkyvillä ja ohjeiden helposti saatavilla. Piilotettujen toimintojen etsiminen ja tuominen näkyviin hidastaa käyttäjän tehtävien suorittamista ja aiheuttaa turhautumista. Käyttöliittymän tulisi tarjota visuaalisia vihjeitä siihen miten käyttäjä saa tehtävänsä suoritettua, ei piilotta vaihtoehtoja.

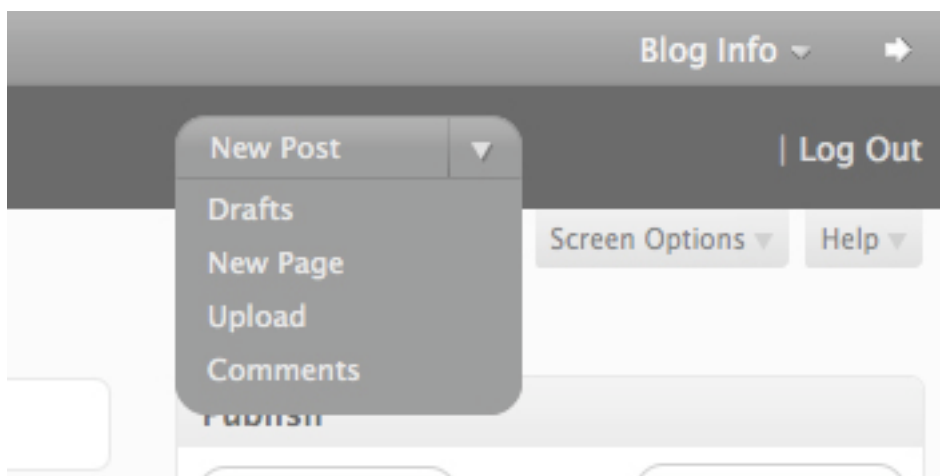
4.3.7 Joustavuus ja käytön tehokkuus

Edistyneille käyttäjille tulisi tarjota mahdollisuus käyttää pikanäppäimiä ja -komentoja, ja lisäksi muokata niitä sopimaan omiin tarpeisiinsa. Aloittelevilta käyttäjiltä nämä edistyneet toiminnot pitää kuitenkin piilottaa. Myös kenttien automaattinen täyttyminen perustuen esimerkiksi useimmin käytettyihin vaihtoehtoihin nopeuttaa käyttöä.



Kuva 10 Wordpressin tekstinkäsittelyosiossa voi käyttää halutessaan pikanäppäimiä.

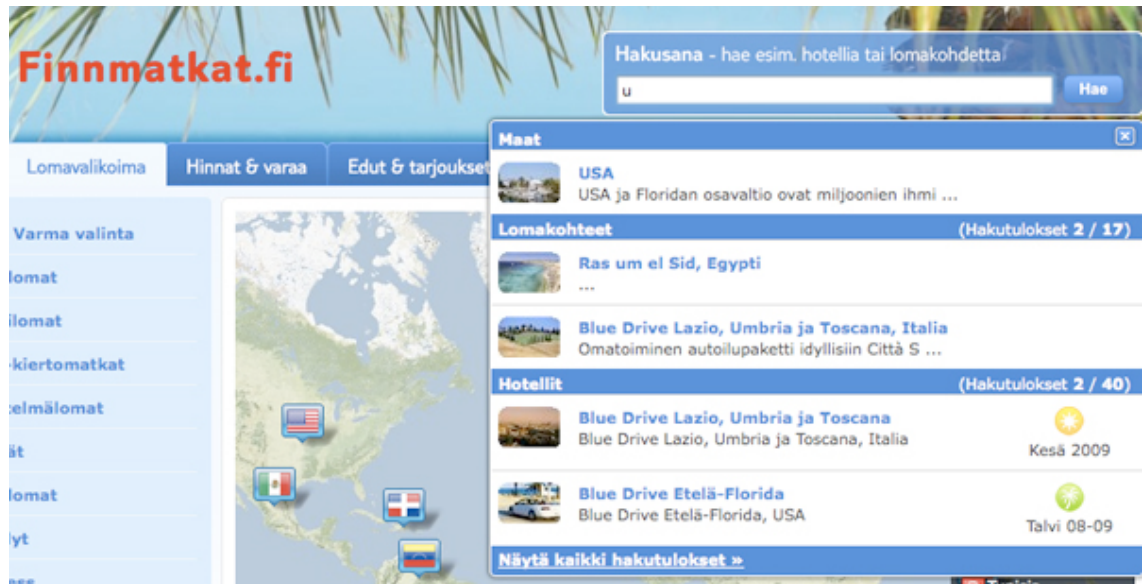
Pikanäppäinkomennot näytetään työkaluvihjeessä, joka aukeaa kun vie hiiren painikkeen päälle. Esimerkiksi lihavoinnin pikakomento on Ctrl / Alt + Shift + B.



Kuva 11 Wordpressin ylälaudassa on pikalinkit yleisimmiin suoritettuihin tehtäviin, joten kokeneen käyttäjän ei tarvitse käyttää aikaa useiden sivujen selaamiseen toiminnon suorittamiseksi.

Myös kirjanmerkkien käyttö pitäisi tehdä mahdolliseksi, joko selaimen puolelta tai toteuttamalla vastaava ominaisuus sovellukseen, jotta käyttäjä voi palata samaan tilanteeseen helposti uudestaan. RIA-sovellukset voidaan myös toteuttaa helposti personoitaviksi, jolloin käyttäjä voi itse muokata käyttöliittymää haluamansa kaltaiseksi.

Sovelluksessa voidaan myös mitata yleisimpiin tehtäviin kuluva aikaa ja mikäli käyttäjä jatkuvasti suorittaa samoja tehtäviä ilman pikakomentoja, voidaan tarjota mahdollisuutta luoda toiminnolle pikanäppäin tai korostaa tehtävän pikakomento tehostevärillä.



Kuva 12 Finnmatkat-sivustolla haku on toteutettu siten, että hakulaatikon alle aukeaa automaattisesti hakutuloksia sitä mukaa kun käyttäjä kirjoittaa. Tällöin jo yhden kirjaimen kirjoittaminen usein riittää toivottuun hakutulokseen. Aikaa säästää myös se, että sovellus antaa välittömän palautteen mikäli hakutuloksia ei löydy.

4.3.8 Esteettinen ja minimalistinen design

Dialogissa ei saisi olla epärelevanttia tai hyvin harvoin tarvittavaa informaatiota. Ylimääräinen sisältö vie tilaa ja huomioarvoa tarpeelliselta informaatiolta. Jokaisen elementin kohdalla pitäisi miettiä mitä lisäarvoa se antaa palveluntarjoajalle tai käyttäjälle. Jos lisäarvo on olematon, elementti tulisi jättää pois. Yksinkertaisuus ja elegantit ratkaisut ovat yleensä parhaita. Lisäksi minimalismi yleensä rajoittaa myös tiedostokokoa, jolloin latausajat lyhentyvät.

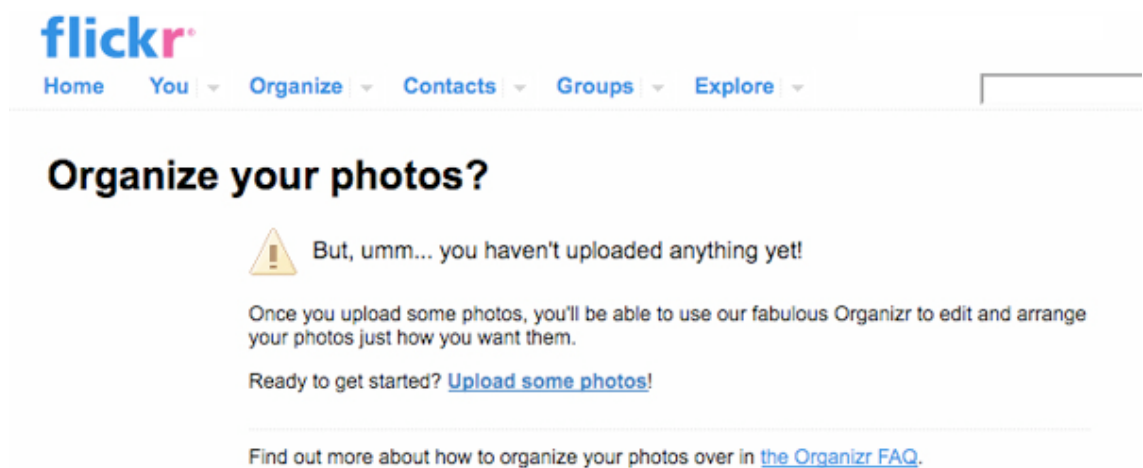
Animaation käyttöä tulisi harkita tarkkaan. Animaatiolla voidaan osoittaa siirtymiä tilasta toiseen ja näin parantaa käytettävyyttä, mutta se tulee toteuttaa siten että se ei hidasta käyttäjän toimintoja. Ylimääräinen liike häiritsee käyttäjää ja katkaisee tehtävän suorituksen. Myös äänen käyttöä tulisi välttää ja rajoittaa se vain niihin tilanteisiin

jolloin se on käytettävyyden kannalta perusteltua.

4.3.9 Auta käyttäjää tunnistamaan virheet ja toipumaan niistä

Virheilmoitukset pitäisi esittää tavallisella kielellä, tarkasti selittää ongelma ja lisäksi ehdottaa ratkaisua. RIA-sovelluksissa virheilmoitukseen voidaan lisätä mahdollisuus myös korjata ongelma tarjoamalla asiaankuuluvat toiminnot tai syötekentät itse ilmoituksessa.

Koska virheilmoitukset voidaan esittää varsinaisen käyttöliittymän päällä, voidaan niissä myös antaa selkeitä ohjeita ja selittää työnkulkuja tarvittaessa. Mikäli ei ole selvää mistä virhe johtui, kannattaa esittää todennäköisimmät vaihtoehdot siihen mitä käyttäjä yritti saavuttaa ja tarjota niihin ohjeita. RIA-sovelluksessa voidaan myös tarjota mahdollisuus ottaa suoraan yhteyttä oikeaan henkilöön avun saamiseksi esimerkiksi reaaliaikaisen keskustelun tai videopuhelun avulla.



Kuva 13 Flickr.com - Virheilmoitus

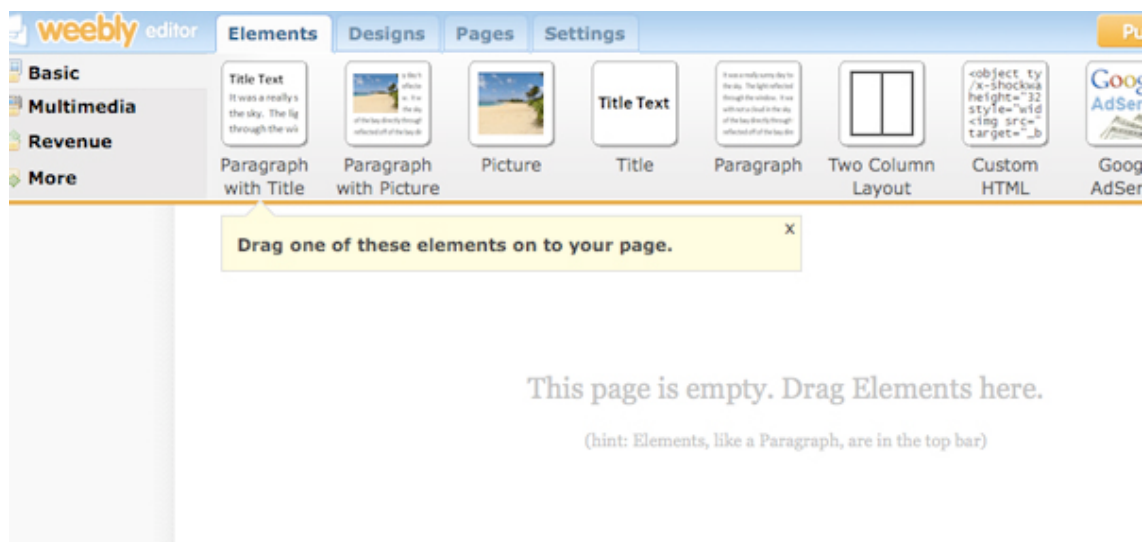
Flickr.com antaa virhetilanteessa selkeän ilmoituksen, jossa kerrotaan virhetilanteen syy, mitä käyttäjä voi tehdä korjatakseen tilanteen sekä tarjoaa linkin oikeaan paikkaan. Lisäksi virheilmoituksessa annetaan linkki ohjeeseen, joka saattaisi olla juuri tässä tilanteessa hyödyllinen käyttäjälle.

4.3.10 Opastus ja ohjeistus

Parhaassa tapauksessa sovellusta osataan käyttää ilman ohjeistusta, mutta sen pitäisi olla kuitenkin tarvittaessa helposti saatavissa. Tiedon pitäisi olla helposti haettavissa ja keskittynyt käyttäjän tehtäviin tarjoten helposti ymmärrettävät askeleet, joilla tehtävä voidaan suorittaa.

RIA-sovelluksessa voidaan apuna käyttää animaatiota tai osoittaa käyttäjälle tarvittavat toiminnot käyttöliittymästä. Käyttöliittymässä itsessään pitäisi myös tarjota käyttäjälle vihjeitä ja ohjeita. Työkaluvihjeet ja paikalliset ohjeistukset auttavat käyttäjää suorittamaan tehtävänsä ilman että huomion kohde täytyy siirtää pois tehtävästä erilliseen ohjeeseen.

(Cato 2001, McMullin 2003)

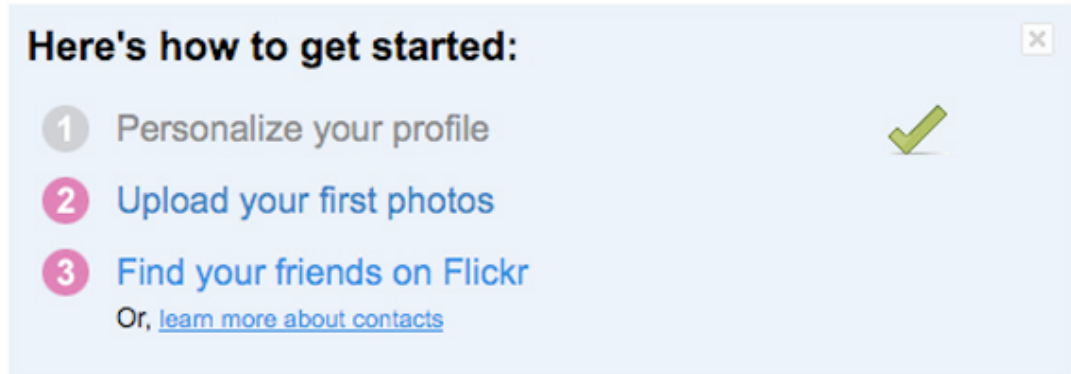


Kuva 14 Weebly.com

Esimerkiksi Weebly-sovelluksessa on hyvin toteutettu ohjeistus. Aloittelevaa käyttäjää ohjataan tutoriaalityyppisesti tutustumaan eri toimintoihin. Ohjelaatikko hyppää esiin alareunasta ja pompahtaa oikeaan kohtaan animoidusti, jolloin katse kiinnittyy oikeaan kohtaan ruudulla. Käyttöliittymässä on myös käytetty työkaluvihjeitä sellaisissa painikkeissa joissa ei ole erikseen selittävää tekstiä.

Weebly-sovelluksessa on aina esillä linkki ohjesivulle, jossa on kattavasti opastettu sovelluksen käyttöön kuvia apuna käyttäen. Kaiken lisäksi ohjesivujen hyödyllisyydestä

kerätään palautetta, jonka voi lähettää yhtä linkkiä painamalla. Ohjesivulla on myös helposti löydettävissä linkki tukipalveluun, jonne voi lähettää kysymyksiä ongelmatilanteissa.



Kuva 15 Flickr opastaa aloittelevia käyttäjiä linkitetyn tehtävälistan avulla

4.4 Käyttäjätestaus

Käyttäjätestaus on tärkein käytettävyyden mittari, sillä se kertoo todellisuuden siitä miten oikeat ihmiset eli tulevat käyttäjät sovellusta käyttävät. Vaikka sovellus olisi hyvin suunniteltu, saattavat suunnittelijat olla väärässä siitä miten sovellusta käytetään autenttisessa tilanteessa. Käyttäjätestaus paljastaa tarkasti mitkä ovat käyttöliittymän ongelmakohdat. Käyttäjätestaus kannattaa aloittaa jo suunnitteluvaiheessa ja testata esimerkiksi prototyyppiä ennen varsinaisen tuotteen valmistusta. Käyttäjätestauksella voidaan myös arvioida kahden eri suunnitteluratkaisun välistä paremmuutta.

Käyttäjätestaukseen valitaan erilaisia käyttäjiä, jotka kuvastavat mahdollisimman tarkasti oletettuja loppukäyttäjiä. Ennen testausta tehdään suunnitelma, jossa määritellään tavoitteet eli mitä tuloksia testauksella halutaan saada. Testaustilanteessa testikäyttäjille annetaan tehtäviä, jotka he suorittavat ilman apua testaaajan seuratussa tilannetta vierestä muistiinpanoja tehden. Testikäyttäjä kertoo ääneen miksi hän tekee tiettyjä päätöksiä ja mitä hän ajattelee käyttäessään sovellusta. Tutkijat voivat tehdä johtopäätöksiä perustuen käyttäjien kommentteihin ja siihen kuinka kauan tehtävien suorittamiseen meni aikaa. Kannattaa myös tutkia kuinka oppiminen vaikuttaa käytettävyyteen antamalla joidenkin käyttäjien tutustua ohjelmaan jonkin aikaa ja sen jälkeen suorittaa tehtäviä, sillä joissakin tapauksissa käytön tehokkuus saattaa olla

tärkeämpää kuin nopea oppiminen.

Käyttäjättestaus ei kuitenkaan ole ongelmaton tutkimusala. Tulokset eivät välttämättä ole luotettavia, sillä ihmiset ovat erilaisia. Mikäli joltakin menee tehtävän suorittamiseen pitkä aika, ei voida suoraan päätellä että käyttöliittymässä on ongelma, sillä kyseisen henkilön luonteenpiirteet saattavat yhtä lailla vaikuttaa testitulokseen. Testin toistaminen useilla käyttäjillä tietysti parantaa luotettavuutta, mutta useisiin testitapauksiin ei välttämättä ole aikaa ja rahaa. Tilastojen avulla voidaan arvioida tulosten luotettavuutta. Myös tulosten oikeellisuuteen vaikuttaa erilaisia tekijöitä, kuten kuinka hyvin testikäyttäjät vastaavat todellisia käyttäjiä. (Nielsen 1993, sivut 170-200)

4.5 Esimerkki: Heuristinen analyysi

Esimerkissä esitellään heuristista evaluointia todellisen sovelluksen avulla. Esimerkkinä toimii kuvien julkaisuun tarkoitettu sovellus Flickr.

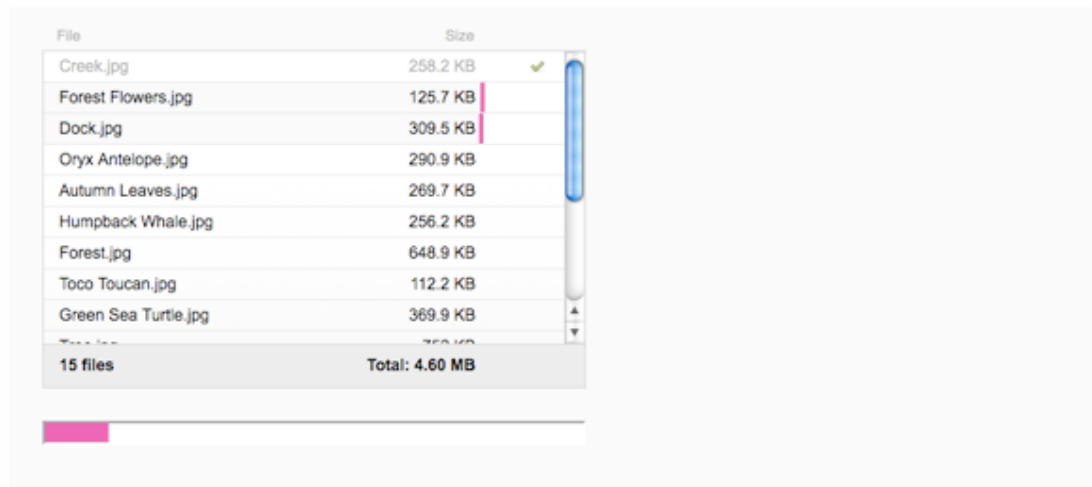
4.5.1 Flickr

Tilan näkyminen käyttäjälle: Flickrin kirjautuessa käyttäjä saa ensimmäisenä eteensä tehtävälistan, jonka suorittamalla pääsee alkuun. Kolmen kohdan tutoriaalissa kerrotaan paljonko askeleita on ja mitkä niistä on suoritettu. Myös monet toiminnot, kuten kuvien lisääminen, suoritetaan askel kerrallaan.

Flickr on toteuttanut latausaikojen indikoinnin todella onnistuneesti. Useita kuvia ladatessa jokaisen kuvan latauksen eteneminen näytetään omana animoituna palkkina, lisäksi näytetään jäljellä oleva aika kokonaisuudessaan.

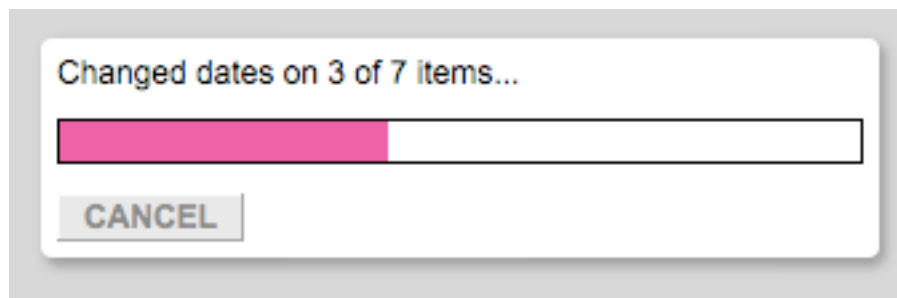
Upload photos to Flickr

You've used 1% of your 100 MB limit this month - that leaves 99.27 MB. [Upgrade?](#)



Kuva 16 Flickr näyttää selkeästi miten kuvien latausprosessi etenee

Myös kuvia muokatessa sovellus kertoo järkkymättä mitä se on tekemässä ja kauanko aikaa siihen kuluu. Lisäksi se tarjoaa mahdollisuuden pysäyttää toiminto sen ollessa käynnissä, tosin hieman harhaanjohtavalla nimityksellä, sillä Cancel-painike ei peruuta jo tehtyjä muutoksia. Tämä on kuitenkin kätevä silloin, jos käyttäjä huomaa toiminnon suorittamiseen menevän liian kauan aikaa.





Jopa pienissä asioissa on pidetty kiinni käytettävyyden periaatteista. Esimerkiksi kuvan otsikkotekstiä muokatessa elementtiin ilmestyy teksti "Saving..." , kunnes teksti saadaan tallennettua.

Tree



saving...

  Only you can see this photo ([edit](#))


Uploaded on Jan 15, 2009 | [Delete](#)

[0 comments](#)

Desert Landscape



[click here to add a description](#)

  Only you can see this photo ([edit](#))

Uploaded on Jan 15, 2009 | [Map](#) | [Delete](#)

[0 comments](#)

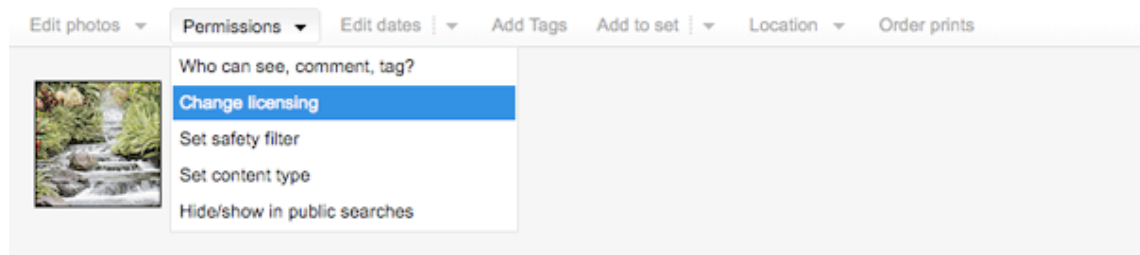
Kuva 17 Kuvatekstin tallennuksessa on myös tilasta kertova teksti

Vastaavuus tosimaailman kanssa: Flickrissä käytetty kieli on yleisesti ottaen käyttäjäystävällistä. Esimerkiksi Permissions-valikossa on ensimmäisenä valintana “Who can see, comment, tag?”, joka varmasti selittää itsensä käyttäjälle. Myös Edit photos-valikossa on käytetty ilmausta “Titles, tags and descriptions” yleistävän ja ehkä vaikeaselkoisemman termin sijasta. Yleisesti ottaen valikoissa ja ohjeissa on pyritty lyhyen ja ytimekkään sijasta mahdollisimman selittävään tekstiin.

Myös kuvien editointi on tehty mielekkääksi. Kuvia voi siirtää erilaisiin ryhmiin hiirellä vetämällä. Tosimaailman vastaava tilanne olisi pino kuvia, joita voi järjestää erilaisiin pinoihin haluamallaan tavalla ja lisätä pinoihin erilaisia muistilappuja. Ainoa poikkeus metaforasta on, että oletuksena kuvat eivät häviä alkuperäisestä pinosta, mutta sellaisenkin asetuksen saa halutessaan käyttöön.

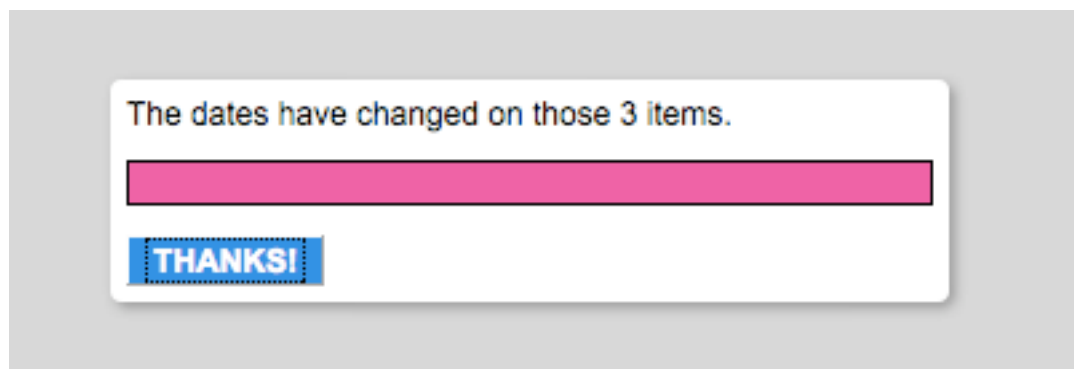
Yhdenmukaisuus ja standardit: Flickr käyttää kuvien järjestelyyn tarkoitettussa osiossaan työpöytäsovelluksista tuttuja pudotusvalikkoja, jotka ovat myös visuaaliselta ilmeeltään tunnistettavia. Konventiota on kuitenkin sovellettu hämäävällä tavalla, sillä itseasiassa navigaation elementit toimivat monella eri tavalla: toiset toimivat painikkeina, toiset avaavat alavalikon. Lisäksi mukana on sekä alavalikollisia painikkeita, että pelkkiä painikkeita. Vaikka tarkkaan katsoessa pienet visuaaliset vihjeet paljastavat toimintaperiaatteen, on lopputulos epäselvä ja virheen mahdollisuus

suuri.



Kuva 18 “Add tags” ja “Order prints” ovat painikkeita jotka avaavat dialogin, “Edit dates” ja “Add to set” toimivat itse painikkeina, mutta viereinen nuoli avaa niille alavalikon, “Edit photos”, “Permissions” ja “Location” avaavat alavalikon sekä nuolesta että otsikosta.

Virhetilanteiden välttäminen: Flickr varmistaa lähes kaikissa tilanteissa, että toiminto on varmasti tarkoituksellinen. Virheiden tekeminen on tehty vaikeaksi, mutta jatkuva vahvistaminen saattaa toisaalta ärsyttää ja hidastaa sovelluksen käyttöä. Esimerkiksi kuvien järjestäminen eri kriteereillä on toiminto, jonka saa helposti palautettua mihin tahansa asetukseen. Tällöin varmistuksen kysyminen tuntuu jo turhalta. Myös onnistuneet toiminnot täytyy aina kuitata erillisessä dialogissa, vaikka jo niiden toteuttaminen ilmaistaan selvästi, eikä peruutusmahdollisuutta ole.



Kuva 19 Flickr pyytää kiittausta suoritetuista toiminnoista, vaikka ne kestäisivät vain muutamia sekunteja, mikä hidastaa tarpeettomasti sovelluksen käyttöä. Laatikko voisi esimerkiksi kadota automaattisesti lyhyen ajan jälkeen. Ilmoituksen käyttö on kuitenkin perusteltua mikäli toimintojen suorittaminen kestää niin kauan, että käyttäjä siirtää huomionsa muualle siksi aikaa kun toimintoa suoritetaan.

Tunnistaminen, ei muistaminen: Flickrissä kaikki tarvittavat toiminnot ovat helposti

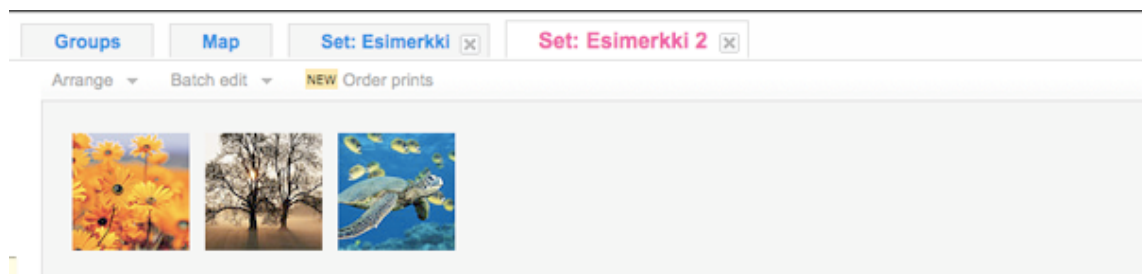
löydettävissä, joten niitä ei tarvitse erikseen etsiä. Monimutkaisemmat toiminnot suoritetaan aina tutoriaalityyppisesti askel kerrallaan, mikä auttaa varsinkin satunnaista käyttäjää.

Joustavuus ja käytön tehokkuus: Flickr ei tarjoa pikanäppäimiä, mutta oikotiet useimmin käytettäviin toimintoihin löytyvät alareunasta. Linkkien nimet eivät välttämättä aloittelijalle aukea, mutta oikotiet ovatkin toisaalta kokeneempia käyttäjiä varten.



Kuva 20 Flickrissä on sivujen alareunassa oikotielinkkejä usein käytettyihin toimintoihin.

Flickr myös lisää esimerkiksi muokattavana olevat kuvaryhmät yläreunan tabeihin, josta niihin pääsee helposti takaisin. Näin useiden eri ryhmien muokkaaminen samanaikaisesti nopeutuu huomattavasti.



Kuva 21 Muokattavana olevat kuvaryhmät ilmestyvät yläreunaan tabeiksi, jolloin niihin voi palata vaivattomasti ja nopeasti.

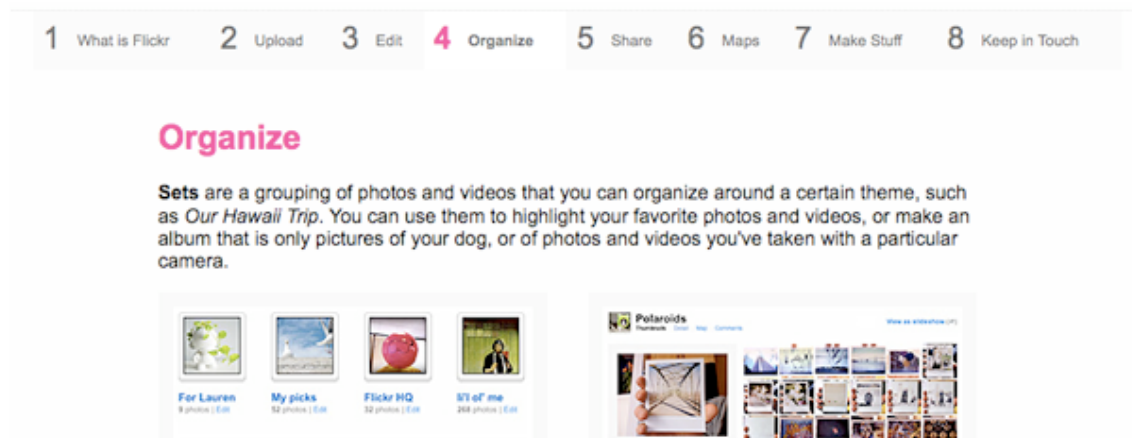
Lisäksi monet rutiinit, kuten kuvatekstin muokkaus tai oikeuksien asettaminen, on mahdollista tehdä suoraan valokuvia selatessa. Usein toistuvat varmistusdialogit kuitenkin hidastavat joidenkin rutiinien suorittamista.

Esteettinen ja minimalistinen design: Flickrissä sivuilla on paljon erilaisia ja erivärisiä elementtejä. Elementtien selkeämpi ryhmittely saattaisi helpottaa sivun hahmottamista, sillä eri osiot sekoittuvat visuaalisesti toisiinsa. Animaatiot ovat kuitenkin hillittyjä ja

informatiivisia.

Auta käyttäjää tunnistamaan virheet ja toipumaan niistä: Sovelluksessa tapahtuu varsinaisia virheitä melko harvoin, mutta virhetilanteen sattuessa kohdalle virheilmoitus on helposti ymmärrettävä: ilmoitus kertoo käyttäjän kielellä mistä on kyse ja ehdottaa ratkaisua ongelmaan.

Opastus ja ohjeistus: Flickrin rekisteröitymisen jälkeen aloittelijoille näytetään ensimmäiseksi tutoriaali, joka antaa perustiedot sivuston ominaisuuksista. Lisäksi sivuilla on toimintaohjeita, mikäli ei muuten ole selvää mitä käyttäjän pitää tehdä edetäkseen. Lisäksi kaikilta sivuilta löytyy linkki ohjeeseen ja mahdollisuuksien mukaan suoraan ohjesivulle, joka käsittelee kyseisen sivun toimintoja. Mikäli käyttäjällä on ongelmia tutoriaalin ja paikallisten ohjeiden jälkeenkin, löytyy ohjeosiosta myös tutustumiskierros sivuston ominaisuuksiin.



Kuva 22 Flickr opastaa käyttäjää myös kuvitetun esittelykierroksen avulla. Yläreunassa näytetään mitä ominaisuuksia esitellään, missä kohtaa käyttäjä on ja kuinka paljon sisältöä on kokonaisuudessaan.

4.6 Yhteenveto

Flickr pärjasi heuristisessa analyysissä melko hyvin, mikä ei sinänsä ollut yllättävää. Suurempia vikoja ei oikeastaan ollut, mutta pientä parannettavaa ja pohtimisen aihetta ehkä olisi, esimerkiksi varmistustekstien käytössä ja joidenkin sivujen visuaalisessa

hälyssä. Flickrissä on paljon sellaista, josta voisi ottaa mallia web-sovelluksen suunnittelussa, ja se onkin ollut jo pitkän aikaa yksi edelläkävijöistä alallaan.

5 Yhteenveto

RIA-tekniikoilla voidaan toteuttaa työpöytäohjelmistojen kaltaisia vuorovaikutteisuudeltaan rikkaita selainkäyttöisiä sovelluksia. Ajax-tekniikan ja asiakaspuolen laskennan avulla ne saadaan myös toimimaan nopeammin kuin perinteiset web-sivut.

RIA-sovellukset ovat hitaasti yleistymässä ja tekniikat kehittyvät nopeaan tahtiin. Täydellistä ratkaisua web-sovellusten toteuttamiseen ei toistaiseksi kuitenkaan ole, sillä kaikilla olemassa olevilla tekniikoilla on omat huonot puolensa, liittyen esimerkiksi selaintukeen, saavutettavuuteen tai visuaalisen ilmeen hallintaan. Standardien kehittäminen ja niiden käyttöönotto kuitenkin parantaa tilannetta ja suunta on parempaan päin.

RIA-sovellusten käytettävyys ja käyttöliittymäsuunnittelu ovat yhtäläillä kehitysvaiheessa. Monia hyviä käytäntöjä voidaan poimia sekä työpöytäohjelmistoista että web-sivuista, mutta lopullinen ratkaisu hakee vielä muotoaan. Uusien tekniikoiden yleistyessä hienosta tekniikasta tulee usein itseisarvo, jolloin käytettävyys usein kärsii. Käyttäjät eivät ole myöskään tottuneet esimerkiksi sivun osittaiseen päivittymiseen, joten suunnittelijan on mietittävä miten käyttäjälle tuodaan ilmi että jotain on tapahtunut.

Suunnittelijat ovat kuitenkin ottaneet haasteen vastaan ja erinomaisia artikkeleita ja tutkimuksia julkaistaan tiheään tahtiin. Internetistä löytyy jo nyt lukuisia hyvin suunniteltuja ja toteutettuja web-sovelluksia, joista suunnittelijat voivat ottaa mallia ja poimia ideoita. Vanhat analysointimenetelmät ovat myös edelleen toimiva tapa arvioida sovelluksen käytettävyyttä, joskin niitä täytyy soveltaa muuttuvaan alustaan.

Lähteet

- Cato, John.** User-Centered Web Design. Pearson Education Limited (2001)
- Cooper, Reimann & Cronin.** About Face 3: Essentials of Interaction Design. Wiley Publishing, Inc. (2007)
- Jadav, Ameeta D.** Designing usable web interfaces. Pearson Education, Inc (2003)
- Nielsen, Jakob.** Usability Engineering. Academic Press (1993)
- Silver, Marc.** Exploring Interface Design. Delmar Learning (2005)
- Vesa, Jyrki.** TiVi Tietoviikko –lehti. Numero 33. (ilmestynyt 3.10.2008)

Internet-lähteet

- Cecil, Richard F.** Documenting the Design of Rich Internet Applications: A Visual Language for State (2007)
[<http://www.uxmatters.com/MT/archives/000251.php>] (luettu 15.10.2008)
- Follett, Jonathan.** Interfaces That Flow: Transitions as Design Elements (2007)
[<http://www.uxmatters.com/MT/archives/000187.php>] (luettu 15.10.2008)
- Heller, David.** RIAs: The Technology Is Exciting, but They Really Do Help Users (2005)
[<http://www.uxmatters.com/MT/archives/000041.php>] (luettu 13.1.2009)
- Maurer, Donna.** Usability for Rich Internet Applications (2006)
[http://www.digital-web.com/articles/usability_for_rich_internet_applications/] (luettu 9.10.2008)
- McMullin, Jess.** Usability Heuristics for Rich Internet Applications (2003)
[http://www.boxesandarrows.com/view/usability_heuristics_for_rich_internet_applications] (luettu 9.10.2008)
- Moritz, Florian.** Rich Internet Applications (RIA): A Convergence of User Interface Paradigms of Web and Desktop - Exemplified by JavaFX (2008)
[<http://www.flomedia.de/diploma/documents/DiplomaThesisFlorianMoritz.pdf>] (luettu 9.10.2008)
- Mullet, Kevin.** The Essence of RIA (2003)
[http://download.macromedia.com/pub/solutions/downloads/business/essence_of_ria.pdf] (luettu 25.10.2008)

Perfetti, Christine. Deconstructing Web Applications

[http://www.uie.com/events/web_app_summit/2008/articles/deconstructing_web_applications/] (luettu 9.10.2008)

Porter, Joshua. Designing Powerful Web Applications: An Interview with David (Heller) Malouf

[http://www.uie.com/events/web_app_summit/2008/articles/malouf_interview/] (luettu 9.10.2008)

Spool, Jared M. Five Usability Challenges of Web-Based Applications (2007)

[http://www.uie.com/articles/usability_challenges_of_web_apps/] (luettu 9.10.2008)

Spool, Jared M. 7 Critical Considerations for Designing Effective Applications

[http://www.uie.com/events/web_app_summit/2008/articles/designing_effective_apps/] (luettu 9.10.2008)

Useita kirjoittajia. Wikipedia: AJAX

[<http://en.wikipedia.org/wiki/AJAX>] (luettu 24.10.2008)

Useita kirjoittajia. Wikipedia: Heuristic Evaluation

[http://en.wikipedia.org/wiki/Heuristic_evaluation] (luettu 21.3.2009)

Useita kirjoittajia. Wikipedia: Rich Internet applications

[http://en.wikipedia.org/wiki/Rich_Internet_application] (luettu 9.10.2008)

Willemsen, Joost. Improving User Workflows with Single-Page User Interfaces (2006)

[<http://www.uxmatters.com/MT/archives/000149.php>] (luettu 15.10.2008)

Microsoft. Microsoft Silverlight: Light up the Web

[<http://www.microsoft.com/silverlight/default.aspx>] (luettu 28.2.2009)

Adobe. Flex 3

[<http://www.adobe.com/products/flex/>] (luettu 28.2.2009)

Adobe. Flash Player Statistics

[http://www.adobe.com/products/player_census/flashplayer/] (luettu 28.2.2009)

Esimerkit

Google Docs [<http://docs.google.com>]

Google Maps [<http://maps.google.com>]

Google Gmail [<http://mail.google.com>]

Weebly [<http://www.weebly.com>]

Grokker [<http://www.grokker.com>]

Wordpress [<http://www.wordpress.com>]

Flickr [<http://www.flickr.com>]

Finmatkat [<http://www.finmatkat.fi>]

Gap [<http://www.gap.com>]

Yahoo [<http://www.yahoo.com>]

Tag Graph [<http://taggraph.com>]

Lovely Charts [<http://my.lovelycharts.com>]

A Website Named Desire [<http://www.visitmix.com/labs/descrip/awebsitenameddesire>]