

FACEBOOK-SOVELLUSTEN KEHITYS: SUOMALAISEN YRITYKSEN FACEBOOK- LANSEERAUS

Jarmo Kontkanen

Opinnäytetyö
Huhtikuu 2013

Mediatekniikan koulutusohjelma
Tekniikan ja liikenteen ala





Tekijä(t) KONTKANEN, Jarmo	Julkaisun laji Opinnäytetyö	Päivämäärä 24.04.2013
	Sivumäärä 43 sivua	Julkaisun kieli Suomi
	Luottamuksellisuus () saakka	Verkkojulkaisulupa myönnetty (X)
Työn nimi FACEBOOK-SOVELLUSTEN KEHITYS: SUOMALAISEN YRITYKSEN FACEBOOK-LANSEERAUS		
Koulutusohjelma Mediatekniikan koulutusohjelma		
Työn ohjaaja(t) NIEMI, Kari		
Toimeksiantaja(t) Into-Digital Oy		
Tiivistelmä <p>Sosiaalinen media on 2000-luvulla noussut suureen suosioon ja samalla tärkeäksi markkinointikanavaksi. Internetin käytetyimmäksi sivustoksi on noussut Facebook, jonka käyttäjämäärä lähentelee jo miljardin käyttäjän rajaa. Valtava volyymi ja oikeiden kohderyhmien hyvä tavoitettavuus tekee Facebookista houkuttelevan markkinointivälineen, jolle oikein hyödynnettynä on vaikea löytää kustannustehokasta kilpailijaa.</p> <p>Opinnäytetyö käsittelee Facebook-sovelluskehityksen elinkaarta keskittyen sovelluksen tekniseen kehitykseen ja sen työkaluihin. Työssä käydään läpi Facebookin rajapintoja sekä oleellisimpia toimintoja kampanjamuotoisen sovelluksen toteuttamiseen ja julkaisemiseen.</p> <p>Tietoperustaa sovellettiin käytäntöön toteutetun kampanjan ja sovelluksen kautta. Kyseessä oli suomalaisen yrityksen Facebook-lanseeraus ja kampanjan starttivaiheessa julkaistu sovellus. Opinnäytetyön tavoitteena oli nivoa teoriaa yhteen toteutuksen kanssa sekä saada aikaan hyvä yleiskuva sovelluskehityksen perusvaatimuksista.</p> <p>Opinnäytetyön tavoitteet saavutettiin ja kampanjan katsottiin onnistuneen useilla mittareilla; yrityksen Facebook-sivun uusien tykkääjien suuruusluokka menee kymmenissä tuhansissa, ja yhden tykkääjän hankintahinnassa päästiin alle yhteen euroon. Opinnäytetyötä tullaan käyttämään Into-Digitalilla uusien työntekijöiden perehdytyksessä sovelluskehitykseen ja Facebook-rajapintaan.</p>		
Avainsanat (asiasanat) Facebook, sovelluskehitys, sosiaalinen media		
Muut tiedot		



Author(s) KONTKANEN, Jarmo	Type of publication Bachelor's Thesis	Date 29042013
	Pages 43 pages	Language Finnish
	Confidential <input type="checkbox"/> Until	Permission for web publication <input checked="" type="checkbox"/> (X)
Title FACEBOOK-APPLICATION DEVELOPMENT: THE FACEBOOK LAUNCH OF A FINNISH CORPORATION		
Degree Programme Media Engineering		
Tutor(s) NIEMI, Kari		
Assigned by Into-Digital Oy		
Abstract <p>While the social media has been growing as a 21st century phenomena and services based on social interaction have been climbing towards the top of the most used websites they have also proven to be a very formidable marketing channel. As of 2011 Facebook is the most used website having almost a billion users. Considering the huge volume and a good reach for the desired user segments Facebook is an alluring marketing environment.</p> <p>This bachelor's thesis revolves around a Facebook application and its lifespan focus being on the technical aspects such as application development and its tools. Facebook APIs and the most crucial operations for the creation and publication of a campaign-based application are also taken into account.</p> <p>The knowledge was put to use through an existing application created for a campaign, being part of a Facebook launch for a Finnish company. The main objective for the bachelor's thesis was to combine the technical know-how with a working implementation as a concrete result and create a good view of the basic requirements for Facebook application development.</p> <p>The goals were reached and the campaign was considered a success by several indicators; the magnitude of users that liked the Facebook-page of the company went up to tens of thousands while the price of one obtained like for the page remained under one euro. The bachelor's thesis will be used as an introduction to Facebook application development for the new employees at Into-Digital.</p>		
Keywords Facebook, application development, social media		
Miscellaneous		

SISÄLTÖ

KÄSITTEISTÖ.....	4
Lyhenteet.....	4
Muut käsitteet.....	7
1 OPINNÄYTETYÖN LÄHTÖKOHDAT.....	11
1.1 Toimeksiantaja ja taustatietoa.....	11
1.2 Opinnäytetyön tarpeet, tavoitteet ja rajaus.....	11
2 2000-LUVUN INTERNET JA SOSIAALINEN MEDIA.....	13
2.1 Ilmiö: Sosiaalistunut Internet.....	13
2.2 Internetin käytetyimmät sosiaaliset palvelut.....	14
2.3 Facebook – lippulaiva ja suunnannäyttävä.....	15
3 FACEBOOK-SOVELLUKSET.....	16
3.1 Lähtökohdat.....	16
3.2 Työn vaiheet.....	17
3.3 Sovelluskehitys.....	18
3.3.1 Pohjatyö.....	19
3.3.2 Toteutus.....	20
3.3.3 Testaus, korjauskierrokset ja ylläpito.....	21
3.4 Facebookin sovellushallinta ja API.....	22
4 ANALYTIIKAN KERÄÄMINEN FACEBOOK-SOVELLUKSISTA.....	28
4.1 Tarve.....	28
4.3 Analytiikan keräystavat.....	29
4.3.1 Eräitä valmiita analytiikkapalveluita.....	29
Google Analytics.....	29
Snoobi.....	30
Facebook Insights.....	31
4.3.2 Omat menetelmät.....	31
5 ESIMERKKI: SUOMALAISEN YRITYKSEN FACEBOOK-LANSEERAUS.....	32
5.1 Kampanjan ja sovelluksen määrittely.....	32
5.2 Suunnittelu.....	32

5.3 Rakenne.....	33
5.4 Toteutus.....	34
5.4.1 Pohjustus.....	34
5.4.2 Materiaalien käsittely.....	35
5.4.3 Tekniikka.....	36
5.4.4 Testaus.....	38
5.3.5 Julkaisu.....	39
6 JOHTOPÄÄTÖKSET JA POHDINTA.....	39
6.1 Toteutus.....	39
6.2 Kampanja.....	40
LÄHTEET.....	42

KUVIOT

KUVIO 1. Facebook-sovelluksen hallinnan perusnäkyä.....	20
KUVIO 2. Kuvio sovelluksen perusasetuksista.....	23
KUVIO 3. Dialogi, jolla sovelluksen voi lisätä hallinnoimalleen Facebook-sivulle.....	23
KUVIO 4. Esimerkki Javascript SDK:n käyttöönotosta.....	24
KUVIO 5. Sovellukseen tai sivustolle kirjautumisen client-side -arkkitehtuuri.....	26
KUVIO 6. Sovellukseen tai sivustolle kirjautumisen server-side -arkkitehtuuri.....	26
KUVIO 7. Esimerkkejä Facebook Login -toiminnosta.....	27
KUVIO 8. Esimerkki lisäoikeuksien pyytämisestä.....	27
KUVIO 9. Rakennekaavio sovelluksen näkymistä.....	34

KÄSITTEISTÖ

Tässä esitellään opinnäytetyössä käytettyjä olennaisia suomen- ja englanninkielisiä käsitteitä sekä lyhenteitä, joiden ymmärtäminen helpottaa opinnäytetyön lukemista.

Lyhenteet

API

Application Programming Interface eli rajapinta, jonka avulla eri sovellukset voivat ns. keskustella keskenään, suorittaa pyyntöjä ja vaihtaa informaatiota. Hyvin toteutettu rajapinta tarjoaa eri osapuolille ainoastaan tietyt komennot, joiden kautta toimia. Näin kumpikaan osapuoli ei pääse käsiksi toisen järjestelmään, mutta voi hyödyntää siellä olevia toiminnallisuuksia.

CSS

Cascading Style Sheets, erityisesti HTML-dokumenteille kehitetty tyyliohje ja säännöstö. CSS-tyylidokumenttia käytetään html-dokumentin muotoiluun. Yleinen nimitys CSS:n eri versioista.

DOM

Document Object Model, eräänlainen puumainen rakenne, joka kuvaa dokumentin elementtejä olioina. Tämä mahdollistaa dokumentin rakenteen joustavan käsittelyn tai sen monistamisen suoraan monille eri alustoille.

HTML (XHTML)

Hypertext Markup Language on kuvauskieli, jolla voidaan kuvata hyperlinkkejä sisältävää tekstiä sekä ko. dokumentin rakennetta. XHTML eli eXtensible Hypertext Markup Language on HTML:stä kehitetty merkintäkieli, joka täyttää XML:n muotovaatimukset.

HTTP

Hypertext Transfer Protocol, WWW-palvelinten ja selainten tiedonsiirtoon käytettävä protokolla.

JPG, JPEG

Häviöllinen kuvanpakkausformaatti. Hyvinä puolina häviöttömiin on oikein asetuksin suoritettuna pakkauksena pieni tiedostokoko suhteessa lähes huomaamattomaan kuvan laadun huonontumiseen.

JS (myös JavaScript)

Komentosarjakieli, joka mahdollistaa verkkosivuilla esimerkiksi animaatiot tai dynaamisesti päivittyvän rakenteen ja informaation.

JSON

JavaScript Object Notation, tiedonsiirtomuoto, jota voi helposti käyttää JavaScript-sovelluksissa, mutta myös monissa muissa ohjelmointikielissä. Käytännössä se on tietyn merkein eroteltua dataa merkkijonon muodossa.

LAMP

Linux Apache MySQL PHP/Perl/Python, LAMP:lla tarkoitetaan yleistä kokoelmaa avoimen lähdekoodin ohjelmia, joilla voi muodostaa WWW-palvelimen ohjelmiston sivustojen pohjalle.

Linux

Linux (tulee sanoista Linux is not Unix) on avoimeen lähdekoodiin perustuva Unix:n kaltainen käyttöjärjestelmä. Linux on yleisesti käytössä esimerkiksi WWW-palvelinten käyttöjärjestelmänä.

MySQL

My Structured Query Language. Sana "My" tulee MySQL:n alkuperäisen kehittäjän tyttären nimestä. MySQL on yksi laajimmin käytetyistä relaatiotietokantojen hallintajärjestelmistä. Opinnäytetyössä ja sovelluskehityksessä on myös mainittu

MySQLi, eli MySQL improved, joka on PHP:n alkuperäistä MySQL-tukea kehittyneempi rajapinta.

OS

Operating System eli käyttöjärjestelmä. Tässä tapauksessa tällä tarkoitetaan käyttäjän tietokoneen käyttöjärjestelmää, josta yleisimpinä esimerkkeinä mainittakoon Windows, Mac OS X sekä eri Linux-distribuutiot.

PHP

PHP: Hypertext Preprocessor on ohjelmointikieli, jota käytetään varsinkin internetpalvelinympäristöissä.

PNG

Häviötön kuvanpakkausformaatti. Hyvä yksinkertaisen ja/tai pienikokoisen grafiikan pakkaamiseen, tiedostokoko ja laatu voivat olla näissä tapauksissa jopa parempia kuin häviöllisellä pakkauksella. Kuvan yksityiskohtien ja värien lisääntyessä tiedostokoko kasvaa kuitenkin nopeasti.

URL

Uniform Resource Locator, merkkijono, joka kuvaa tietyn sisällön sijaintia Internetissä.

XFBML

eXtensible Facebook Markup Language, merkintäkieli, jonka avulla voi integroida Facebookin toiminnallisuuksia omaan Internetsivuun tai sovellukseen.

XML

Extensible Markup Language, standardi, jolla dokumentin rakenne ja informaation merkitys on kuvattavissa.

WWW

World Wide Web on standardisoitu hypertekstijärjestelmä, joka toimii Internet-verkossa.

Muut käsitteet

Apache

Avoimeen lähdekoodiin perustuva WWW-palvelinohjelmisto.

APP ID

Application Identifier, uniikki merkkijono, joka toimii Facebook-sovelluksen tunnisteena.

APP Secret

Application Secret, uniikki merkkijono, jolla Facebook tunnistaa pyyntöjen tulevan sovelluksesta eikä kolmansilta osapuolilta.

app_data

Valinnainen muuttuja, jonka sivu voi saada osana signed_request -muuttujaa. Käytännössä app_data on Facebook-sovelluksen url:n GET-parametri, joka voi sisältää merkkijonon. Tämä on ainoa keino Facebookissa viedä iframeen tietoja sen ulkopuolelta. Käyttötapa voi olla esimerkiksi; käyttäjä saapuu sovellukseen yksilöidystä url:sta, ja sovelluksessa tarvitsee tunnistaa sen perusteella kuka hän on tai mistä hän tulee. Tieto kulkee url:sta sovellukseen app_data -muuttujan mukana.

Back end, ”bäkkäri”

Käyttäjälle ”näkyvä” sovelluksen osa. Sisältää ja käsittelee sovelluksessa liikkuvan datan, jonka kanssa käyttäjä voi olla vuorovaikutuksessa front endin (ks. Front end) kautta. Esimerkki: käyttäjä täyttää ja lähettää lomakkeen (front end), back end käsittelee ja tallentaa datan sekä lähettää viestin front endille tapahtuman statuksesta (onnistui/epäonnistui), jolloin front end näyttää käyttäjälle siitä seuraavan tapahtuman.

Blog, bloggaus

Blog on kontaminaatio sanoista "web log", ja lyhyesti se on jonkun ylläpitämä Internetissä julkaistu sarja kirjoituksia, alusta näille kirjoituksille. Bloggaaminen tarkoittaa tämän palvelun ylläpitoa ja sisällön julkaisemista.

Canvas, Canvas-sovellus

Facebook-sovelluksen oma sivu.

Client-side -tapahtuma

Käyttäjän asiakasohjelmassa – tässä yhteydessä Internet-selain – tapahtuva toimenpide. Ks. myös server-side -tapahtuma.

Cookie

Eväste (myös keksiksi suomennettu), on käyttäjän tietokoneelle palvelimelta tallennettavaa dataa. Sivusto voi esimerkiksi tallentaa käyttäjän täyttämän lomakkeen tiedot evästeeseen niin, että seuraavan kerran käyttäjän tullessa sivulle lomake on ko. tiedoilla esitäytetty.

Debug, debuggaus

Debug on johdettu englanninkielisestä sanasta "bug", joka ohjelmoinnin kontekstissa tarkoittaa virhettä. Debug / debuggaus tarkoittaa virheen paikallistamista ja sen korjaamista.

FBID

Facebook-käyttäjän uniikki tunniste, merkkijono

Front end, "frontti"

Sovelluksen käyttöliittymä, sovelluksen käyttäjälle näkyvä osa. Mahdollistaa käyttäjän vuorovaikutuksen sovelluksen kanssa ja ns. tarjoaa rajapinnan datan syöttämiseen sovelluksen back endin käsiteltäväksi.

Graph API

Facebook-alustan ydin, tarjoaa rajapintoja Facebookin toiminnallisuuksien käyttöön.

Iframe

HTML-elementti, joka mahdollistaa html-dokumentin upottamisen osaksi toista html-dokumenttia.

Like / Recommend, Tykkää / Suosittele

Facebookin toiminto, jolla käyttäjä ilmaisee pitävänsä jostain, ja näin levittää informaatiota ko. asiasta.

Server-side -tapahtuma

Palvelimella käsiteltävä tapahtuma tai pyyntö. Ks. myös client-side -tapahtuma.

Sessio

Istunto (yleisesti käytetään puhuttaessa PHP:n `$_SESSION` -erikoismuuttujasta), on evästeen (kts. Cookie) kaltainen datan tallennustoiminto.

signed_request

Muuttuja, jonka sovellus saa Facebook API:lta latauduttuaan Facebook-sivun iframessa.

Tab, page tab, tab-sovellus

Facebook-sovelluksen sivu, joka on esimerkiksi yrityksen sivun alasivu.

Unix

Unix on laitteistoriippumaton käyttöjärjestelmä, jonka kehitys on aloitettu jo 1960-luvulla.

Web 1.0 / Web 2.0 (O'Reilly, 2005)

Internetille muutoksen myötä annetut "versionumerot", jotka kuvaavat Internetin muutoksen laajuutta, eivät niinkään viittaa minkään teknologian muutokseen. Käsitteet ovat edelleen osin

kiistanalaisia, eikä kiveen hakattua määritelmää ole. Muutosta käsitellään tarkemmin luvussa 2.1.

1 OPINNÄYTETYÖN LÄHTÖKOHDAT

1.1 Toimeksiantaja ja taustatietoa

Opinnäytetyön tilaaja ja toimeksiantaja, Into-Digital Oy – jäljempänä Into-Digital – on digitoimisto, jonka toiminta keskittyy mainostoimistojen ja markkinointiviestintäalan palvelemiseen. Kymmenen vuoden kokemuksella noin tuhannen projektin vuosivauhtia etenevä Into-Digital on Suomen kokonein yritys alallaan. Into-Digital työllistää 16 henkilöä kolmessa toimistossa eri puolilla Suomea. (Aalto 2012.)

Into-Digital toteuttaa asiakkaiden tarpeiden mukaan monenlaista sisältöä Internetiin, mm. sivustoja, erilaisia mainoskampanjoita ja Facebook-sovelluksia. Opinnäytetyö keskittyy Into-Digitalin ja sitä kautta myös asiakkaiden tarpeisiin Facebook-sovellusten suhteen. Sovellusten käyttäjämäärät vaihtelevat paljon, sovelluksen sisällöstä ja tyylistä riippuen joistain sadoista lähes sataan tuhanteen. Sata tuhatta käyttäjää sovelluksella on Suomen mittakaavalla paljon ja tavoitettavuusprosentti on hyvä.

Opinnäytetyön tarkoituksena oli esitellä Facebook-sovelluksen elinkaarta ja sovellusten käyttämistä osana yrityksen markkinointia erään toteutuksen kautta sekä luoda hyvä yleiskuva sovelluskehityksen perusvaatimuksista.

1.2 Opinnäytetyön tarpeet, tavoitteet ja rajaus

Facebookiin oli liittynyt vuoden 2011 heinäkuuhun mennessä eri arvioiden mukaan n. 1,5 - 2 miljoonaa suomalaista käyttäjää (Pönkä 2011). Yksi tapa lähestyä heitä on perustaa Facebookiin yrityksen oma sivu, julkaista siellä informaatiota ja lisätä sinne sovelluksia potentiaalisten asiakkaiden käytettäviksi. Yrityksen kannalta Facebook-sivun ja -sovelluksen haasteista tärkeimmiksi voi mainita mm. kuinka ihmiset saadaan kiinnostumaan sivusta ja jakamaan

sitä, kuinka heidät saadaan käyttämään sovelluksia ja kuinka yritys voi hyötyä tästä?

Markkinoinnin ja mainonnan perimmäinen tarve on saada näkyvyyttä, josta seurauksena myynti saa lisäpotkua. Toki asiakkailla tai erilaisilla hankkeilla voi olla myös ei-taloudellisia tavoitteita, kuten mm. informaation levittäminen tai julkisuuskuvan luominen; pohjimmiltaan näihin pätevät samat lainalaisuudet, kuin taloudellisen hyödyn tavoitteluunkin. Näkyvyyden metsästyksen pohjaa myös Facebook-markkinointi, jossa näkyvyyden aseina toimivat pääasiassa sovellukset ja sivut sekä niiden jakaminen ja tykkääminen; lyhyesti sanottuna yrittäjän läsnäolo ja vuorovaikutus mahdollisten asiakkaiden kanssa. Facebookissa käyttäjiä on jo sellainen massa, että potentiaali lumipalloefektiin on olemassa. Tykkäykset ja jaot ovat siis tavoitteena, mutta ne eivät varsinaisesti itsessään tuo lisäarvoa, vaan niiden seuraukset: näkyvyys, ja kulluttajien taipumus ostaa tuotteita, joista on olemassa muistikuva.

Facebook-markkinoinnin olennaisia kysymyksiä ja tarpeita käsitelläkseen oli opinnäytetyössä sivuttava Internetin murrosta ja sosiaalisen median nk. vallankumousta, mutta opinnäytetyön tärkein painopiste oli yksittäinen Facebookissa sijaitseva yrityksen sovellus, jonka tarkoitus on tukea yrityksen toimintaa sen sijaan että olisi itsessään yrityksen tuote. Tavoitteena oli esitellä kampanjamuotoinen Facebook-sovellus, käsitellä sen elinkaarta ja ominaisuuksia sekä toteutusta.

Sovelluksen ei tarvitse olla suuri tai monimutkainen, enemmänkin kyse on luovuudesta; suuri panostus ei aina merkitse suurta tuottoa, kuinka saada mahdollisimman pienistä jutuista mahdollisimman paljon iloa. Esimerkiksi vanhan koulukunnan ”rämpytä painikkeita X ja Y mahdollisimman nopeasti” -tyylinen peli voi poikia yllättäviä tuloksia. Yleisimmin käyttäjiä houkutellessa erilaisin palkinnoin: esimerkiksi jonkun pienen kisan jälkeen osallistumislomake, jonka täyttämällä voi voittaa jotain. Pienen palkinnon houkutellessa sovellukselle tuhansia jakajia voi panostus tulla nopeasti takaisin.

2 2000-LUVUN INTERNET JA SOSIAALINEN MEDIA

2.1 Ilmiö: Sosiaalistunut Internet

Ns. vanha Internet, jota myöhemmin on kutsuttu myös nimityksellä Web 1.0 oli yksinkertainen alusta, jolle lähinnä tiedotusvälineet ja yritykset tuottivat sisältöä. Informaation kulku oli siis pääosin yksisuuntainen perinteisen median tapaan, julkaisijoilta lukijoille ja selaajille. Internetin sisältö oli enimmäkseen staattista ja hitaasti päivittyvää, eivätkä selaajat kohdanneet toisiaan. Voidaan sanoa, että Internet oli vielä kehityksensä alkutaipaleella ja muistutti luonteeltaan esimerkiksi sanomalehteä. (Kinnunen 2010.)

2000-luku on ollut Internetille murroksen aikaa. Internetin muutoksen laajuudesta kertoo se, että sille on annettu oma nimitys ja versionumero: alettiin puhua Web 2.0:sta. Nimitys ei kaikesta huolimatta viittaa minkään varsinaisen teknologian päivittymiseen, vaan Internetin ja sisällön luonteen rajuun muutokseen.

Web 2.0:n suurin muutos oli interaktiivisuus; nyt käyttäjät pystyivät itse vaikuttamaan Internetin sisältöön ja siellä tapahtuviin asioihin. Ne, jotka Web 1.0:n aikaan olivat vain selaajan roolissa, pystyvät nyt luomaan itse sisältöä Internetiin ja jakamaan sitä eri tavoin.

Muutos on tehnyt Internetistä ja sen sosiaalisista palveluista markkinoinnin kultakaivoksen; ihmisiä on mahdollista saavuttaa entistä enemmän ja kustannustehokkaammin. Sen sijaan, että ostetaan banneritilaa jonkin sivuston sivupaneelistä, voi esimerkiksi äänestytää tarjoustuotteita tai järjestää tarjouskilpailuja ja luoda näin interaktiivisen kanssakäymisen pohjaa suoraan asiakkaiden kanssa. Tutkimus osoittaa, että mitä enemmän myyjä oli itse toimijana ja aktivoimassa ihmisiä toimintaan ja kanssakäymiseen, sitä enemmän hän sai ihmisiltä vastinetta takaisin. Jari Lähdevuoren (2012, 19) mukaan passiivisuus ei kannusta käyttäjiä vuorovaikutukseen.

Opinnäytetyössä käytetään usein käsitettä sosiaalinen media. Tällä tarkoitetaan digitaalisia viestintäkanavia, joissa kuka tahansa voi tuottaa sisältöä ja joiden viestinnässä toteutuu moni-moneen -yhteys. Sosiaalisiin medioihin voidaan lukea esimerkiksi yhteisöpalvelut Facebook ja Google+, kuvapalvelu Flickr, videopalvelu Youtube ja blogipalvelu Blogspot. Palveluita voi olla siis hyvinkin erilaisia, mutta yhteinen tekijä on se, että kaikissa sisältöä voi tuottaa kuka tahansa, sisältöä voi selata kuka tahansa, ja sisältöä voi kommentoida sekä jakaa. (Hintikka n.d.)

2.2 Internetin käytetyimmät sosiaaliset palvelut

Heinäkuussa 2011 julkaistu 1000 suosituimman Internetsivuston listassa Internetin muutos näkyy selvästi. Kymmenen suosituimman palvelun joukosta täysin Web 2.0:aan perustuksensa luoneita sivustoja löytyy neljä: Facebook, Youtube, Wikipedia ja Blogspot, joista Facebook ja Youtube pitävät listan kahta kärkipaikkaa. (The 1000 most-visited sites on the web 2011.)

Jokainen edellämainituista palveluista perustuu aiemmin Internetin selaajien roolissa olleiden ihmisten toimintaan ja aktiivisuuteen. Selaajat luovat, jakavat ja kommentoivat sisältöä itse.

Facebook on yhteisöllinen Internetpalvelu, johon käyttäjä voi kirjautua omalla nimellään ja julkaista itsestään tietoa, vaihtaa kuulumisia kavereidensa kanssa sekä esimerkiksi pelata pelejä tai kommunikoida yritysten kanssa. Facebookia käsitellään tarkemmin luvussa 2.3.

Youtube on yhteisöllinen videopalvelu, johon käyttäjät voivat julkaista videoita sekä kommentoida, tykätä ja jakaa muiden käyttäjien tuotoksia. Sivuston suosio on kasvanut huimaksi, ja myös yritykset löysivät Youtuben. Yrityksillä on omat sivut ja tunnukset, joiden kautta voivat julkaista esimerkiksi tuote-esittelyjä tai elokuvien trailereita.

Wikipedia on voittoa tavoittelematon nk. sanakirjapalvelu, jossa käyttäjät luovat sanakirjan sisällön itse. Nykyään Wikipedia on yksi suosituimmista lähteistä tiedonhauille laajuutensa ja laatunsa ansiosta. Vaikka monet tahot edelleen

epäilevät Wikipedian tietojen luotettavuutta palvelun luonteen takia, on se todettu jo vuonna 2005 yhtä luotettavaksi kuin arvostettu Encyclopaedia Britannica (Terdiman 2005).

Blogspot on bloggauspalvelu, jossa käyttäjät voivat julkaista kirjoituksiaan ja julkaisijoita eli nk. bloggaajia on laidasta laitaan: päiväkirjaa pitävistä aktiivisiin poliittisiin kannanottoihin, ruokaohjeista teknologiauutisiin. Bloggaus on tullut jäädäkseen, ja Internet on osoittanut, että blogata voi kaikesta.

2.3 Facebook – lippulaiva ja suunnannäyttäjä

Opinnäytetyössä käsiteltävä sivusto, sosiaalinen palvelu Facebook, oli oikeassa paikassa oikeaan aikaan; alunperin yliopiston sisäiseen käyttöön vuonna 2004 perustettu sivusto haali vuoden aikana lähes miljoona käyttäjää. Palveluun otettiin mukaan nopeasti lisää yliopistoja, ja lopulta Facebook avattiin suurelle yleisölle kokonaan. Tässä vaiheessa, noin kahdessa vuodessa, käyttäjäkunta oli jo yli kymmenkertaistunut 12,5 miljoonaan käyttäjään. Siitä kasvu on jatkunut, ja Facebookin mukaan palvelulla oli maaliskuussa 2012 jo 901 miljoonaa aktiivista käyttäjää (Key Facts 2012).

Suomalaisia Facebook kertoo palvelustaan löytyvän noin 2 miljoonaa, ja kuten jo aiemmin mainittua, eri lähteiden arviot aktiivisten suomalaiskäyttäjien todellisesta määrästä vaihtelevat 1,5 ja 1,7 miljoonan välillä. Kun kokonaisuudessa kansakunnasta on tavoitettavissa noin kolmannes, ei se ole markkinoinnissa enää yhdentekevänä ohitettava seikka. Facebook-markkinoinnista onkin tullut yhä suosituempaa, ja esimerkiksi Pyhäjärven (2011) mukaan taktinen Facebook-markkinointi onkin start-up -yrittäjälle erittäin kustannustehokas vaihtoehto, ellei jopa paras markkinointikeinoja ja näkyvyyttä hakiessa.

Sosiaalisen median tarjoamasta suuresta volyymistä huolimatta tulee huomioida, että näkyvä ja kallis markkinointi sosiaalisessa mediassa ei takaa menestystä. Kaushikin (2012) sanoin, ”In the end, business success (amazing products and services) can be hugely amplified by social media. Social media success can rarely make up for core problems with your business / business strategy / products and services.” Vapaasti suomentaen Kaushik toteaa, että

sosiaalisella medially voi olla huikea vahvistava vaikutus toimivalle yritykselle ja hyvillä tuotteilla sekä palveluille, mutta sosiaalinen media harvemmin voi paikata perustavanlaatuisia ongelmia yrityksen toiminnassa.

3 FACEBOOK-SOVELLUKSET

3.1 Lähtökohdat

Sovellukselle asetetut tarpeet ja tavoitteet ovat yleensä aina samat, ja ne voi jakaa karkeasti kahteen osaan: Näkyvyyden ja käyttäjien määrän lisääminen, sekä jo olemassa olevan käyttäjäkunnan tietojen kerääminen asiakasrekisteriin. Ensimmäisessä vaiheessa uudet tulokkaat ja yrityksen sivun sekä sovelluksen käyttäjät pitää saada kiinnostumaan, aktivoitumaan, jakamaan tietoa ja palaamaan sivulle tai sovellukseen uudestaan. Onnistunut kampanja voi saada aikaan lumipalloefektin: tuloksena on joukko tyytyväisiä käyttäjiä, joiden tykkäykset ja jaot tuovat lisää näkyvyyttä, ja sitä kautta sivu alkaa ruokkia sekä itseään että yritystä. Toisessa vaiheessa pyritään hyödyntämään olemassaolevaa käyttäjäkuntaa ja saada heidät lisäämään tietonsa asiakasrekistereihin.

Esimerkki käyttäjäkunnan mahdollisesta laajenemisesta: keskimäärin suomalaisella Facebook-käyttäjällä on 138 kaveria (Suomalaisella on verkossa - - 2010). Yksi käyttäjä tykkää tai jakaa sovelluksen, jolloin se voi päätyä 138 käyttäjän näkyville. Jos jokainen heistä kävisi katsomassa, mitä kaveri on tehnyt, on jako saavuttanut jo 19044 käyttäjää. Mikäli jokainen heistä kävisi tekemässä saman, on jaon saavuttama käyttäjämäärä yli 2,6 miljoonaa, mikä jo ylittää suomalaisten määrän Facebookissa.

Tapahtumaketju ja tilanne on äärimmäisen teoreettinen eikä todennäköinen, mutta kuvastaa silti potentiaalia voimakkaasti kasvaviin ilmiöihin.

Vertailun vuoksi, Suomen tavoittavimman kanavan, MTV3:n, parhaaseen katselu-aikaan mainos voi tavoittaa yli 10-vuotiaista tv:n katsojista keskimäärin n. 1,3 miljoonaa, eli noin puolet edellämainitun Facebook-skenaarion tavoitettavuudesta. (TV-mittaritutkimuksen tuloksia 2012.)

Käyttäjien ”koukuttaminen” ja aktivoiminen on siis tärkeä osa toimivaa Facebook-markkinointia. Osa sovelluksen toiminnallisuuksista on mahdollisia ”pullonkauloja”, jotka karsivat käyttäjiä ja näin myös mahdollista näkyvyyttä. Esimerkiksi sivusta tykkääminen ja luvan antaminen sovellukselle voivat olla käyttäjille kynnyksysymyksiä. Jotkut sovelluksen toiminnallisuudet kuitenkin saattavat vaatia jonkun edellämainituista toimiakseen, joten sovelluskehitys on usein kompromissien tekemistä.

3.2 Työn vaiheet

Työ alkaa asiakasyrityksen tarpeesta. Tarve voi esimerkiksi joko uuden tuotteen lanseerauksesta ilmoittaminen tai ylipäättään näkyvyyden lisääminen. Kun tarve, ns. ongelma, on saatu määriteltyä voi alkaa työ ongelman ratkaisemiseksi. Ensimmäinen vaihe on pääosin ideointia ja kommunikointia, josta onnistuessa syntyy mahdolliselle kampanjalle määritelmä ja tavoitteet. Lopputulos on esiselvitys ongelman ratkaisun toteutukselle. Koska opinnäytetyössä painopiste oli Facebook-sovelluksilla – jäljempänä sovellus – työn vaiheet jatkuvat sillä oletuksella, että osana ongelman ratkaisua on sovellus.

Osapuolten päästyä yhteisymmärrykseen esiselvityksestä ja kampanjan perustasta alkaa selvitystyö siitä, kuinka ja millainen sovellus parhaiten tukisi kampanjaa ja sen tavoitteita. Tuloksena syntyy raaka ehdotus sovelluksen toiminnasta ja kuvaus sen mahdollisesta sisällöstä. Tämän jälkeen voi alkaa sovelluksen tarkempi määrittely ja suunnittelu niin sisällön kuin tekniikankin osalta. Lopulta tuloksena on yksityiskohtainen suunnitelma sovelluksen toiminnasta sekä kustannusarvio kyseisestä toteutuksesta.

Edellä mainittujen vaiheiden onnistuessa työ etenee lopulta tuotantoon ja sovelluksen varsinaiseen toteutukseen. Työn vaiheet voi jakaa karkeasti seuraavasti:

1. Työmääräarvio toteuttavilta kehittäjiltä
2. Projektin aikataulut ja tarvittaessa päiväkohtaiset tulostavoitteet
3. Toteutustyö
4. Testaus
5. Ylläpito
6. Lopetus

Lopetukseen kuuluu kehityksen osalta sovelluksen sulkeminen ja tulosten kerääminen. Tulokset voivat olla esimerkiksi analytiikkaa sovelluksen käytöstä, rekisteri osallistujien tiedoista tai kyselyn lopputulokset. Sovelluksen ”saattotyö” jatkuu projektinjohdon ja asiakkaan kesken tulosten luovuttamisella, analysoinnilla ja projektin virallisella päättämisellä.

3.3 Sovelluskehitys

Luvussa käsitellään sovelluksen kehitystä yleisellä tasolla teknisen toteutuksen osalta. Pohjalla on oletus siitä, että kehittäjä on saanut toteutusta varten riittävät pohjamateriaalit, esimerkiksi sovelluksen rakennekaavion, alustavat testisisällöt sekä graafisen ulkoasun ja ohjeistuksen. Toteutustyön vaiheiden määrä ja järjestys voivat vaihdella jonkin verran projektikohtaisesti riippuen esimerkiksi aikataulullisista seikoista tai projektiin osallistuvien henkilöiden määrästä. Pääosin työn vaiheiden voidaan kuitenkin katsoa jakautuvan seuraavasti:

1. Pohjatyö
2. Toteutus
 1. Taittotyö
 2. Toiminnallisuudet

3. Testaus
4. Korjaukset → tarvittaessa testaus uudestaan
5. Ylläpito
6. Lopetus

3.3.1 Pohjatyö

Ensimmäisessä vaiheessa tulee luoda sovellukselle tiedostorakenne ja ladata tarpeelliset kirjastot esimerkiksi tietokantayhteyksien ja Facebook API:n käyttöön. Tässä vaiheessa palvelimella sijaitseva projekti on vielä käytännössä html-sivusto, joka ei liity Facebookiin mitenkään. Facebookiin luodaan erikseen sovellus Facebookin kehittäjien hallinnassa. (ks. kuvio 1.) Hallinnassa olennaisimpina tietoina sovellukselle asetetaan sen tyyppi (canvas tai välilehti), sijainti ja uudelleenohjaukset Facebookin sisällä sekä sijainti Facebookin ulkopuolella (ks. kuvio 2.) Palvelimelle luotu sovelluspohja yhdistetään Facebookin sovellukseen asettamalla palvelimelle ladattuun kirjastoon sovelluksen APP ID ja APP Secret. Facebook API on tarkemmin käsittelyssä luvussa 3.4. Sovelluksen kehityksen ajan se kannattaa pitää Facebookin hallinnassa nk. hiekkalaatikko-tilassa, jolloin se on ainoastaan sille merkittyjen henkilöiden ja isäntäsivunsa ylläpitäjien käytettävissä.

The screenshot shows the Facebook App Settings interface. On the left is a sidebar with navigation options: Settings (Basic, Developer Roles, Permissions, Payments, Realtime Updates, Advanced), App Details (Review Status, Custom Open Graph, Localize, Alerts, Insights), and Related links (Use Debug Tool, Use Graph API Explorer, See App Timeline View, Delete App). The main content area is titled 'Apps > Open Graph Getting Started > Basic'. It features a header with the app name and ID/Secret, followed by a 'Basic Info' section with fields for Display Name, Namespace, Contact Email, App Domains, Hosting URL, and Sandbox Mode. Below this is a section for selecting integration options with Facebook, including Website with Facebook Login, App on Facebook, Mobile Web, Native iOS App, Native Android App, and Page Tab. A 'Save Changes' button is at the bottom.

KUVIO 1. Facebook-sovelluksen hallinnan perusnäkökulma

3.3.2 Toteutus

Toteutus voidaan jakaa karkeasti kahteen osaan, front endiin ja back endiin. Tässä yhteydessä voidaan käyttää myös termejä taittotyö ja toiminnallisuudet. Useinkaan näiden toteutus ei ole toisistaan riippuvaista siinä määrin, etteikö niitä voisi toteuttaa aluksi erikseen. Taittotyö sisältää jo olemassa olevien materiaalien perusteella sovelluksen ulkoasun ja näkymien muuntamisen määriteltujen päätelaitteiden selaimissa toimivaan muotoon. Mobiililaitteiden ja tablettien käytön yleistymisen on tuonut mukanaan omat haasteensa toteutukseen; erilaisten tekniikoiden ja erikokoisten näyttöjen määrä on lisääntynyt samalla kun useat mobiiliselainversiot toimivat hieman poikkeavasti ns. normaaleihin pc-selaimiin verrattuna. Huomioon otettavien asioiden määrä ja testauksen tarve on kasvanut jopa siinä määrin, että usein

joudutaan rajaamaan taattu toimivuus tiettyihin laitteisiin esimerkiksi aikatauluihin liittyvistä syistä.

Toiminnallisuuksien toteutus on liikkuvien osa sovelluksen teknisessä toteutuksessa aikataulujen osalta, ja se voi käsittää esimerkiksi pelin mekaniikan toteutuksen tai lomakkeen tarkastukset ja tietokantayhteydet. Toiminnallisuuksien suunnittelu alkaa sovellusrakenteen ja tietoyhteyksien hahmottamisesta sekä käyttötapausten kartoittamisesta. Toteutus pohjaa näihin suunnitelmiin, jotka tarpeen tullen päivitetään vastaamaan muutoksia mahdollisten tulevien ongelmakohtien arvioinnin helpottamiseksi. Sovelluksen sisällöstä riippuen tulee arvioida erikseen, onko esimerkiksi luokkarakenteille koodissa tarvetta, vai onko sovellus niin pienitöinen ja kertaluontoinen toteutus ettei se ole tarpeellista.

3.3.3 Testaus, korjauskierrokset ja ylläpito

Testaus jakautuu usein ensin sisäiseen testaus- ja korjauskierrokseen, kuten ylläolevassa listassa esitetty kohdat 3-4, jonka jälkeen sovellus menee ulkopuoliseen testaukseen asiakkaalle tai asiakkaille. Kummallakin kierroksella toimenpiteet ovat lopulta samat, mikäli testauksessa ilmenee korjattavaa joltain osin, puutteet korjataan ja testaus alkaa uudestaan. Puutteen ja korjauksen laadusta riippuen testausta ei kuitenkaan tarvitse aina välttämättä suorittaa täydellisenä uudestaan, varsinkaan viimeisillä kierroksilla, mikäli tehty muutos ei voi edes teoriassa vaikuttaa mihinkään muuhun sovelluksen osaan.

Testauskierrosten jälkeen todetessa sovelluksen olevan sopimuksen mukaisessa julkaisukunnossa se julkaistaan käyttäjien nähtäväksi erikseen sovittuna ajankohtana. Tässä vaiheessa sovellukseen lisätään analytiikan keräystavat, esimerkiksi Google Analytics, sekä se poistetaan Facebookin nk. hiekkalaatikkotilasta. Sovellusta voidaan tässä vaiheessa vielä testata ja seurata, että käyttäjien toimet onnistuvat. Ylläpidon aikana sovellukseen harvemmin tehdään enää ainakaan isoja muutoksia, työ on lähinnä seurantaa ja tarvittaessa pieniä päivityksiä.

Kun kampanjamuotoinen sovellus lopulta suljetaan, esimerkiksi kilpailun päättymisen tai käyttäjämäärän putoamisen takia, se poistetaan isäntäsivultaan, voidaan laittaa takaisin Facebookin hiekkalaatikkotilaan ja varmuudeksi suojata palvelimella sijaitseva kansio esimerkiksi käyttäjätunnuksen ja salasanan kyselyn taakse.

3.4 Facebookin sovellushallinta ja API

Facebookin kehitystyökalut ja API mahdollistavat monenlaisten toimintojen käytön sovelluksen ja Facebookin välillä. Opinnäytetyössä kiinnostuksen kohteena oli pääasiassa Facebookiin sivun välilehteen iframeen upotettu sovellus ja siihen liittyvät olennaisimmat toiminnot.

Sovellusten luomiseksi Facebook-käyttäjällä tarvitsee olla hyväksytty kehittäjätili. Sovellus luodaan Facebookin kehittäjähallinnassa ja luontivaiheessa sen ainoa pakollinen tieto on sovelluksen nimi, jonka tulee olla uniikki. Sovellukselle voi myös määritellä "namespace"-ominaisuuden, jota voi käyttää kustomoitujen Graph API -kutsujen ja objektien määrittämiseen. Namespacea voi myös käyttää sovelluksen url:na, sovelluksen url voi olla ilman määriteltyä namespacea esimerkiksi <http://apps.facebook.com/1234> ja määritellyn namespacen esimerkiksi <http://apps.facebook.com/sovelluksennimi>. Esimerkkien url:ssa 1234 on Facebook-sovelluksen uniikki id, kun namespacea se voidaan muuttaa luettavaan muotoon ja korvata id selkokiekisillä sanoilla. Myös namespacen tulee olla uniikki, eli kahdella Facebook-sovelluksella ei voi olla samaa namespacea.

Sovelluksen luomisen jälkeen sille tulee asettaa muutamat perusasetukset, että sen voi lisätä sivulle ja aktivoida käytettäväksi. (Ks. kuvio 2.)

Page Tab

Page Tab Name: [?]

Page Tab URL: [?]

Secure Page Tab URL: [?]

Page Tab Edit URL: [?]

Page Admin Control: [?] Enabled Disabled

Page Tab Width: [?] Narrow (520px) Wide (810px)

KUVIO 2. Kuvio sovelluksen perusasetuksista

Asetusten asettamisen jälkeen helpoin tapa lisätä sovellus jollekin hallinnoimalleen Facebook-sivulle on käyttää selaimella Facebookin tarjoamaa palvelua, suoraa url:a, jonka parametreiksi annetaan sovelluksen APP ID ja sijainti. Tällöin avautuu sivu, Facebookin dialogi, joka näyttää valitun sovelluksen tiedot ja listauksen sivuista, joille sen voi lisätä. (Ks. kuvio 2.)

f Add Page Tab

Select the Facebook Pages to add AlanMcConnell to:

Choose Facebook Pages ▼

Add Page Tab Cancel

KUVIO 3. Dialogi, jolla sovelluksen voi lisätä hallinnoimalleen Facebook-sivulle

Kun sovellus on luotu, saatettu asetukset kuntoon ja lisätty Facebook-sivun välilehteen, se näkyy sivun sovelluslistauksessa. Tällöin käyttäjä voi käyttää sovellusta iframessa osana Facebook-sivua. Kun käyttäjä avaa sovelluksen, sovellus saa Facebook API:lta signed_request -muuttujan, jonka voi parsia esimerkiksi omalla PHP-kirjastolla.

Muuttuja `signed_request` sisältää oletuksena JSON-objektin `page`, josta käy ilmi sivun `id` ja kaksi muuta olennaista tietoa:

1. Onko käyttäjä sivun ylläpitäjä
2. Onko käyttäjä tykännyt sivusta

Lisäksi `signed_request` voi sisältää muuttujan `app_data`, mikäli sellainen on asetettu sivun alkuperäisessä url:ssa. Käytännössä muuttuja on merkkijono, mutta se voi myös sisältää esimerkiksi enkoodattuna useita avain-arvo-pareja. Muuttuja `app_data` on Facebookissa ainoa tapa viedä iframeen sovellukselle tietoja sen ulkopuolelta, ja sitä voi käyttää esimerkiksi käyttäjän tunnistamiseen hänen saapuessaan uniikista url:sta.

Sovellukseen kannattaa ja usein on myös pakko ladata käyttöön Facebookin Javascript SDK (Software Development Kit). Sen avulla sovellus voi esimerkiksi kommunikoida Facebook API:n kanssa dynaamisesti ilman tarvetta sivun lataukselle joka kyselyä varten. Javascript SDK ladataan käyttöön sivuston HTML-koodissa heti `<body>` -tagin avaamisen jälkeen. (Ks. kuvio 4.)

```
<div id="fb-root"></div>
<script>
  window.fbAsyncInit = function() {
    // init the FB JS SDK
    FB.init({
      appId      : 'YOUR_APP_ID', // App ID from the App Dashboard
      channelUrl  : '//WWW.YOUR_DOMAIN.COM/channel.html', // Channel File for x-domain communication
      status      : true, // check the login status upon init?
      cookie      : true, // set sessions cookies to allow your server to access the session?
      xfbml       : true // parse XFBML tags on this page?
    });

    // Additional initialization code such as adding Event Listeners goes here
    FB.Canvas.setAutoGrow();

  };

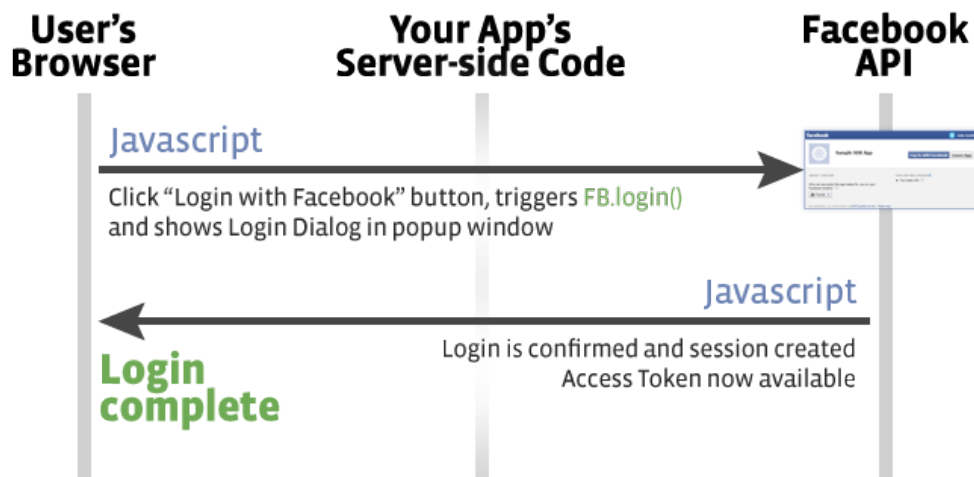
  // Load the SDK's source Asynchronously
  (function(d, debug){
    var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];
    if (d.getElementById(id)) {return;}
    js = d.createElement('script'); js.id = id; js.async = true;
    js.src = "//connect.facebook.net/en_US/all" + (debug ? "/debug" : "") + ".js";
    ref.parentNode.insertBefore(js, ref);
  })(document, /*debug*/ false);
</script>
```

KUVIO 4. Esimerkki Javascript SDK:n käyttöönotosta

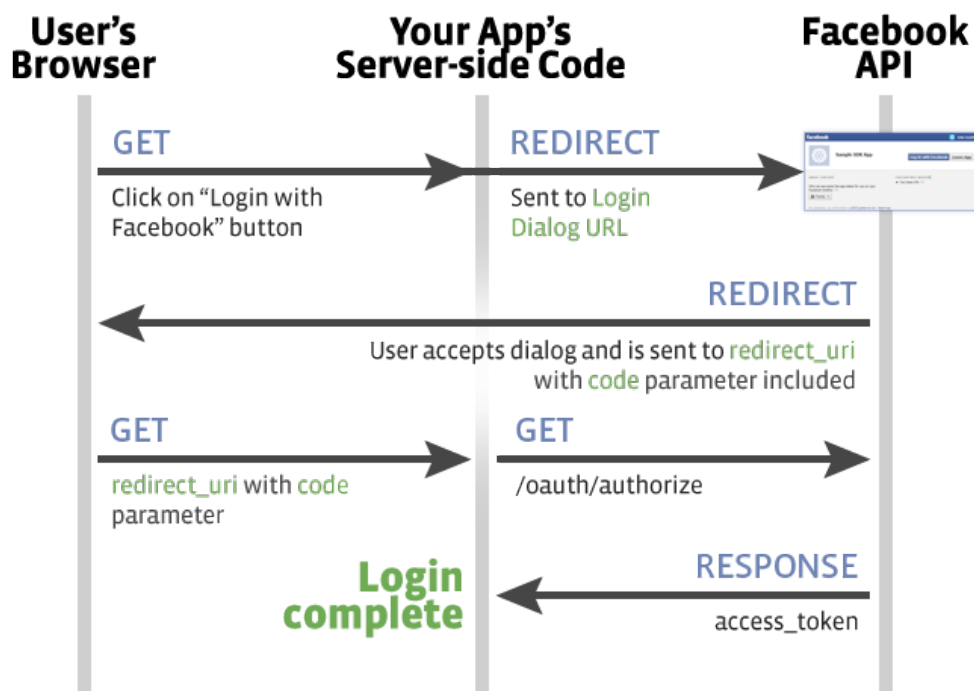
Esimerkissä Javascript SDK ladataan käyttöön asynkronisesti ja alustetaan FB -objekti sovelluksen tiedoilla. Asynkroninen käyttöönotto ei hidasta itse sovelluksen tai sen osien latausta, koska sen valmistumista ei tarvitse odotella ennen seuraavaan tehtävään siirtymistä. Tämä käyttöönottopa kuitenkin aiheuttaa sen, että sovelluksen ohjelmoinnissa on otettava huomioon tilanne, jossa käyttöönotto ei ehkä ole vielä valmis. Esimerkiksi FB -objektin käyttämiseksi tarvitsee tarkistaa, onko objektia vielä olemassa, tai sen puuttuessa javascript-virhe voi rikkoa sovelluksen toiminnan.

Yleisimpinä toimintoina voi pitää esimerkiksi kutsua *FB.Canvas.setAutoGrow()*, jolla kerrotaan Facebookille, että sovelluksessa voi olla dynaamista sisältöä ja sen koko voi muuttua. Tällöin Facebook kuuntelee sovelluksen tapahtumia ja muuttaa iframen kokoa tarvittaessa.

Javascript SDK on pakollinen esimerkiksi niissä tapauksissa, kun sovelluksessa halutaan mahdollistaa nk. lupakysely, eli käyttäjän kirjautuminen sovellukseen. Tällöin sovellus voi saada käsiteltäväksi käyttäjästä Facebookin määrittelemiä tietoja tai oikeuksia, joita se ei muuten saa. Täytyy muistaa, että sovellus sellaisenaan ei tiedä käyttäjästä mitään, ellei käyttäjä ole erikseen antanut sille lupaa päästä käsiksi kyseisiin tietoihin. Lupakysely/kirjautuminen voidaan tehdä ohjelmoinnin keinoin sekä client-side (ks. kuvio 5) – että server-side -tapahtumana (ks. kuvio 6). (Login Architecture 2013)



KUVIO 5. Sovellukseen tai sivustolle kirjautumisen client-side -arkkitehtuuri



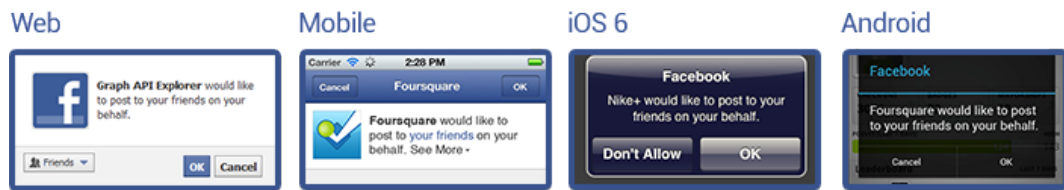
KUVIO 6. Sovellukseen tai sivustolle kirjautumisen server-side -arkkitehtuuri

Joissain tapauksissa client-side -kirjautuminen ei ole paras vaihtoehto, esimerkiksi mikäli siihen ei haluta käyttää javascriptiä, mutta se on helpoin tapa ja siksi myös Into-Digitalilla käytetyin. Tavallisella sovellukseen kirjautumisella käyttäjä antaa sovellukselle luvan päästä käsiksi käyttäjän kaverilistaukseen ja julkisiin tietoihin: fbid, koko nimi, etunimi, sukunimi, profiilin url, käyttäjätunnus, sukupuoli, käyttäjätilin kieli sekä ikäryhmä. (Ks. kuvio 7.)



KUVIO 7. Esimerkkejä Facebook Login -toiminnosta

Saadakseen muita tietoja tai oikeuksia sovelluksen tulee pyytää aina erikseen lupa ko. tietoihin ja se vaatii aina käyttäjän hyväksynnän käyttää niitä tietoja tai oikeuksia. Esimerkkinä oikeuksista, joita käyttäjä voi sovellukselle antaa, on esimerkiksi käyttäjän omalle seinälle tai kavereille kirjoittaminen (ks. kuvio 8.)



KUVIO 8. Esimerkki lisäoikeuksien pyytämisestä

Facebookilla on melkko tarkka seulonta siitä, kuinka Facebookin toimintoja ja käyttäjän tietoja tai oikeuksia sovelluksissa käytetään. Sovellus voi menettää oikeutensa ko. toimintoihin kokonaan nopeastikin, mikäli Facebook havaitsee niiden käytössä puutteita tai epäkohtia.

Nyrkkisääntönä tietojen ja oikeuksien käytölle; mitä enemmän ja laajemmin tietoja käyttäjältä halutaan, sitä epätodennäköisempää on, että käyttäjä tulee sitä hyväksymään ja käyttämään sovellusta. Kannattaa siis yrittää pärjätä mahdollisimman vähällä ja pyrkiä pyytämään ainoastaan tietoja, jotka ovat sovelluksen toiminnan kannalta ehdottoman pakollisia. Kun kyse on normaalista mainoskampanja-muotoisesta sovelluksesta ei useinkaan tarvita lupakyselyä ollenkaan ja mikäli tarvitaan, riittää pääsy käyttäjän julkisiin tietoihin. (The login dialog 2013)

4 ANALYTIIKAN KERÄÄMINEN FACEBOOK-SOVELLUKSISTA

4.1 Tarve

Datan kerääminen sovelluksesta tai Internetsivusta ylipäätäänkin voi olla hyödyllistä monista syistä, joista yleisimpinä ja kiinnostavimpina nostettakoon esille seuraavat:

- Sovelluksessa vietetty aika
- Mahdollinen paluuprosentti
- Käyttäjistä saatu tekninen tieto (OS, Internetselain, miltä sivulta käyttäjä on tullut sovellukseen, käyttäjän maantieteellinen sijainti)
- Aikajana, milloin sovellusta on käytetty eniten
- Mitkä sovelluksen sivuista ovat olleet suosituimpia
- Käyttäjän eteneminen sovelluksessa
 - Kuinka käyttäjajakauma häviää vähitellen sovelluksen loppua kohti

- Mahdolliset pullonkaulat

Kehittäjän kannalta dataa voi käyttää hyödyksi sovelluksen debuggauksessa. Esimerkiksi, mikäli huomaa jonkin selaimen osuuden putoavan reilusti jossain sovelluksen vaiheessa, lienee syytä testata se vaihe itse uudelleen; jokin voi olla pielessä.

Kuitenkin suurempi mielenkiinto asiakkaan puolelta kohdistuu käyttäjäkuntaan ja sen kiinnostuksen kohteisiin, mm. millaisia käyttäjät ovat, kuinka he käyttävät sovellusta ja mitkä osiot saavat eniten suosiota. Tältä pohjalta voi kehittää omaa toimintaa, sovellusta tai sovelluksia ja niiden toimintaa.

4.3 Analytiikan keräystavat

Tässä luvussa esitellään erilaisia tapoja kerätä dataa sivustolta tai sovellukselta. Omien metodien kehittäminen on mahdollista ja myös pakollista, mikäli haluaa sivustoltaan erikoisempaa tietoa. Valmiit palvelut tarjoavat erilaiset perustoiminnot, joilla pääsee hyvinkin pitkälle.

Normaalisti on siis erittäin suositeltavaa käyttää valmiita palveluita, koska se vähentää omaa työmäärän tarvetta huomattavasti. Riittävän laajojen seurantalajärjestelmien kehittäminen on todella vaativaa, eikä se ole enää tarkoituksenmukaista varsinkaan pienemmissä projekteissa.

4.3.1 Eräitä valmiita analytiikkapalveluita

Google Analytics

Google Analytics on käytetyin analytiikan keräyspalvelu (Usage of traffic analysis tools for websites 2012). Palvelu on perustoiminnoiltaan ilmainen ja vanhana palveluna kehittynyt hyvin monipuoliseksi.

Google Analyticsin käyttöön tarvitaan käyttäjätunnukset. Hallinnassa luodaan ensin tili, jolle lisätään omaisuus (jokin verkko-osoite). Sen jälkeen palvelu antaa seurantakoodin, joka lisätään osaksi HTML-dokumenttia, mieluiten head-

osion loppuun. Seurantakoodi on javascriptiä, ja sitä voidaan sivunlatausten yhteydessä kutsua myös manuaalasti javascript-komentona. Tästä on hyötyä erityisesti Facebook-sovelluksissa, joissa koko sovelluksen sisältö voi olla yhdessä dokumentissa: tällöin sivunlatauksiin perustuva seuranta ei toimi, koska se lähettää kutsun vain kerran, mutta kun käyttäjä navigoi sovelluksessa, javascriptillä näkymiä piilottaessa ja avatessa voi samalla lähettää Google Analyticsiin seurantakutsun.

Google Analyticsin heikko lenkki lienee sen päivitysväli: seuranta ei ole reaaliaikaista, vaan tapahtuu viiveellä. Esimerkiksi, jos sovellus vastaa kutsuihin hitaasti, Google Analyticsistä ei voi saada tietoa, onko kyseessä piikki käyttäjämäärissä, vai onko jokin muu vialla.

Snoobi

Snoobi on nostettu osaksi opinnäytetyötä, koska:

- Se on suomalainen palvelu
- Sitä pyydetään Google Analyticsin jälkeen seuraavaksi eniten
- Se on kehittynyt, ja tarjoaa erityisesti Facebookia varten kehitetyt työkalut

Snoobia ei käytetä läheskään yhtä paljoa, kuin Google Analyticsiä, eikä se juhli edellä viitatulla käytetyimpien seurantapalvelujen listallakaan, mutta sillä on silti oma osansa toimivaa seurantaa.

Snoobi on suomalainen maksullinen analytiikan keräyspalvelu. Se toimii Google Analyticsistä poiketen reaaliaikaisesti. Tällä on hyvät ja huonot puolensa; hyvänä puolena analytiikka on ajantasaista ja näin käyttäjä pysyy ajan hermolla paremmin, huonona puolena se ei sovi todella vilkkaasti liikennöidylle sivulle, jossa tapahtuu joka hetki paljon.

Snoobi on kehittänyt Facebook-sovelluksia ja -sisältöä varten omat työkalut, joten erityisesti Facebook-sovellusten seurantaan se soveltuu mainiosti.

Facebook Insights

Facebookilla on myös oma palvelu datan keräämiseen ja visuaaliseen esittämiseen. Facebook Insights on n. kerran vuorokaudessa päivittyvä analytiikkasivu, jonka Facebook-sivun admin voi nähdä.

Insights tarjoaa nähtävälle dataa Facebookista, jota ulkopuoliset palvelut eivät voi tarjota. Esimerkiksi sivun aktiivisuuden tason, sivun tykkääjät, sivun tykkääjien kaverit, kuinka tietoisuus sivusta on levinnyt, ja tilastoinnin kehitys aikajanaalla.

Facebook Insightsia on moitittu siitä, ettei se olisi kaiken tilastoinninsa kanssa aivan sataprosenttisen tarkka, mutta yhteiskäytössä muiden analytiikkapalveluiden kanssa sillä on osansa palvelun kehityksen seuraamisessa.

4.3.2 Omat menetelmät

Dataa sovelluksen käytössä ja erityisesti sovelluksessa etenemisessä voi kerätä tehokkaasti myös omin menetelmin. Mikäli sovelluksen eri näkymät on toteutettu eri tiedostoissa siirtymisenä ja sivun latauksina, voi dataa kerätä talteen esimerkiksi get- tai post-parametreinä, ja suorittaa yksinkertainen päivitys tietokannassa olevaan yksilöityyn riviin. Get-parametrinä data kulkee sovelluksen URL-osoitteessa, kun taas post-parametrinä se voidaan sijoittaa piilotettuun lomake-elementtiin. Facebookissa tällä ei yleensä ole väliä, koska sovelluksen todellinen URL ei iframesta johtuen käyttäjälle näy.

Mikäli sovellus on toteutettu yhtenä tiedostona, jossa näkymät sijaitsevat joko flash-tiedostossa tai eri html-elementeissä, voi dataa kerätä javascript-taulukkoon tai -muuttujaan, ja lähettää tietokantaan päivitettäväksi Ajax-tekniikalla.

5 ESIMERKKI: SUOMALAISEN YRITYKSEN FACEBOOK-LANSEERAUS

5.1 Kampanjan ja sovelluksen määrittely

Into-Digital toteutti syksyllä 2012 erään suomalaisen yrityksen – jäljempänä tässä luvussa käytetään termiä asiakasyritys – Facebook-lanseerauksen. Kampanjan ja toteutuksen tavoitteena oli kerätä vähintään 30000 tykkääjää asiakasyritykselle kampanjan aikana. Toinen olennainen tavoite oli päästä kustannuksissa selkeästi alle mediatoimistojen arvioiman yhden tykkääjän hankintahinnan, n. 3 euroa per hankittu tykkääjä. Asetettu tavoite oli päästä alle 1 euron tasolle per hankittu tykkääjä. (Aalto, 2012.)

5.2 Suunnittelu

Sovelluksen mekanismina toimi arvonta viihteellisesti toteutettuna hedelmäpelinä, jonka jokainen osallistuja nosti pääpalkinnon voittosummaa tietyllä asetetulla korotuksella. Pääpalkinnon arvolle määriteltiin lähtötaso ja yläraja, ja pääpalkinnon lisäksi oli mahdollisuus voittaa päivittäin muita palkintoja pika-arvontana, joista käyttäjä sai heti osallistuessaan tiedon mahdollisesta voitosta. Kampanjamekanismi ja palkinnot oli mitoitettu siten, että niiden oli tarkoitus kannustaa osallistujia kutsumaan uusia tykkääjiä reilusti asetettua tykkääjämäärän minimitalvoitetta enemmän. Nimenomaan palkintojen runsaus oli tässä toteutuksessa yksi perustavimmista ominaisuuksista.

Arvontaan osallistumisen suunniteltu prosessi:

1. Käyttäjä saapuu asiakasyrityksen Facebook-sivulle ja näkee kehoituksen tykkätä Facebook-sivusta voidakseen osallistua arvontaan.
2. Käyttäjä tykkää asiakasyrityksen Facebook-sivusta.
3. Käyttäjä osallistuu pääpalkinnon arvontaan pelaamalla hedelmäpeliä,

samalla käyttäjä saa heti tiedon mahdollisesta pika-arvonnan voitosta.

4. Käyttäjää kehoitetaan kutsumaan kavereitaan mukaan kasvattamaan palkintosummaa.
5. Käyttäjää kehoitetaan osallistumaan huomenna uudelleen pika-arvontaan hedelmäpeliä pelaamalla.
6. Käyttäjä palaa seuraavana päivänä pelaamaan ja kutsuu uusia kavereita mukaan.

Lisäksi pääpalkinnon arvontaan osallistuminen rajoitettiin yhteen kertaan per käyttäjä, vaikka peliä olisi pelannut useamman kerran. Pääpalkinto siis arvottiin tunnistettujen henkilöiden kesken, ei pelikertojen mukaan.

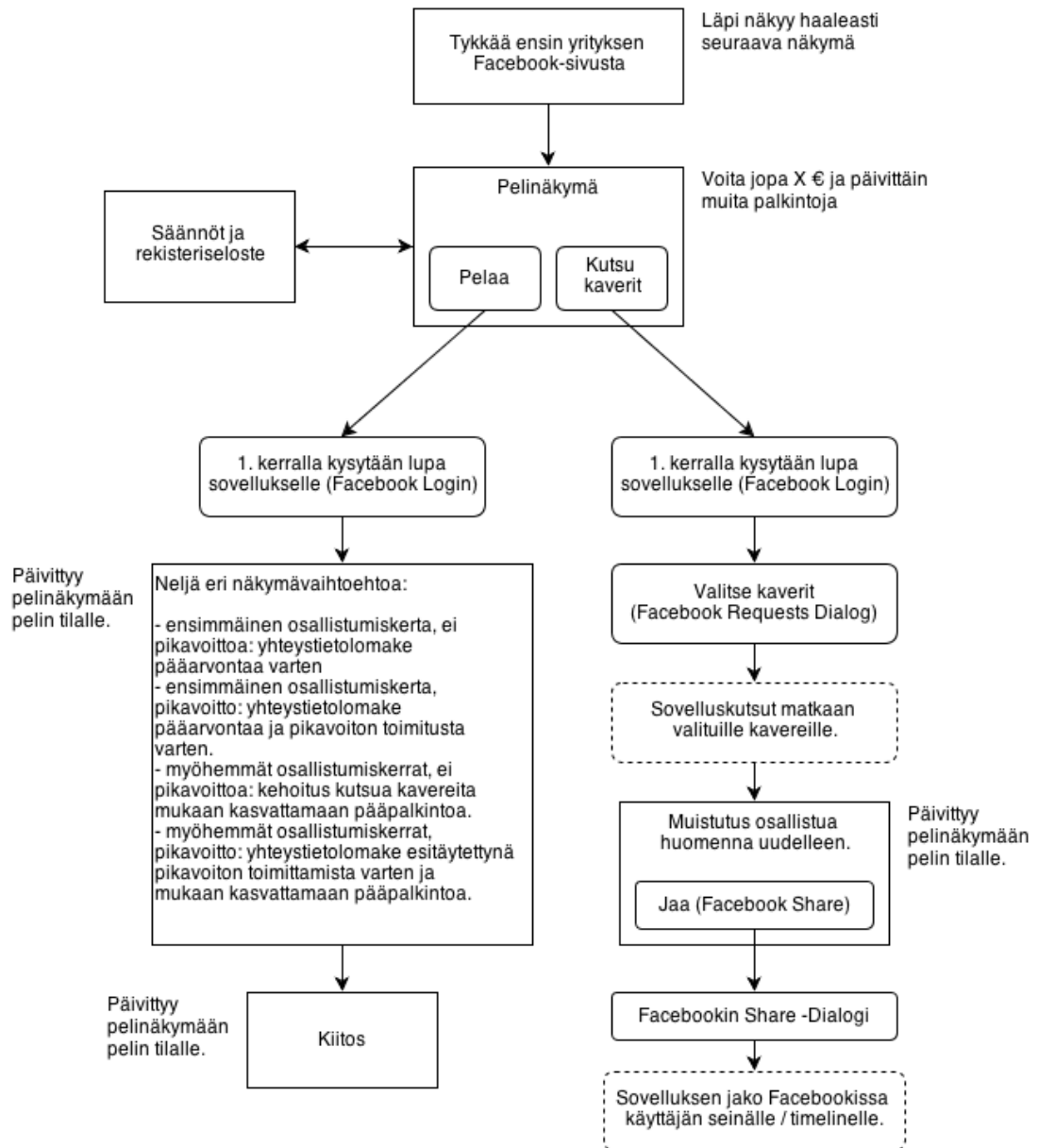
Pikapalkintoja sen sijaan pystyi voittamaan päivittäin pelaamalla hedelmäpeliä. Mahdollistamalla useampi pelikerta ja tekemällä siitä houkuttelevampaa kasvatettiin todennäköisyyttä kaverien kutsumiselle.

Kampanjan suositeltu kesto oli vähintään kaksi, mieluiten kolme viikkoa.

Riittävän pitkällä ajalla pystyttiin varmistamaan se, että viraaliefekti ennätti käynnistyä kunnolla ja tieto kampanjasta alkoi levitä myös käyttäjien kesken. Osaksi kampanjaa vaadittiin asiakasyrityksen Facebook-sivuilla tapahtuvaksi päivittäin kilpailuun liittyvää nostatusta ja kommentteihin reagointia nojaten esimerkiksi Lähdevuoren (2012, 19) julkistamiin tutkimustuloksiin.

5.3 Rakenne

Sovelluksen voi kuvata putkimaisena rakenteena, jossa tiettyihin näkymiin tai lopputuloksiin päästäkseen on tietyn tapahtumaketjun toteuduttava (ks. kuvio 9).



KUVIO 9. Rakennekaavio sovelluksen näkömistä

5.4 Toteutus

5.4.1 Pohjustus

Opinnäytetyössä ei käsitellä web-kehityksen ruohonjuuritason perusteita, koska niissä riittää asiaa useampaan opinnäytetyöhön. Oletuksena siis oli, että pohjalla on vähintään auttava kokemus web-kehitystyöstä ja HTML:stä.

Alustaksi valittiin pilvipalvelin sen hallittavuuden, vikasietoisuuden ja hyvien skaalausmahdollisuuksien perusteella. Tietokannalle varattiin niinkään oma palvelin erikseen. Toteutuksen pohjalle valittiin LAMP -ympäristö. Valinta oli lopulta helppo useista syistä; ympäristö on Into-Digitalin kaikille kehittäjille ennestään tuttu, se on monipuolinen ja mukautuva, sille löytyy runsaasti tukea ja hyvä dokumentaatio, sekä sen katsottiin olevan riittävän tehokas ja optimoitavissa tarvittavaa käyttöä varten. Sovelluksen front end perustui HTML:ään, Javascriptiin ja CSS:ään. Toteutustapa valittiin lähinnä mobiililaitteet mielessä pitäen; vaikkakin peli olisi ollut nopeampi ja kenties näyttävämpi toteuttaa käyttäen Flash-kehitysympäristöä, sillä valinnalla olisi suljettu kohderyhmästä suuri osa mobiilikäyttäjiä pois.

Toteutuksen alussa kehittäjä jakoi sovelluksen kehityksen eri vaiheisiin ja laati sen perusteella työmääräarviot eri vaiheille sekä tavoitteet vaiheiden valmistumisesta työpäivän tarkkuudella. Tämän tarkoituksena on lähinnä helpottaa projektipäällikön suunnitelmien laatimista sekä toimia omana muistilistana työn edistymisen seuraamiselle. Vaiheistus ja työmääräarviot tuli tarkastaa vähintään päivittäin ja samalla laatia projektipäällikölle selvitys edistymisestä sekä tavoitteiden toteutumisesta.

5.4.2 Materiaalien käsittely

Työn alkamiseksi ja sujuvan etenemisen varmistamiseksi ensimmäisenä tuli tarkastaa toimitetut materiaalit. Materiaalien tuli olla tiettyjen vaatimusten mukaiset. Kun materiaaleissa huomasi puutteita – esimerkiksi puuttuvia tekstejä tai näkymien taittoja – saattoi ne listata ja vaiheistuksen perusteella ilmoittaa koska niiden tulisi olla saatavilla.

Sovellusten näkymien ulkoasu toimitettiin Adobe Photoshop -ohjelmiston tukemissa tiedostomuodoissa. Käytännössä näistä ulkoasuohjeistuksista eli nk. taitoista tuli arvioida kaikki ne osat, jotka pystyi toteuttamaan suoraan CSS-tyyleillä, ja pilkkoa loput elementit omiksi kuvikseen käytettävään muotoon. Työvaihetta helpotti toimitettujen materiaalien perusteella etukäteen paperille tehdyt hahmotelmat, ns. rautalankamallit eri näkymistä ja niiden

elementeistä sekä sijoittelusta. Pienikokoisten ja graafisesti yksinkertaisten elementtien kuvanpakkausmetodina käytettiin PNG-pakkausta, kun suuremmat graafiset kokonaisuudet, esimerkiksi taustakuvat, käsiteltiin JPG-tiedostomuotoon.

5.4.3 Tekniikka

Käytännössä sovellus oli yksi HTML-dokumentti, jonka sisältöä hallittiin AJAX-tekniikoilla. Sovelluksessa tapahtuva toiminto siis käsiteltiin ensin Javascriptillä, tarvittaessa toiminto eteni AJAX:lla varsinaisen back end:n käsiteltäväksi sovelluksen PHP-kirjastoihin, jotka vielä tarpeen vaatiessa kommunikoivat MySQL-tietokannan kanssa.

Esimerkki 1, yksinkertainen näkymän vaihto, käyttäjä klikkaa jotain sovelluksen painiketta. Painike on HTML-linkkielelementti, johon on Javascriptissä kiinnitetty ”kuuntelija”. Se on Javascript-metodi, joka odottaa tapahtumaa, ja mikäli tapahtuma käy toteen, tekee määritellyn tapahtumaketjun. Jos käytössä on Javascript-kirjasto jQuery, tapahtuman koodi voisi näyttää yksinkertaisimmillaan allaolevalta esimerkiltä.

```
$('#painike').click(function(event) {
    event.preventDefault();
    $('#nakyma1').hide();
    $('#nakyma2').show();
});
```

Painikkeen painalluksen tapahtuessa estetään linkkielelementin oletustoiminnallisuus, esimerkiksi sivunlataus, ja korvataan se omalla toiminnallisuudella. Esimerkissä piilotetaan html-elementti, jonka ID eli tunniste on nakyma1, ja vastaavasti näytetään elementti jonka tunniste on nakyma2.

Esimerkki 2, tapahtuman tarvitsee suorittaa AJAX-kutsu back end:lle ja suorittaa seuraava toimenpide takaisin palautuvan datan perusteella. Tässä tapauksessa kyse on lomakkeen tietojen lähetyksestä ja tallennuksesta

```
$("#lomake").live("submit",
```

```

    tallenna);
function tallenna() {
    var virheet = new Array();
    // Tarkistetaan lomakkeen virheet,
    // mikäli löytyy, korostetaan ne ja
    // kehoitetaan korjaamaan, esim:
    if(!$('#ika').is(':checked'))
        virheet.push('ika');
    $.each(virheet, function(index, value) {
        // virheenkäsittely ennen lähetystä
    });

    // Jos virheitä ei löydy, edetään
    if(virheet.length < 1) {
        // AJAX-kutsu
        $.ajax({
            type: 'POST',
            url: 'http://esimerkki.fi',
            data: $("#lomake").serialize(),
            success: function (data, status) {
                var vastaus = $.parseJSON(data);
                // Mikäli vielä tuli virheitä, käsitellään
                if(vastaus .status == 'error') {
                    // virheenkäsittely
                }
                else {
                    // Esim. siirtymä kiitosnäkyään
                }
            }
        });
    }
}
}

```

Ylläolevassa esimerkissä tiettyyn URL:iin lähetetty lomakkeen data luetaan, käsitellään, tarkastetaan ja tallennetaan. Näihin toimenpiteisiin on käytetty yrityksen omia PHP-kirjastoja, eikä lähdekoodia siltä osin ole julkaistavissa. Yleisestiottaen tietokantojen pohjalla toimii MySQL ja sitä käytetään PHP:n

MySQLi -rajapinnan avulla. Tiedon ja muuttujien kuljettaminen Javascriptin ja PHP-kirjastojen välillä toteutettiin yhteensopivuuden takaamiseksi joko standardoituna URL-enkoodattuna merkkijonona tai JSON-merkkijonona. Näissä etuna oli se, että kummastakin ohjelmointikielestä löytyi valmiiksi rajapinnat datan käsittelyä ja muuntamista varten.

Analytiikan keräämiseen sovelluksessa käytettiin Google Analytics -palvelua. Koska sovelluksessa ei tapahdu sivunlatauksia, piti kaikki kutsut palveluun määritellä ja toteuttaa erikseen. Näkymien avaamiset päädyttiin imitoimaan palveluun sivunlatauksiksi ja loput toiminnot tallennettiin palveluun tapahtumina.

5.4.4 Testaus

Sovelluksen testaus alkoi kehittäjän osalta näkymien taiton testauksella eri päätelaitteilla ja selaimilla. Käytännössä erikseen testataan viisi käytetyintä selainta ja tarvittaessa niiden useampi versio sekä Windows- että Mac Os X-käyttöjärjestelmissä (Browser Statistics 2013). Tällä kertaa toteutukseen ei tehty erikseen mobiililaitteille optimoitua taittotyötä, vaan testauksessa huomioitiin se, että sovellus vain toimi mobiililaitteilla. Tämä tarkoittaa lähinnä sitä, että mobiililaitteella näkymä on sama kuin suuremmilla näytöillä, mutta varmistetaan, että laite skaalaa sen oikein ja toiminnot ovat edelleen käytettävissä.

Tavoitteena taiton osalta oli päästä elementtien suhteen lähes pikselin tarkkuuteen ja vastaavuuteen sekä muilta osin – esimerkiksi fontit – niin lähelle toimitettuja graafisia ohjeistuksia kuin mahdollista. Tämä on jokaisen projektin lähtökohta ellei erikseen ole muuta mainittu, ja taittotyön korjauskierros tähtää aina tähän tulokseen. Esimerkiksi fonttien ja tekstin osalta on kuitenkin huomioitavaa, että eri selaimet eri käyttöjärjestelmillä voivat piirtää fontin hiukan eri tavalla, joten sille ei voi mitään niin kauan kuin tekstin pitää dokumentissa tekstinä.

Toiminnallisuuksien testaukseen osallistui Into-Digitalilta sekä kehittäjä itse että testaajiksi erikseen määrätyt työntekijät. Työn toiminnallisuutta ei

koskaan testaa pelkästään tekijä itse. Kehittäjä itse testaa sovelluksen toiminnallisuudet läpi käyttötapausten mukaan ja tavoitteena on ottaa huomioon kaikki mahdolliset tilanteet, joihin sovelluksessa voi joutua. Ulkopuolisista testaajista osa saattaa saada erillisen ohjeistuksen testaukseen, mutta testaamassa on aina oltava myös ohjeistamattomia henkilöitä. Testauksessa ei ilmaantunut korjaamattomia loogisia ongelmia tai huomioimattomia virhetilanteita.

Lopuksi osaksi testausta kuuluu analytiikan toimivuus. Tämä toteutettiin niin, että Googlen Analytics -palveluun luotiin erikseen tili sovelluksen testausta ja varsinaista käyttöä varten. Testitili avattiin käyttöön hiukan ennen kehityksen loppua ja sen saama data tarkastettiin ennen sovelluksen julkaisua.

5.3.5 Julkaisu

Juuri ennen julkaisua sovelluksen sisältötekstit tarkastettiin ja tietokannasta tyhjennettiin kaikki testidata sekä nollattiin palkintotiedot. Lisäksi Google Analytics -tili vaihdettiin sovelluksen varsinaiseen käyttötiliin, käytiin vielä checklist tarkastettavista asioista läpi, jonka jälkeen Facebookin sovellushallinnassa nostetaan sovellus pois kehittäjien ”hiekkalaatikkotilasta”, jolloin se näkyy julkisesti yrityksen sivun käyttäjille. Julkaisun aikoihin ja tovi siitä eteenpäin päivystimme sovelluksen toimintaa, sinne virtaavia käyttäjiä ja dataa sekä sen oikeellisuutta.

6 JOHTOPÄÄTÖKSET JA POHDINTA

6.1 Toteutus

Toteutustyö sujui lopulta melko jouhevasti, koska käytetyt tekniikat olivat ennalta tuttuja. Into-Digitalilta löytyvä laaja kokemus Facebook-toteutuksista, sovelluksista ja kampanjoista on hyvä voimavara ja tuo pohjalle omat

rutiininsa sekä hyvän toimintavarmuuden, josta on ollut helppo ammentaa. Toteutusta auttoi osaltaan huomattavasti myös etukäteen tehty selkeä ja tarkka määrittely sovelluksen toiminnasta sekä kesken sovelluskehityksen vaadittujen muutostöiden huomattavan vähäinen määrä. Usein suurimmat ongelmat sovelluskehityksessä johtuvat joko puutteellisesta määrittelystä tai jo laaditun suunnitelman ja toimintalogiikan muuttamisesta kesken kehityksen. Jatkuva kaksisuuntainen viestintä asiakasrajapinnassa piti osapuolet ajan tasalla.

Testauksessa käytetyt keinot on ennestään muissa vastaavan kokoluokan projekteissa havaittu toimiviksi ja riittäviksi, eikä niiden voi katsoa epäonnistuneen. Teknisiä ongelmia sovelluksen elinkaaren aikana ei lopulta ilmennyt ja tähän pohjaten voi katsoa sovelluksen onnistuneen teknisesti sekä täyttäneen sille asetetut tekniset vaatimukset.

6.2 Kampanja

Into-Digitalin ja asiakasyrityksen asettamat tavoitteet kampanjan osalta toteutuivat lopulta niinikään hyvin. Mediatoimistojen arvioiden mukaan yhden tykkääjän hankintahinta on noin kolme euroa, ja tavoiteasetanta perustui siihen, että kampanja pääsisi alle yhden euron tasolle per hankittu tykkääjä. Tavoite täyttyi ja lopputulos asettuu 60-70 sentin haarukkaan tykkääjämäärän suuruusluokan noustessa kymmeneen tuhansiin.

Kampanjan onnistumisen taustalla vaikuttaneiksi syiksi voinee mainita lisäksi asiakasyrityksen maineen ennalta, kampanjan onnistuneet tukitoimenpiteet ja riittävän kampanja-ajan mahdollistama viraaliefektin käynnistyminen. Kampanjan tukitoimenpiteet määriteltiin, suunniteltiin ja sovittiin jo ennalta aiempiin kokemuksiin pohjautuen. Myös selkeä suunnitelma koko lanseerauksesta ylipäättään kampanjaa suuremmassa kontekstissa ja pidemmällä aikavälillä auttoi suuntaamalla voimavaroja ja uusia tavoitteita eteenpäin jo ennen kampanjaa ja sen aikana.

Jälkiviisaana voi todeta, että kampanjan perusidea oli riittävän yksinkertainen ollen samalla julkaisuvaiheessa melko uniikki. Sovelluksen käyttäminen tai

kampanjaan osallistuminen ei vienyt liiaksi ihmisten aikaa omaten silti ns. koukuttavia elementtejä ja riittävästi houkuttimia saaden ihmiset sekä käyttämään sitä kerran että palaamaan sovellukseen uudestaan.

Toteutettu sovellus ja kampanjatyyppe on hyvä pohja uudelle toteutukselle ja seuraava versio on jo suunnitteilla. Vastaavantyyllisiä toteutuksia on alkanut näkyä muuallakin, joten sitäkin voinee pitää eräänlaisena onnistumisen mittarina.

LÄHTEET

Aalto, E. 2012. CEO. Into-Digital Oy. Keskustelu 20.2.2012

Browser Statistics. 2013. Viitattu 14.4.2013.
http://www.w3schools.com/browsers/browsers_stats.asp, W3Schools

Hintikka, K. n.d. Sosiaalinen media. Viitattu 10.4.2012.
<http://kans.jyu.fi/sanasto/sanat-kansio/sosiaalinen-media>, Jyväskylän yliopisto, Kansalaisyhteiskunnan tutkimusportaali

Kaushik, A. 24.9.2012. Facebook Advertising / Marketing: Best Metrics, ROI, Business Value. Viitattu 26.1.2013. <http://www.kaushik.net/avinash/facebook-advertising-marketing-best-metrics-roi-business-value/>, Occam's Razor -blogi

Key Facts. 2012. Viitattu 2.5.2012. <http://newsroom.fb.com/content/default.aspx?NewsAreaId=22>, Facebook

Kinnunen, T. 28.4.2010. Mitä on Web 2.0? Viitattu 12.4.2012.
<http://www.internetmarkkinointi.fi/mita-on-web-2-0/>

Login Architecture. 2013. Viitattu 24.4.2013.
<https://developers.facebook.com/docs/concepts/login/login-architecture/>, Facebook

Lähdevuori, J. 2012. Suomalaisten Facebook-sivujen tila 2012. Viitattu 10.4.2012.
http://www.hkstrategies.fi/files/hillknowlton/materiaalit/Suomalaisten_Facebook-sivujen_tila_2012.pdf, Hill+Knowlton Strategies

O'Reilly, T. 30.9.2005. What Is Web 2.0. Viitattu 28.4.2012.
<http://oreilly.com/web2/archive/what-is-web-20.html>, O'Reilly Media Inc.

Pyhäjärvi, M. 20.5.2011. Taktinen Facebook-markkinointi on start-up yrittäjän paras kaveri. Viitattu 12.4.2012.
http://www.speakersforum.fi/blogi/sisalto/taktinen_facebook-markkinointi_on_start-up_yrittaejaen_paras_kaveri/, Speakersforum-blogi

Pönkä, H. 2011. Facebookissa ei ole 2 miljoonaa suomalaista. Paljonko sitten? Viitattu 10.4.2012. <http://harto.wordpress.com/2011/07/22/facebookissa-ei-ole-2-miljoonaa-suomalaista-paljonko-sitten/>, Lehmätkin lentäis -blogi

Suomalaisella on verkossa 138 kaveria, japanilaisella vain 28. 2010. Viitattu 10.4.2012. <http://www.hs.fi/ulkomaat/artikkeli/Suomalaisella+on+verkossa+138+kaveria+japanilaisella+vain+28/1135260796926>, Helsingin Sanomat

Terdiman, D. 15.12.2005. Study: Wikipedia as accurate as Britannica. Viitattu 28.4.2012. http://news.cnet.com/2100-1038_3-5997332.html, CNET News

The 1000 most-visited sites on the web. 2011. Viitattu 10.4.2012. <http://www.google.com/adplanner/static/top1000/>

The Login Dialog. 2013. Viitattu 1.4.2013. <https://developers.facebook.com/docs/concepts/login/permissions-login-dialog/>, Facebook

TV-mittaritutkimuksen tuloksia. 2012. Viitattu 10.4.2012. <http://www.finnpanel.fi/tulokset/tv/kk/share/2012/2/10plus.html>, Finnpanel

Usage of traffic analysis tools for websites. 28.4.2012. Viitattu 28.4.2012. http://w3techs.com/technologies/overview/traffic_analysis/all, W3Techs