



TAMPEREEN TEKNILLINEN YLIOPISTO

**VILLE KENTTA**  
**WEB-SOVELLUKSEN MUUNTAMINEN MOBIILIIYHTEEN-**  
**SOPIVAKSI, CASE: SURVEYPAL**

Diplomityö

Tarkastaja 1: Jukka Huhtamäki  
Tarkastaja 2: Ossi Nykänen  
Tarkastajat ja aihe hyväksytty  
Tietotekniikan tiedekuntaneuvoston  
kokouksessa 5.11.2014

# TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO

Tietotekniikan koulutusohjelma

**VILLE KENTTA: WEB-SOVELLUKSEN MUUNTAMINEN MOBIILIIHTEEN-SOPIVAKSI, CASE: SURVEYPAL**

Diplomityö, 54 sivua, 48 liitesivua

Helmikuu 2015

Pääaine: Hypermedia

Tarkastajat: Jukka Huhtamäki, Ossi Nykänen

Avainsanat: mobiili, web-sovellus, mukautuminen

Tässä työssä tutkitaan, mitä toteutustapaa Surveypal-sovelluksen mobiiliystävällisemmäksi saattamisessa tulisi hyödyntää. Työssä avataan web-sovellusten erityispiirteitä yleisesti ja mobiililaitteiden yhteydessä. Ongelman ratkaisemiseksi esitellään kaksi vaihtoehtoista toteutustapaa, erilliset toteutukset (tietokone- ja mobiilitoteutus) ja yksi, mukautuva toteutus. Eri toteutustavoille löydetään erilaisia hyviä ja huonoja puolia. Tutkimus osoittaa erillisen mobiilitoteutuksen olevan parempi ratkaisu tilanteessa, jossa mobiilikäyttäjillä on eri tarpeet, kuin tietokonekäyttäjillä ja mukautuvan toteutuksen vastaavasti tilanteessa jossa käyttäjällä voidaan olettaa olevan samat tarpeet kontekstista riippumatta.

Eri toteutustapojen suhtautumista *World Wide Web Consortium*:in määrittelemiin, mobiilien web-sovellusten parhaisiin käytäntöihin arvioidaan erityisesti suositusten toteuttamiseen vaaditun työpanoksen osalta. Surveypal-sovelluksen nykyisen toteutus osoittautuu hyväksi pohjaksi mukautuvalle toteutukselle ja kokonaan alusta tehtävä erillinen mobiilitoteutus todetaan huomattavasti työläemmäksi vaihtoehdoksi.

Surveypal-käyttäjille esitetyn kyselyn vastauksista käy ilmi, että vain 17 % kyselyyn vastanneista käyttäjistä oli käyttänyt sovellusta mobiililaitteilla, mutta käyttäjät suorittavat yli puolet yleisestä internetselailustaan mobiililaitteilla. Käyttäjät arvioivat sovelluksen eri ominaisuudet keskimäärin tärkeiksi myös mobiilissa kontekstissa, mutta pitivät tulosten katselua kuitenkin muita selkeästi tärkeämpänä.

Arvioinnin pohjalta saatuja tuloksia käytetään argumentteina muodostettaessa suositusta mobiilitoteutuksen toteutustavasta. Suositelluksi toteutustavaksi päätyy Surveypal-sovelluksen nykytilan pohjalta toteutettava mukautuva toteutus. Tärkeimpinä perusteluina mukautuvan toteutuksen puolesta toimivat sen toteuttamiseen ja ylläpitoon vaadittava pienempi työmäärä sekä Surveypal-käyttäjien selkeät tarpeet käyttää sovelluksen kaikkia ominaisuuksia myös mobiilissa kontekstissa.

## ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY

Master's Degree Programme in Information Technology

**VILLE KENTTA: MAKING A WEB-APPLICATION MORE MOBILE-FRIENDLY,  
CASE: SURVEYPAL**

Master of Science Thesis, 54 pages, 48 Appendix pages

February 2015

Major: Hypermedia

Examiner: Jukka Huhtamäki, Ossi Nykänen

Keywords: mobile, web-application, adaptation

In this thesis, a method for implementing more mobile-friendly version of SurveyPal-application is studied. General and mobile specific characteristics of web-application are covered. To solve the problem, two options for the implementation are introduced: separate implementations (for mobile and desktop devices) and singular responsive implementation. Advantages and disadvantages are found for both of the options. The study suggests that separate implementations are a better option when mobile and desktop users have different needs. In contrast, responsive implementation is found to be a better option when the users have the same needs, despite of the context.

Mobile web application best practices (by *World Wide Web Consortium*) are used to evaluate the two implementation methods. In the evaluation, especially the resources needed for an option to fulfill the specified best practices are taken into account. The current state of the SurveyPal-application is found to be a good base for a responsive implementation. The separate mobile implementation is found to be considerably more laborious to implement.

The survey targeted at SurveyPal-users showed that only 17% of the users had used the application with a mobile device. The same users said that they use the internet more with mobile than other devices. The results also show that the users consider all the main features of the application important also in the mobile context, inspecting results being clearly the most important feature.

The results received from the evaluation are being used as arguments while forming the suggestion for the method of the mobile implementation. A responsive implementation, based on the current implementation of the SurveyPal-application is suggested. The most important arguments for the responsive implementation include the smaller amount of work needed and the users' need for the ability to use all of the features of the application also with the mobile devices.

## ALKUSANAT

Haluan kiittää Surveypal Oy:tä mahdollisuudesta liittää diplomityöni tuotantokäytössä olevaan Surveypal-sovellukseen ja siihen liittyvään ongelmaan. Tutkimuksen kohteen tarjoamisen lisäksi haluan kiittää Surveypal Oy:tä lähdeaineistojen tarjoamisesta sekä mahdollisuudesta kirjoittaa diplomityötä varsinaisen kehitystyön ohella. Haluan kiittää myös diplomityöni ohjaajia Jukka Huhtamäkeä ja Ossi Nykästä, jotka antoivat paljon ideoita, rakentavaa kritiikkiä ja motivaatiota diplomityön kirjoituksen eri vaiheissa.

Tämän työn aloittamisen ja loppuun saattamisen välissä ehti tapahtua paljon. Muutimme vaimoni kanssa ensimmäiseen omaan asuntoomme, kaunis ja loputon ilo säteilevä esikoistyttäremme syntyi sekä rakas isoäitini nukkui pois. Tämän työn tekeminen ei olisi ollut mahdollista ilman vaimoani Aliisaa, joka auttoi minua jaksamaan kaiken tämän sekä normaalia kiireisemmän arjen läpi.

Lopuksi haluan kiittää kaikkia perheenjäseniäni ja ystäviäni, jotka ovat olleet mukana opintaipaleellani ja auttaneet suoraan tai johdannaaisesti tämän pisteen saavuttamisessa.

# SISÄLLYS

1. Johdanto . . . . .	1
1.1 Tutkimuksen ongelma . . . . .	1
1.2 Dokumentin rakenne . . . . .	2
2. Teoria . . . . .	3
2.1 Web-ohjelmat . . . . .	3
2.1.1 Alustariippumattomuus . . . . .	3
2.1.2 Tuntelemattomat päätelaitteet . . . . .	4
2.1.3 Laitteistojen asettamat rajoitteet . . . . .	4
2.1.4 Ohjelmistojen asettamat rajoitteet . . . . .	6
2.2 Mobiili web . . . . .	6
2.2.1 Mobiiliselailun erityispiirteet . . . . .	6
2.2.2 Tietokoneelle optimoidun verkkosivun käyttö mobiililaitteella . . . . .	8
3. Erilaisiin päätelaitteisiin mukautuminen . . . . .	10
3.1 Erilliset optimoidut toteutukset eri päätelaitteille . . . . .	10
3.1.1 Case: breastcancer.org . . . . .	11
3.1.2 Laitetyypit . . . . .	12
3.1.3 Laitteen tunnistaminen . . . . .	14
3.1.4 Laitteen erityispiirteiden huomioiminen . . . . .	15
3.2 Yksi, kaikkiin päätelaitteisiin mukautuva toteutus . . . . .	19
3.2.1 Case: Cyber-Duck . . . . .	20
3.2.2 Mukautuva verkkosuunnittelu . . . . .	22
4. Aineisto ja menetelmä . . . . .	28
4.1 Aineisto: Surveypal-kyselytyökalu . . . . .	28
4.1.1 Yleiskuvaus . . . . .	28
4.1.2 Teknologiat . . . . .	31
4.2 Menetelmä . . . . .	33
5. Arviointi . . . . .	35
5.1 Mobiilien web-sovellusten parhaat käytännöt . . . . .	35
5.1.1 Sovellusdata . . . . .	36
5.1.2 Turvallisuus ja yksityisyys . . . . .	36
5.1.3 Käyttäjän tietoisuus ja hallinta . . . . .	37
5.1.4 Resurssien hillitty käyttö . . . . .	37
5.1.5 Käyttökokemus . . . . .	39
5.1.6 Erilaisten toimituskontekstien käsittely . . . . .	40
5.1.7 Muuta huomioitavaa . . . . .	41
5.1.8 Yhteenvedo . . . . .	41
5.2 Surveypal-käyttäjien kanta sovelluksen mobiilikäyttöön . . . . .	43

6. Suositus Surveypal web-ohjelman mobiilitoteutuksen toteutustavasta . . .	50
7. Yhteenveto . . . . .	53
Lähteet . . . . .	55
A. Liite - Surveypal - kuvankaappaukset (työpöytätietokone) . . . . .	59
B. Liite - Surveypal - kuvankaappaukset (älypuhelin) . . . . .	65
C. W3C:n suositukset . . . . .	70
C.1 Sovellusdata . . . . .	71
C.2 Turvallisuus ja yksityisyys . . . . .	72
C.3 Käyttäjän tietoisuus ja hallinta . . . . .	73
C.4 Resurssien hillitty käyttö . . . . .	75
C.5 Käyttökokemus . . . . .	81
C.6 Erilaisten toimituskontekstien käsittely . . . . .	90
C.7 Muuta huomioitavaa . . . . .	95
D. Liite - Kysely: Surveypal mobiili . . . . .	97
E. Liite - Kyselyn tulokset: Surveypal mobiili . . . . .	102

## KUVAT

2.1	Tilasto internetselailuun käytettyjen tietokoneiden ja mobiililaitteiden näyttöresoluutioista (StatCounter [9]) . . . . .	5
2.2	danskebank.fi Android 4.1.2 selaimessa perustilassaan ja zoomattuna	8
3.1	Tilasto internetselailuun käytettyjen tietokoneiden näyttöresoluutioista (StatCounter [9]) . . . . .	13
3.2	Tilasto internetselailuun käytettyjen mobiililaitteiden näyttöresoluutioista (StatCounter [9]) . . . . .	13
3.3	fi.wikipedia.org etusivu Google Chrome -selaimessa PC-tietokoneessa	15
3.4	fi.m.wikipedia.org etusivu Firefox for Android -selaimessa älypuhelimessa perustilassaan ja navigointivalikko avattuna . . . . .	16
3.5	Rakenteeltaan sama sisältö esitettynä selaimen oman ikkunan ( <i>viewport</i> ) leveydellä (koodiesimerkki 3) sekä 980 pikselin leveydelle (oletus) emuloituna Samsung Galaxy S3 Mini älypuhelimella . . . . .	17
3.6	<i>Cyber-Duck</i> :in verkkosivun uusi toteutus ( <a href="http://www.cyber-duck.co.uk">http://www.cyber-duck.co.uk</a> ) älypuhelimien, tabletin ja tietokoneen verkkoselaimessa . . . . .	21
3.7	Sama <i>HTML</i> -dokumentti (koodiesimerkki 4) Motorola Moto G älypuhelimella pysty- ja vaakasuorassa sekä tietokoneella . . . . .	27
4.1	SurveyPal - aloitusnäky . . . . .	29
4.2	SurveyPal - kyselyn luominen ja muokkaus . . . . .	30
4.3	SurveyPal - kyselyn tulokset . . . . .	31
4.4	<i>Issue-Based Information System (IBIS)</i> -metodin notaatio . . . . .	34
5.1	Tulokset: Oletko käyttänyt SurveyPal -sovellusta jollain mobiililaitteella, kuten tablet-tietokoneella tai älypuhelimella? (taulukko E.1) . . . . .	43
5.2	Tulokset: Millä mobiililaitteella voisit kuvitella käyttäväsi SurveyPal-sovellusta? (Taulukko E.5) . . . . .	45
5.3	Tulokset: Arvio, miten tärkeitä SurveyPal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - <b>Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen)</b> (taulukko E.6) . . . . .	46
5.4	Tulokset: Arvio, miten tärkeitä SurveyPal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - <b>Kyselyn luominen ja muokkaaminen</b> (taulukko E.7) . . . . .	46
5.5	Tulokset: Arvio, miten tärkeitä SurveyPal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - <b>Kyselyn julkaiseminen</b> (taulukko E.8) . . . . .	47

5.6	Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - <b>Vastausten kerääminen</b> (kutsujen lähettäminen ja kyselylinkin jakaminen) (taulukko E.9) . . . . .	47
5.7	Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - <b>Tulosten katselu</b> (taulukko E.10) . . . . .	48
5.8	Tulokset: Arvioi, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä (taulukot E.6 - E.10) . . . . .	49
6.1	Suositus Surveypal-sovelluksen mobiilituen toteutustavasta perusteltuna <i>IBIS</i> -metodin avulla 1/2 . . . . .	51
6.2	Suositus Surveypal-sovelluksen mobiilituen toteutustavasta perusteltuna <i>IBIS</i> -metodin avulla 2/2 . . . . .	52
A.1	Surveypal - kirjautuminen . . . . .	59
A.2	Surveypal - aloitusnäky . . . . .	60
A.3	Surveypal - kyselyn luominen ja muokkaus . . . . .	60
A.4	Surveypal - kyselyn esikatselu . . . . .	61
A.5	Surveypal - kyselyn teema . . . . .	61
A.6	Surveypal - kyselyn mobiili teema . . . . .	62
A.7	Surveypal - vastausten kerääminen . . . . .	62
A.8	Surveypal - kyselyyn saadut vastaukset . . . . .	63
A.9	Surveypal - kyselyn tulokset aloitusnäkyssä . . . . .	63
A.10	Surveypal - kyselyn tulokset . . . . .	64
B.1	Surveypal - kirjautuminen . . . . .	65
B.2	Surveypal - aloitusnäky . . . . .	65
B.3	Surveypal - kyselyn luominen ja muokkaus . . . . .	66
B.4	Surveypal - kyselyn esikatselu . . . . .	66
B.5	Surveypal - kyselyn teema . . . . .	67
B.6	Surveypal - kyselyn mobiili teema . . . . .	67
B.7	Surveypal - vastausten kerääminen . . . . .	68
B.8	Surveypal - kyselyyn saadut vastaukset . . . . .	68
B.9	Surveypal - kyselyn tulokset aloitusnäkyssä . . . . .	69
B.10	Surveypal - kyselyn tulokset . . . . .	69
C.1	Surveypal - Tilin luonti itsepalveluna (käyttäjä hyväksyy käyttöehdot ja rekisteriselosteen viimeistelemällä tilinsä luonnin) . . . . .	74
C.2	Surveypal - CRM:n kautta luodun tilin aktivointi (Käyttäjän on hyväksyttävä käyttöehdot ja rekisteriseloste tilin aktivoinnin yhteydessä) . . . . .	74
C.3	Spinner Surveypal-sovelluksessa . . . . .	83



D.1	Kysely: Surveypal-mobiili, sivu 1 . . . . .	98
D.2	Kysely: Surveypal-mobiili, sivu 2 (vastaaja on vastannut myöntävästi ensimmäisen sivun kysymykseen (kuva D.1)) . . . . .	99
D.3	Kysely: Surveypal-mobiili, sivu 3 . . . . .	100
D.4	Kysely: Surveypal-mobiili, sivu 4 . . . . .	101

## TAULUKOT

5.0	Tässä luvussa esiintyvien taulukoiden rakenne . . . . .	35
5.1	W3C-suositukset: Sovellusdata (liite C.1) . . . . .	36
5.2	W3C-suositukset: Turvallisuus ja yksityisyys (liite C.2) . . . . .	36
5.3	W3C-suositukset: Sovellusdata (liite C.3) . . . . .	37
5.4	W3C-suositukset: Sovellusdata (liite C.4) . . . . .	38
5.5	W3C-suositukset: Käyttökokemus (liite C.5) . . . . .	39
5.6	W3C-suositukset: Erilaisten toimituskontekstien käsittely (liite C.6) .	40
5.7	W3C-suositukset: Muuta huomioitavaa (liite C.7) . . . . .	41
5.8	Suositusryhmien yhteispisteet . . . . .	42
C.0	Tässä liitteessä esiintyvien taulukoiden rakenne . . . . .	70
C.1	W3C-suositus: Käytä evästeitä säästeliäästi . . . . .	71
C.2	W3C-suositus: Käytä asiakaspään tietovarastoteknologioita lokaalille datalle . . . . .	71
C.3	W3C-suositus: Toista lokaali data . . . . .	72
C.4	W3C-suositus: Älä suorita suodattamatonta tai epäluotettua <i>JSON</i> - dataa . . . . .	72
C.5	W3C-suositus: Varmista, että käyttäjää informoidaan henkilökohtai- sen ja laitteeseen liittyvän informaation käytöstä . . . . .	73
C.6	W3C-suositus: Mahdollista automaattinen sisäänkirjautuminen . . . .	75
C.7	W3C-suositus: Pakkaa siirrettävät resurssit . . . . .	75
C.8	W3C-suositus: Minimoi sovelluksen ja datan koko . . . . .	76
C.9	W3C-suositus: Vältä uudelleenohjauksia . . . . .	76
C.10	W3C-suositus: Optimoi verkkopyynnöt . . . . .	77
C.11	W3C-suositus: Minimoi ulkopuoliset resurssit . . . . .	77
C.12	W3C-suositus: Yhdistä staattiset kuvat yhdeksi komposiittiresurssik- si ( <i>Sprite</i> ) . . . . .	78
C.13	W3C-suositus: Sisällytä taustakuvat CSS-tyylitömäärittelyjen sisään .	78
C.14	W3C-suositus: Välimuistita resurssit merkitsemällä resurssin viite . .	79
C.15	W3C-suositus: Välimuistita AJAX-data . . . . .	79
C.16	W3C-suositus: Älä lähetä evästetietoja tarpeettomasti . . . . .	80
C.17	W3C-suositus: Pidä DOM:in koko kohtuullisena . . . . .	80
C.18	W3C-suositus: Optimoi sovelluksen käynnistymisnopeus . . . . .	81
C.19	W3C-suositus: Minimoi havaittu latenssi . . . . .	82
C.20	W3C-suositus: Varaudu erilaisiin interaktiometodeihin . . . . .	84
C.21	W3C-suositus: Säilytä fokus päivitettäessä sivua dynaamisesti . . . .	85

C.22	W3C-suositus: Käytä osiotunnisteita ( <i>fragment ID</i> ) ohjaamaan sovel- luksen näkymiä . . . . .	85
C.23	W3C-suositus: Tee puhelinnumeroista ”Klikkaa soittaaksesi” -painikkeita	86
C.24	W3C-suositus: Varmista tekstikappaleiden sujuvuus . . . . .	86
C.25	W3C-suositus: Varmista, että laitteiden välinen tila on yhdenmukainen	87
C.26	W3C-suositus: Huomioi web-sovellusten käynnistämiseen soveltuvat mobiili-spesifit teknologiat . . . . .	88
C.27	W3C-suositus: Käytä <i>meta viewport</i> -elementtiä toivotun näyttöreso- luution tunnistamiseen . . . . .	89
C.28	W3C-suositus: Suosi palvelinpään tunnistusta niiltä osin kuin se on mahdollista . . . . .	90
C.29	W3C-suositus: Käytä asiakaspään tunnistusta kun tarpeellista . . . .	91
C.30	W3C-suositus: Käytä laitteiden luokittelua sisällön mukautumisen yksinkertaistamiseksi . . . . .	92
C.31	W3C-suositus: Tue ilman <i>JavaScript</i> :iä toimivaa variaatiota soveltu- vin osin . . . . .	93
C.32	W3C-suositus: Tarjoa käyttäjälle mahdollisuus valita käyttöliittymien välillä . . . . .	94
C.33	W3C-suositus: Harkitse canvas-elementin tai <i>SVG</i> :n käyttöä dynaa- miselle grafiikalle . . . . .	95
C.34	W3C-suositus: Informoi käyttäjää automaattisesta verkon käytöstä . .	95
C.35	W3C-suositus: Tarjoa riittävät mahdollisuudet hallita automaattista verkon käyttöä . . . . .	96
E.1	Tulokset: Oletko käyttänyt Surveypal -sovellusta jollain mobiililait- teella, kuten tablet-tietokoneella tai älypuhelimella? . . . . .	102
E.2	Tulokset: Minkä tyyppisellä mobiililaitteella käytit Surveypal-sovellusta?	102
E.3	Tulokset: Mikä käyttöjärjestelmä käyttämässäsi mobiililaitteessa oli? .	102
E.4	Tulokset: Kokemasi mobiilikäytön sujuvuus . . . . .	103
E.5	Tulokset: Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal- sovellusta? . . . . .	103
E.6	Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyiden hallinta (nimeäminen, ja- kaminen, poistaminen) . . . . .	104
E.7	Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyn luominen ja muokkaaminen . .	104
E.8	Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyn julkaiseminen . . . . .	105

E.9 Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Vastausten kerääminen (kutsujen lähettäminen ja kyselylinkin jakaminen) . . . . .	105
E.10 Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Tulosten katselu . . . . .	106
E.11 Tulokset: Sukupuolesi? . . . . .	106
E.12 Tulokset: Sukupuolesi? . . . . .	106
E.13 Tulokset: Arvioi, kuinka suuri osa yleisestä internetselailustasi tapahtuu mobiililaitteilla . . . . .	107

## TERMIT JA NIIDEN MÄÄRITELMÄT

Web-sovellus	Web-teknologioilla (HTML, CSS, JavaScript) toteutettu, verkkoselaimessa suoritettava sovellus.
Natiivisovellus	Sovellus, joka vaatii toimiakseen tietyn ohjelmisto- ja/tai laitealustan.
Tietokone	Työpöytätietokone, normaalikokoinen kannettava tietokone tai vastaava laite.
Työpöytälaite	ks. Tietokone.
Mobiililaite	(Äly)puhelin, taulutietokone tai vastaava pienikokoinen kannettava laite.
Käyttäjä	Sovellusta käyttävä henkilö.

# 1. JOHDANTO

## 1.1 Tutkimuksen ongelma

Internet-yhteyksien valtavan kehittymisen ja yleistymisen myötä myös internet-yhteyttä käyttävät sovellukset ovat jatkuvasti kasvattaneet suosiotaan. Kenties keskeisin esimerkki tällaisesta sovelluksesta on verkkoselain. Aiemmin verkkoselaimia oli pääasiassa ainoastaan tietokoneissa, mutta nykyään verkkoselain löytyy useimmista uusista puhelimista, tablet-tietokoneista, pelikonsoleista ja televisioista. Verkkoselain on paitsi sovellus, se toimii myös alustana sen sisällä ajettaville web-sovelluksille. Tämä tarjoaa ohjelmistojen kehittäjille mahdollisuuden kehittää helposti sovelluksia, jotka ovat saatavilla millä tahansa laitteella, jossa on internet-yhteys ja verkkoselain. Myös palvelun käyttäjän näkökulmasta tällainen sovellus tarjoaa huomattavia etuja, kuten sitomattomuuden tiettyyn päätelaitteeseen. Sama käyttäjä voi käyttää sovellusta helposti kotona television näytöllään, matkustaessaan älypuhelimellaan tai jollain julkisella päätteellä kirjastossa tai internet-kahvilassa.

Koska web-sovellus on yleisesti saatavissa millä tahansa verkkoselaimella, oletetaan sen yleisesti myös olevan käytettävissä kaikilla laitteilla, joissa on verkkoselain. Koska verkkoselain löytyy nykyisin hyvinkin eri tyyppisistä laitteista, asettaa se haasteita web-sovelluksen suunnittelussa ja toteutuksessa. Loppukäyttäjä voi käyttää sovellusta pienikokoisella ja -resoluutioisella mobiililaitteen näytöllä, perinteisellä tietokoneen näytöllä tai isokokoisella ja -resoluutioisella televisiolla. Näissä kaikissa tapauksissa on myös oletettavasti käytössä erilaiset syötelaitteet, kuten kosketusnäyttö, näppäimistö ja hiiri, peliohjain tai television kaukosäädin. Luonnollisesti pelkästään perinteisellä tietokoneella käytettäväksi suunniteltu web-sovellus ei ole käytettävyydeltään optimaalinen esimerkiksi pienikokoisella kosketusnäytöllä. Ongelma voidaan ratkaista toteuttamalla eri laitteille tai laitetyppeille optimoituja toteutuksia tunnistamalla ne erilaisin menetelmin. Toinen ratkaisuvaihtoehto on pyrkiä toteuttamaan web-sovellus, joka sopeutuu millaiseen päätelaitteeseen tahansa ja on käytettävissä näytön koosta ja resoluutiosta riippumatta.

Tämän tutkimuksen tapausesimerkkinä toimii Surveypal web-sovellus. Surveypal on kattava kyselytyökalu, jolla voi luoda monimuotoisia kyselyitä sekä kerätä ja analy-

soida vastauksia. Surveypal on suunniteltu pääosin pelkästään tietokoneella käytettäväksi, eikä sen käytettävyys muilla laitteilla ole välttämättä paras mahdollinen. Tutkimuksen tarkoituksena on selvittää, millä toteutustavalla Surveypal-sovellus kannattaisi muuttua mobiiliystävällisemmäksi. Argumentteja parhaan toteutustavan perustelemiseksi pyritään löytämään *World Wide Web Consortium*:in (*W3C*) määrittämän, mobiilien web-sovellusten parhaita käytäntöjä käsittelevän ohjeistuksen avulla. Argumentteja haetaan lisäksi myös Surveypal-käyttäjille esitetyn, sovelluksen mobiilikäyttöön keskittyvän kyselyn tuloksista.

Tutkimus on tarpeellinen, koska mobiilikäytön tukeminen on Surveypalille selvä kilpailuetu. Surveypal-sovelluksen kaikkien ominaisuuksien tukeminen mobiilissa kontekstissa saattaa kuitenkin olla tarpeetonta ja tämän vuoksi tutkimuksessa kartoitetaan myös asiakkaiden tarpeita. Lisäksi eri ratkaisuvaihtoehtojen punnitseminen on tärkeää, jotta voidaan saavuttaa mahdollisimman hyvä ratkaisu, joka ei vaadi liikaa resursseja, mutta ratkaisee ongelman tarvitulla tasolla.

## 1.2 Dokumentin rakenne

Tässä luvussa avataan tutkimuksen ongelma ja esitellään tämän dokumentin luvuissa käsiteltävät asiat. Luvussa 2 käsitellään web-ohjelmien erityispiirteitä, sekä web-ohjelmien kehitykseen ja käyttöön liittyviä mahdollisuuksia ja ongelmia. Viimeisenä käsitellään web-ohjelmien käyttökokemukseen vaikuttavia tekijöitä. Luvussa 3 käsitellään web-sovellusten mukautumista erilaisiin päätelaitteisiin. Mukautumisen keinoista perinteisempää, erillisiä optimoituja toteutuksia, käsitellään ensimmäisenä. Toisena keinona käsitellään universaalia, yksittäistä toteutusta, joka mukautuu kaikkiin päätelaitteisiin. Luvussa 4 esitellään tämän tutkimuksen materiaalina toimiva Surveypal-kyselytyökalu sekä tutkimuksen metodi. Luvussa käydään läpi työkalun yleiskuvaus, siinä käytetyt teknologiat sekä avataan hieman nykyisiä toteutusratkaisuja ongelmaan liittyen. Luvussa 5 käydään läpi eri toteutusvaihtoehtojen suhtautumista W3C:n suositukseen sekä Surveypal-käyttäjien mobiilikäyttötottumuksia, jotka selvitettiin kyselytutkimuksella. Luvussa 6 esitetään suositus Surveypal-sovelluksen mobiilitoteutuksen toteutustavasta.

## 2. TEORIA

### 2.1 Web-ohjelmat

Web-ohjelmat eroavat tietylle laitteelle tai käyttöjärjestelmälle suunnitelluista natiivisovelluksista monin tavoin. Tässä luvussa käydään läpi web-ohjelmien tärkeimmät mahdollisuudet ja rajoitteet, pääasiassa käyttökontekstin ja teknologian näkökulmista. Puhtaalla web-ohjelmalla tarkoitetaan tässä kontekstissa web-ohjelmaa, joka ei tarvitse asiakaspäässä verkkoselaimen lisäksi mitään liitännäisiä tai apuohjelmia toimiakseen. Puhumme siis HTML, CSS ja JavaScript -ohjelmointikielillä toteutetuista sovelluksista.

#### 2.1.1 Alustariippumattomuus

Yksi web-ohjelmien kiistattomista eduista verrattuna natiivisovelluksiin on sen alustariippumattomuus. Puhtaat web-ohjelmat suoritetaan käytännössä kokonaan irrallaan käyttöjärjestelmästä ja laitteistosta verkkoselaimen tarjoamassa hiekkalaatikossa. Tämä mahdollistaa ideaalitulanteessa täysin saman koodin tulkitsemisen ja suorittamisen kaikissa päätelaitteissa, joissa on verkkoselain. Puhtaat web-ohjelmat ovatkin siis oikein toteutettuna saavutettavuudeltaan merkittävästi parempia kuin perinteiset natiivisovellukset.

Web-sovellukset ja niiden laitteistoriippumattomuus mahdollistavat sovelluksien kehittämisen kaikille alustoille, tekemällä ainoastaan yhden toteutuksen [1, s. 1]. Täten sovelluksen potentiaalinen käyttäjäkunta on huomattavasti laajempi kuin natiivisovelluksien tapauksessa.

Web-ohjelmat tarjoavat monia mahdollisuuksia myös niiden käyttäjille. Käyttäjän ei tarvitse löytää sovelluksesta omalla käyttöjärjestelmällään toimivaa versiota tai asentaa mitään [1, s. 1]. Riittää, että käyttäjä tietää web-sovelluksen URL-osoitteen ja hänellä on käytössään laite, jossa on verkkoselain. Käytön aloittamisen lisäksi sovelluksen päivittäminen on käyttäjän näkökulmasta käytännössä huomaamaton tapahtuma oikein toteutettuna [15, s. 16].



Natiivisovelluksista poiketen web-sovellus ei myöskään ole sidottu mihinkään yksittäiseen laitteeseen, joten käyttäjä voi käyttää sovellusta useilla eri laitteilla. Käyttäjä voi käyttää samaa sovellusta helposti omalla PC:llään, älypuhelimellaan tai vaikka kaverinsa tablet-tietokoneella. Lisäksi käyttäjä voi teoriassa olettaa saman sisällön olevan saatavilla kaikissa näissä laitteissa.

Toisin kuin useiden natiivisovellusten tapauksessa, web-ohjelmien dataa säilytetään pääosin palvelimilla irrallaan käyttäjän päätelaitteesta. Tämä mahdollistaa muun muassa saman sisällön jakamisen käyttäjien kesken sekä yhteistyötä vaativien toimintojen suorittamisen helposti. Esimerkkinä tällaisesta web-sovelluksesta voimme mainita *Google Docs*:in, jonka avulla eri käyttäjät voivat muokata samaa dokumenttia samaan aikaan omilla päätelaitteillaan.

### 2.1.2 Tuntemattomat päätelaitteet

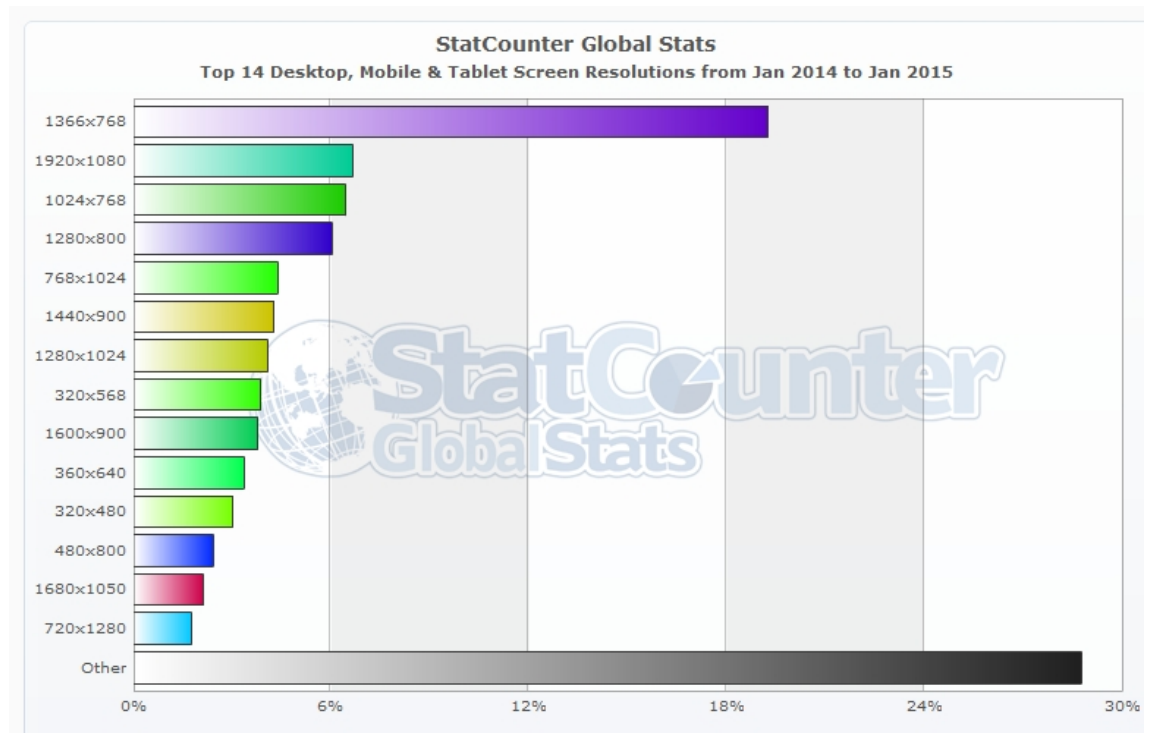
Vaikka alustariippumattomuus onkin pääosin positiivinen asia, tuo se mukanaan myös paljon haasteita. Toisin kuin natiivisovelluksien tapauksessa, web-sovelluksia kehitettäessä on varauduttava valtavaan määrään erilaisia päätelaitteita. Esimerkiksi natiivia Windows tai Android-sovellusta kehitettäessä kehittäjä pystyy tekemään huomattavan paljon turvallisia oletuksia käyttäjän laitteista. Sama etu ei päde web-sovellusta kehitettäessä, koska sen on toimittava molemmissa edellä mainituista ja lukuisissa muissa ympäristöissä.

Koska web-sovellusta on mahdollista käyttää lukuisissa erilaisissa ympäristöissä, sen käyttöliittymän kattava testaaminen voi osoittautua jopa mahdottomaksi tehtäväksi. Uusien laitteiden ja selainversioiden jatkuva lisääntyminen hankaloittaa testaamista entisestään.

### 2.1.3 Laitteistojen asettamat rajoitteet

Päätelaitteiden erilaiset laitteistot aiheuttavat yhdessä niiden käyttökontekstin kanssa paljon ongelmia, jotka on otettava huomioon suunniteltaessa web-sovellusta. Kenties suurimman haasteen asettavat päätelaitteiden fyysisesti ja resoluutioltaan eri kokoiset näytöt.

Näytön koot vaihtelevat alle kolmen tuuman mobiililaitteen näytöstä aina yli 65 tuumaisiin television näyttöihin. Vastaavasti myös suositut näyttöjen vaakaresoluutiot vaihtelevat 320 pikselistä aina 1920 pikseliin (Kuva 2.1) [7, s. 6]. Uusimmat 4K ja 8K Ultra HD televisiot nostavat potentiaalisesti hyödynnettävät vaakaresoluutiot jopa 3840 ja 7680 pikseliin [8].



Kuva 2.1: Tilasto internetselailuun käytettyjen tietokoneiden ja mobiililaitteiden näyttöresoluutioista (StatCounter [9])

Näytön koon ja resoluution suhteissa, pikselitiheyksissä, on myös huomattavia eroja. Esimerkiksi Full HD (1080x1920 pikseliä) resoluutiolla varustettuja laitteita löytyy älypuhelimesta (yli 400 pikseliä tuumalle) [10] isoon televisioon (alle 40 pikseliä tuumalle) [11]. Täten käyttäjä näkee eri näyttölaitteella myös sisällön eri kokoisena. Pikselitiheyksien eroja kompensoivana tekijänä voidaan joissain tapauksissa pitää näyttöjen eroavia katseluetäisyyksiä, mutta ei kuitenkaan aina.

Edellämainituista seikoista johtuen web-sovelluksen käyttöliittymäelementtien asetelu muodostuu ongelmalliseksi. On ilmeistä, ettei 320 ja 7680 pikseliä leveillä näyttöillä pystytä esittämään samaa sisältöä samalla asettelulla, ilman kompromissien tekemistä jomman kumman kustannuksella.

Koska web-ohjelman dataa säilytetään pääosin palvelimilla, on palvelimen ja päätelaitteen välillä oltava hyvä verkkoyhteys. Web-ohjelmia ei yleensä pystytä käyttämään lainkaan ilman verkkoyhteyttä. Tämä rajoite poistuu osittain HTML5:n tarjoamien ohjelmointirajapintojen myötä[2; 3], mutta muiden tuottaman sisällön noutamiseksi omalle päätelaitteelle tullaan jatkossakin tarvitsemaan verkkoyhteys.

### 2.1.4 Ohjelmistojen asettamat rajoitteet

Kuten aiemmin todettiin, web-ohjelma tarvitsee alustakseen verkkoselaimen. Vaikka web-ohjelmia voidaanakin pitää alustariippumattomina, ovat ne kuitenkin riippuvaisia verkkoselaimesta. Vaikka ideaalisessa tilanteessa web-ohjelmat toimisivat kaikissa verkkoselaimissa täysin identtisesti, ei asia kuitenkaan aina näin ole [4; 5]. Eri selaimet tukevat eri ominaisuuksia ja joissain tapauksissa tulkitsevat tukemiaan ominaisuuksia eri tavoin.

Uudet selaimet tuovat mukanaan uusia ominaisuuksia, joita web-ohjelmistojen kehittäjät mielellään käyttävät. Ongelmaksi muodostuu kuitenkin ominaisuuksien puuttuminen vanhemmista selainversioista, joilla saattaa olla hyvinkin laaja käyttäjäkunta. Tästä johtuen kehittäjä ei välttämättä pysty hyödyntämään uusimpia tekniikoita riskeeraamatta potentiaalista käyttäjäkunnan pienenemistä.

## 2.2 Mobiili web

Mobiililaitteiden käyttö verkon selailulaitteena on noussut huomattavasti vuoden 2007 jälkeen. Vuonna 2010 älypuhelin myynti ylitti PC-tietokoneiden (työpöytä ja kannettavat) myynnin määrällisesti [15, s. 7]. Vaikka mobiililaitte voi teoriassa suorittaa täysin saman web-ohjelman kuin tietokonekin, eroaa mobiiliselailu monella tapaa perinteisellä tietokoneella suoritetusta selailusta. Käsittelemme tässä luvussa mobiiliselailun erityispiirteitä teknologian ja käyttökokemuksen näkökulmasta. Mobiililaitteella tarkoitamme tässä kontekstissa verkkoselailuun kykenevää suhteellisen pienikokoista laitetta, kuten älypuhelin tai tablet-tietokonetta.

### 2.2.1 Mobiiliselailun erityispiirteet

Ollakseen mobiili, laitteen on luonnollisesti oltava suhteellisen pienikokoinen. Tästä johtuen myös laitteen näytön on oltava pienikokoinen. Ensimmäiset iOS-, Android- ja WebOS-pohjaiset älypuhelimet olivat näyttöresoluutioltaan 320 \* 480 pikseliä [15, s. 19]. Tämä on 80% vähemmän kuin kauan verkkosivujen suunnittelussa yleisesti käytetty minimiresoluutio, 1024 \* 768 pikseliä [15; 19; 22; 37; 12, s. 23]. Pienemmän koon lisäksi mobiililaitteiden näytöt ovat usein normaalissa käyttöasennossaan pystysuuntaisia (näytön korkeus on suurempi kuin sen leveys), toisin kuin valtaosassa tietokoneita. Edellä mainituilla resoluutioilla mitattuna mobiililaitteen vaakaresoluutio on n. 70 % pienempi vaakasuunnassa ja n. 40 % pienempi pystysuunnassa.

Pienentyneen resoluution lisäksi mobiililaitteiden syötelaitteet eroavat radikaalisti perinteisistä tietokoneen syötelaitteista (näppäimistö ja hiiri). Valtaosassa nykypäivän älypuhelimista ja tableteista syöte annetaan kosketusnäytön avulla, eli käytännössä koskettamalla laitteen näyttöä sormella [6; 15, s. 67]. Sormella annettu syöte on huomattavasti epätarkempi kuin hiirellä annettu [15, s. 68]. Linkki tai painike, joka on helposti klikattavissa hiirellä, voi osoittautua todella hankalaksi kosketusnäyttöä käytettäessä. Koska älypuhelimissa ei useimmissa tapauksissa ole erillistä näppäimistöä, on web-sovelluksen syötekentät syytä suunnitella tämä seikka huomioiden. Kosketusnäytöllä esitettävälle näppäimistölle mahtuu vain rajattu määrä merkkejä. Jos syötekenttään on tarkoitus antaa ainoastaan tietyn tyyppistä dataa, on tämä hyvä tarkentaa myös sovelluksen koodissa asettamalla syötekentälle tarkoituksenmukaisen syötteen ilmaiseva *type*-atribuutti (koodiesimerkki 1). Täten käyttäjälle voidaan näyttää esimerkiksi pelkkä numeronäppäimistö, helpottaen käyttäjän syötteen antamista.

```
<input type='number' />
```

Koodiesimerkki 1: Syötekentän tyyppin määrittäminen HTML5:ssä [16]

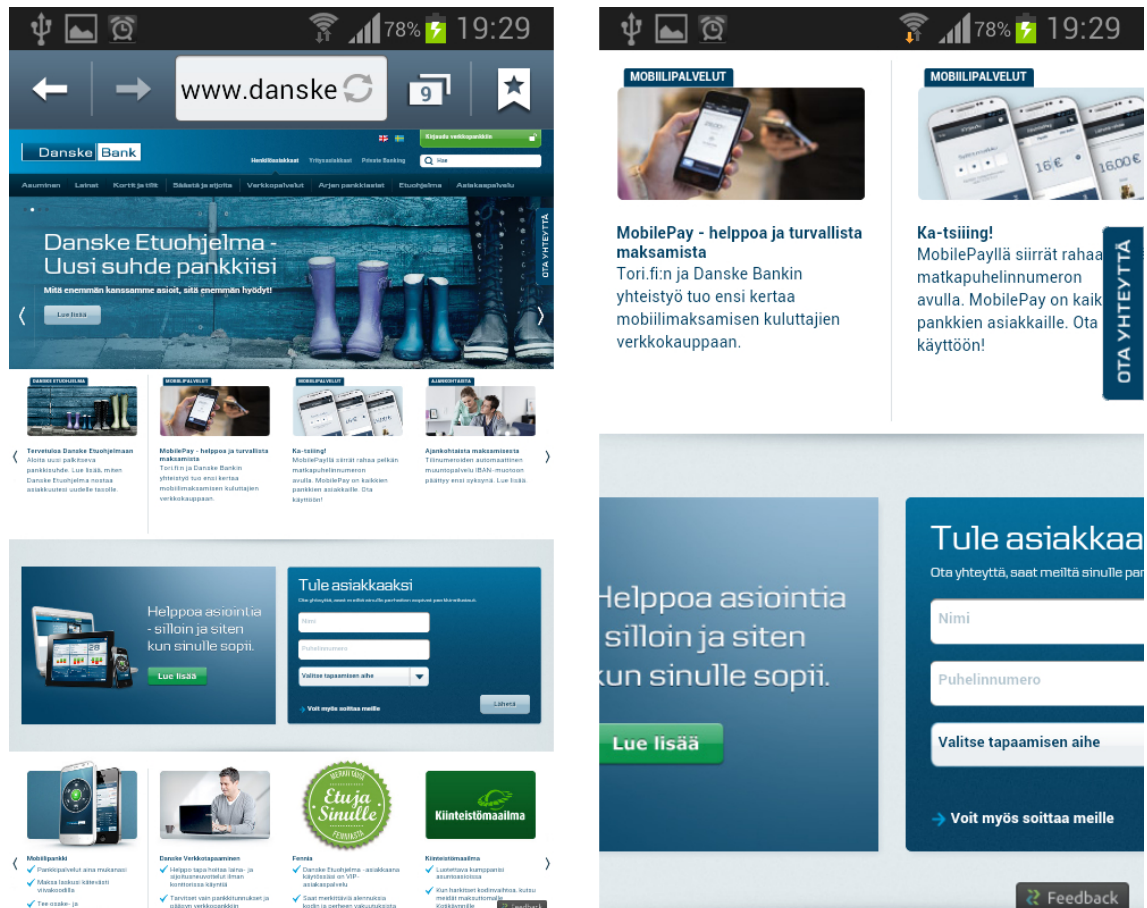
80% vähentynyt näyttötila, vaihtunut näytön orientaatio ja syötetapa pakottaa suunnittelijat priorisoimaan verkkosivulla esitettävän sisällön kokonaan uudelleen. Monimutkaisen navigointivalikon, isojen bannereiden tai mainosten sovittaminen pienelle näytölle ei yksinkertaisesti onnistu, mikäli verkkosivun käytettävyys halutaan säilyttää samalla tasolla kuin isommalla näytöllä käytettäessä.

Teknisten eroavaisuuksien lisäksi myös käyttökonteksti voi olla hyvinkin erilainen verrattaessa mobiiliselailua tietokoneella suoritettuun. Mobiililaitteet ovat lisäksi usein mobiilien verkkoteknologioiden varassa käyttökontekstissaan. Mobiili verkko-yhteys ei useimmiten ole yhtä laadukas kuin kaapelilla toteutettu ja etenkin sen latenssi on usein merkittävästi suurempi. Tästä johtuen isojen tiedostojen ja etenkin suurien tiedostomäärien lataaminen mobiililaitteella saattaa olla verrattaen hyvin hidasta.[15, s. 22]

Heikomman verkkoyhteyden lisäksi myös tilanteet, joissa mobiililaitteita käytetään, ovat yleensä erilaisia. Tietokonetta käytetään yleensä istualtaan pöydän ääressä ja pidempiä aikoja yhtäjaksoisesti. Mobiililaitteita sen sijaan voidaan helposti käyttää erilaisissa ympäristöissä, ja tästä johtuen myös keskittymisen taso ei ole välttämättä samaa luokkaa kuin tietokonetta käytettäessä. *Luke Wroblewski*:n sanoin mobiilikäyttäjät ovat ”yksi peukalo ja yksi silmä”. Yksi peukalo, koska ihmiset piteleivät älypuhelimia yhdellä kädellä käyttäen kosketusnäyttöä peukalollaan. Yksi silmä, koska mobiililaitteiden käyttökontekstissa käyttäjä keskittyy vain osittain laitteen käyttöön.[15, s. 25]

## 2.2.2 Tietokoneelle optimoidun verkkosivun käyttö mobiililaitteella

Useimmat web-ohjelmat on toteutettu ainoastaan tietokonekäyttöä silmällä pitäen. Ohjelman varaama koko näytöllä ja sen asettelu on suunniteltu toimimaan parhaiten tietyn kokoisella näytön resoluutiolla. Tällaisen sovelluksen käyttäminen esimerkiksi pienellä mobiililaitteen näytöllä aiheuttaa monia ongelmia käyttäjälle.



Kuva 2.2: danskebank.fi Android 4.1.2 selaimessa perustilassaan ja zoomattuna

Ellei web-sovelluksen koodissa ole huomioitu eri kokoisia näyttölaitteita, näkee käyttäjä jollain tavalla zoomatun näkymän sovelluksesta (Kuva 2.2). Useimmat mobiiliselaimet emuloivat näytön leveydeksi 980px, ellei verkkosivu esitä erillistä toivetta näyttöalueen leveydestä [12; 17; 18]. Tässä tapauksessa verkkosivun sisältö on kuitenkin yleensä liian pientä luettavaksi ja käyttäjä joutuu zoomaamaan itse verkkosivun sisältöä haluamansa kokoiseksi nähdäkseen sen paremmin. Suuremmaksi zoomattua sisältöä käyttäjä joutuu puolestaan vierittämään vaakasuuntaisesti. Sekä zoomaamisen, että horisontaalisen vierittämisen tarve ovat epätoivottuja, käytettävyyttä heikentäviä tekijöitä [13]. Zoomaaminen on yksinkertaisesti ylimääräinen toimenpide ja sekä vertikaalisen että horisontaalisen vierittämisen tarve hankaloittaa

koko verkkosivun kattamista, koska ruutua on vieritettävä kahteen ulottuvuuteen. Tämä puolestaan kuormittaa käyttäjän muistia.

Erilaisten syötelaitteiden huomioiminen on niin ikään tärkeää web-sovelluksen käytettävyyden kannalta. Kuten luvussa 2.2.1 mainittiin, on sormi (kosketusnäyttöä käytettäessä) huomattavasti summittaisempi syötelaite kuin esimerkiksi tietokonekäytössä useimmiten käytössä oleva hiiri ja sen kursori. Täten tietokonekäyttöön suunnitellut verkkosivun painikkeet voivat osoittautua liian pieniksi kosketusnäytökäytössä. Myös erilaisten kosketustapahtumien huomioiminen on tarpeellista liiallisen viiveen eliminoimiseksi käyttöliittymässä. Ilman erillistä käsittelyä kosketusnäytöllä varustetut laitteet tulkkavat ”klikkauksiksi” tarkoitetut näpäytykset oikein klikkauksiksi, mutta tekevät sen yleensä 300ms viiveellä. Viive johtuu mahdollisen tuplanäpäytyksen odottamisesta, jota käytetään yleensä näkymän zoomaamiseen mobiiliselaimissa [14].

### 3. ERILAISIIIN PÄÄTELAITTEISIIN MUKAUTUMINEN

Tässä luvussa käydään läpi erilaisia tapoja, joilla web-sovellus voi sopeutua päätelaitteeseen, jolla sitä käytetään. Pelkistettynä vaihtoehtoja sopeutumiseen on käytännössä kaksi.

Ensimmäinen, aiemmin suosituin toteutustapa, on toteuttaa erilliset optimoidut toteutukset eri päätelaitteille. Päätelaitteet jaotellaan tässä tapauksessa esimerkiksi työpöytä- ja mobiililaitteisiin. Eri laitetyyppien tunnistus tehdään palvelinpäässä ja palvelin lähettää asiakkaalle joko mobiili- tai työpöytätoimituksen riippuen siitä, kumman tyyppiseksi laitteeksi palvelin asiakkaan tulkitsee.

Toinen, tämän diplomityön kirjoitushetkellä nousevassa suosiossa oleva toteutustapa, on tehdä vain yksi toteutus, jota kaikki päätelaitteet käyttävät. Tämä yksittäinen toteutus tarkastelee asiakaspäässä päätelaitteen ominaisuuksia (esimerkiksi näytön resoluutio ja orientaatio tai syötetapa) ja mukautuu esimerkiksi päätelaitteen näytön resoluutioon asettamalla sille sopivan käyttöliittymäasettelun.

#### 3.1 Erilliset optimoidut toteutukset eri päätelaitteille

Tässä luvussa käsittelemme ratkaisumallia, jossa eri tyyppisille päätelaitteille tehdään omat toteutuksensa. Ratkaisumallissa verkkosivua selaava laite pyritään tunnistamaan palvelinpäässä, mikä mahdollistaa optimoidun toteutuksen tarjoamisen tunnistetun laitteen tyyppin mukaan. Vielä vuonna 2013 tämä ratkaisumalli oli kenties suosituin tapa toteuttaa verkkopalvelu, joka toimii erilaisilla laitteilla [7, s. 9].

Tässä luvussa esitellään case-esimerkki, joka tarjoaa esimerkkejä perusteluista ja toteutustavoista erillisiin optimoituihin toteutuksiin. Case-esimerkin lisäksi luvussa käydään läpi, millaisille laitetyypeille erillisiä toteutuksia yleisimmin tehdään, miten nämä laitetypit tunnistetaan ja miten laitteiden erityispiirteitä voidaan huomioida.

### 3.1.1 Case: breastcancer.org

*Derek Olson* kertoo artikkelissaan *Case Study from Breastcancer.org: Choosing between responsive web design and a separate mobile site to improve mobile visitors experience* [20], millaisiin toteutusteknisiin ratkaisuihin breastcancer.org verkkosivun uudelleensuunnittelussa päädyttiin. Uudelleensuunnittelun pääsyy oli mobiilikäytön lisääntyminen ja ongelman ratkaisuvaihtoehtoina pidettiin mukautuvaan verkkosuunnitteluun siirtymistä tai erillisen mobiilisivun luomista vanhan työpöytä sivun rinnalle.

Breastcancer.org on maailman vilkasliikenteisin rintasyöpään keskittyvä verkkosivu, jopa yli 1,5 miljoonalla kuukausittaisella kävijällä. Sivusto tarjoaa tietoa rintasyöpähoidoista ja niiden sivuvaikutuksista sekä rintasyövän oireista ja sen eri muodoista. Lisäksi sivusto tarjoaa ravitsemuksellisia ja liikunnallisia ohjeita, sekä keinoja pienentää sairastumisen riskiä. Pääsivuston lisäksi breastcancer.org tarjoaa keskustelufoorumin, joka kerää kuukausittain yli 600,000 vierailijaa ja jolla on yli 128,000 rekisteröityntä käyttäjää.

Kesällä 2011 sivuston kävijätilastoja tutkittaessa kävi ilmi, että breastcancer.org on päivitettävä tukemaan mobiilikäyttäjiä paremmin. Sivuston käytettävyyttä mobiililaitteilla päätettiin parantaa erityisesti viidellä eri osa-alueella. Vaakasuntaisen vierittämisen tarve oli poistettava, sivuston tekstin oli oltava riittävän isokokoista ilman zoomaamista, painikkeista oli tehtävä napautusystävälliset (kosketusnäyttö käyttöä ajatellen), sivuston latausnopeus on oltava lyhyt myös heikommilla mobiiliyhteyksillä ja sivuston navigointia sekä sisältöä oli yksinkertaistettava mobiilikäyttäjien tarpeisiin.

Sekä yksittäisen että erillisten toteutuksien hyviä ja huonoja puolia arvioitiin kattavasti, erityisesti breastcancer.org-sivuston näkökulmasta. Arvioinnin tuloksena erilliset toteutukset todettiin paremmaksi vaihtoehdoksi informatiivisen pääsivuston tapauksessa.

Sivuston tilastoja tutkittaessa kävi ilmi, että sivustolle mobiili- ja työpöytäselaimilla saapuvat käyttäjät etsivät sivustolta yleensä erilaista sisältöä. Lisäksi mobiilikäyttäjien ajateltiin olevan voimakkaammin riippuvaisia hyvästä navigointivalikosta, koska he saapuvat useimmiten sivuston etusivulle. Työpöytäkäyttäjät puolestaan saapuivat sivustolle todennäköisemmin hakukoneen kautta tarkempien hakusanojen avulla ja etsivät myös toissijaista informaatiota. Näiden tietojen valossa navigoinnin yksinkertaistaminen ja toissijaisen sisällön karsiminen mobiilissa asetettiin pääprioriteetti.



Breascancer.org:in pääsivusto oli toteutettu sisällönhallintajärjestelmällä, jonka mobiiliystävälliseksi muuntaminen olisi ollut kallista. Ratkaisuna päätettiin käyttää sivuston jo olemassa olevaa verkkosyötettä ja rakentaa sitä hyödyntävä kevyt front-end -ratkaisu ilman sisällönhallintajärjestelmää.

Breastcancer.org:in keskustelufoorumi toteutettiin sen sijaan yhtenä, mukautuvana toteutuksena. Ratkaisua perusteltiin muun muassa sillä, että foorumin käyttäjät ovat useimmiten niin sanottuja ”tehokäyttäjiä” ja he tarvitsevat foorumien täyden toiminnallisuuden myös mobiililaitteilla käytettäessä.

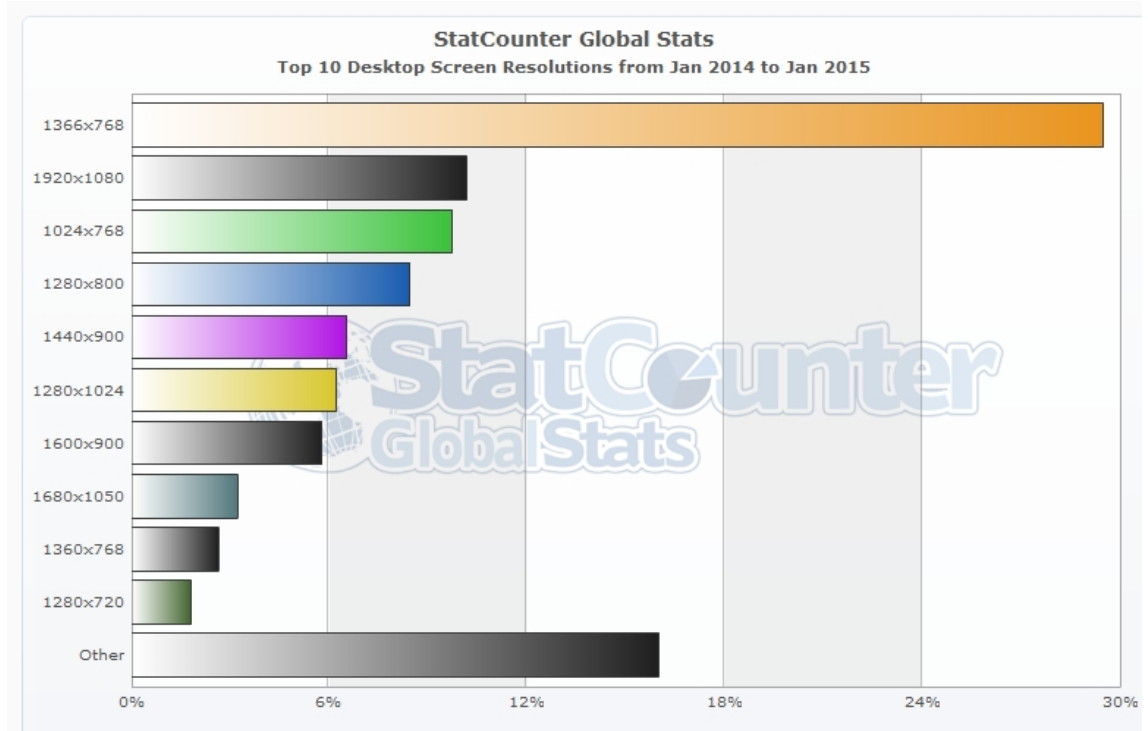
Uusien mobiiliystävällisten sivustojen julkaisun jälkeen suoritettut käytettävyysettit osoittivat mobiilikäyttäjien löytävän etsimänsä sisällön huomattavasti helpommin kuin aiemmin. Mobiilikäyttäjät olivat uudistuksen jälkeen tyytyväisempiä myös kävijätilastojen mukaan. Vaikka breastcancer.org:in käyttäjäkunnan voisi kuvitella käyttävän palvelua epätodennäköisemmin mobiililaitteella, kattaa mobiilikäyttäjät jopa kolmasosan kaikesta sivustolle tulevasta liikenteestä ja määrä on huomattavassa kasvussa.

### 3.1.2 Laitetyypit

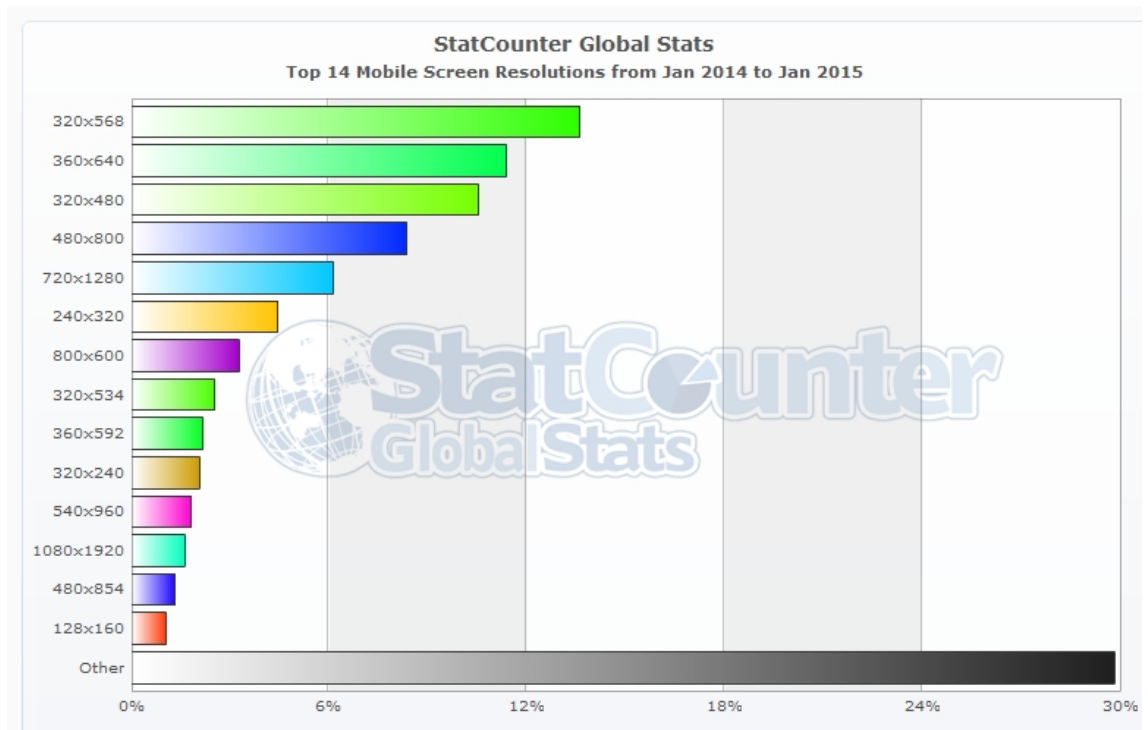
Aiemmin web-sovellusta suunniteltaessa oli kohtuullisen turvallista olettaa, että sitä tullaan käyttämään pääasiassa PC-tietokoneilla. Mobiililaitteet, kuten tablet-tietokoneet ja älypuhelimet ovat kuitenkin kasvattaneet suosiotaan verkkosivujen selailulaitteina. Nykypäivänä myös muissa laitteissa, kuten televisioissa ja pelikonsoleissa on usein verkkoselain, mutta niiden käyttö on ainakin toistaiseksi hyvin marginaalista.

Suunniteltaessa web-sovellusta PC-tietokoneelle oletetaan usein näytön resoluution olevan minimissään 1024 x 768 pikseliä [15, s. 18]. Tämän työn kirjoitushetkellä 1366 x 768 resoluutio oli selkeästi suosituin yli 27% osuudella (kuva 3.1 sivulla 13). Myös kaikki muut eriteltyt resoluutiot ovat yli 1000px leveydeltään, 1024px leveyden ollessa pienin, joskin toiseksi käytetyin resoluutio. Näin ollen voimme turvallisesti olettaa 1000px leveän web-sovelluksen mahtuvan lähes kaikille PC-tietokoneiden näytöille. PC-tietokoneiden syötelaitteiksi voidaan olettaa lähes poikkeuksetta hiiri ja näppäimistö. Tästä syystä myös käyttöliittymän painikkeista voidaan tehdä suhteellisen pienikokoisia hiiren hyvän tarkkuuden johdosta. Erillisen näppäimistön ollessa käytössä myös syötekentät ovat helposti suunniteltavissa, eivätkä ne vaadi mitään erikoishuomioita.

Kuten luvussa 2.2.1 todettiin, siirryttäessä suunnittelemaan web-sovellusta älypuhelimille, käsitellään huomattavan erilaista ympäristöä kuin PC-tietokoneen tapauk-



Kuva 3.1: Tilasto internetselailuun käytettyjen tietokoneiden näyttöresoluutioista (StatCounter [9])



Kuva 3.2: Tilasto internetselailuun käytettyjen mobiililaitteiden näyttöresoluutioista (StatCounter [9])

nessa. Näyttöjen resoluutiot ovat keskimäärin huomattavasti pienempiä (kuva 3.2 sivulla 13) ja syöte annetaan usein kosketuksella.

### 3.1.3 Laitteen tunnistaminen

```
Mozilla/5.0 (Linux; Android 4.4.4; GTI8190 Build/KTU84P)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/36.0.1985.128
Mobile Safari/537.36
```

Koodiesimerkki 2: Samsung Galaxy S3 mini:n Android versiossa 4.4.4 suoritettavan Chrome 36:n selaintunniste

Verkkosivuja käyttäviä laitteita ja verkkoselaimia on pyritty tunnistamaan jo pitkään. Yleisesti tunnistamisen syynä on ollut se, että eri laitteille tai eri selaimille on tahdottu näyttää erilainen versio verkkosivusta. Tunnistaminen voidaan tehdä palvelinpäässä tulkitsemalla verkkoselaimen *HTTP* -kutsussa lähettämää selaintunnistetta (*user agent string*)(Koodiesimerkki 2).

Vaikkakin selaintunnisteet pääsääntöisesti identifioivat kutsun lähettäneen selaimen, eivät ne seuraa mitään standardin mukaista kaavaa [7, s. 202]. Selaintunnisteet eivät itsessään myöskään tarjoa kattavaa tietoa siitä, mitä ominaisuuksia selain tukee. Selaimen ja sen tukemien ominaisuuksien selvittämiseksi on tukeuduttava vertaamaan selaintunnistetta tunnettujen selaintunnisteiden listaan. Tällaisen listan ylläpitäminen itse olisi todella työlästä, joten listan hankkiminen kolmannelta osapuolelta on yleensä käytännöllisempi ratkaisu.

Yksi suosituimmista laitekuvauslistauksien tarjoajista on *WURFL*. *WURFL*:n avulla voidaan saada jopa 500 kohtaa kattava lista ominaisuuksista, joita selaintunnisteen mukainen selain tukee [7, s.203]. *WURFL*:n ohjelmointirajapinta osaa määrittellä myös laitteen virtuaalisia ominaisuuksia kuten käyttääkö kutsun tehnyt laite Android käyttöjärjestelmää, onko sillä iso (yli 480 pikseliä korkea ja leveä) näyttö ja onko laitteen pääasiallinen syötetapa kosketusnäyttö [24]. Laitekuvaus listauksia tarjoavia tahoja on useita, mutta ne ovat pääosin maksullisia [23; 25].

Kevyempänä ratkaisuna voidaan käyttää säännöllistä lauseketta, jonka tarkoituksena on yksinkertaisesti selvittää, onko selain mobiiliselain vai ei. Tällaisia säännöllisiä lausekkeita on saatavilla täysin ilmaiseksi [21]. Tässä tapauksessa laitteen tukemat ominaisuudet jäävät arvoitukseksi, mutta useita kohtuullisen turvallisia oletuksia voidaan kuitenkin tehdä.

### 3.1.4 Laitteen erityispiirteiden huomioiminen

Kuten luvussa 2 todettiin, web-ohjelmia voidaan käyttää lukuisilla erilaisilla laitteilla, joilla on omat erityispiirteensä. Erityisesti web-ohjelmien kannalta tärkeitä huomioitavia tekijöitä ovat laitteen:

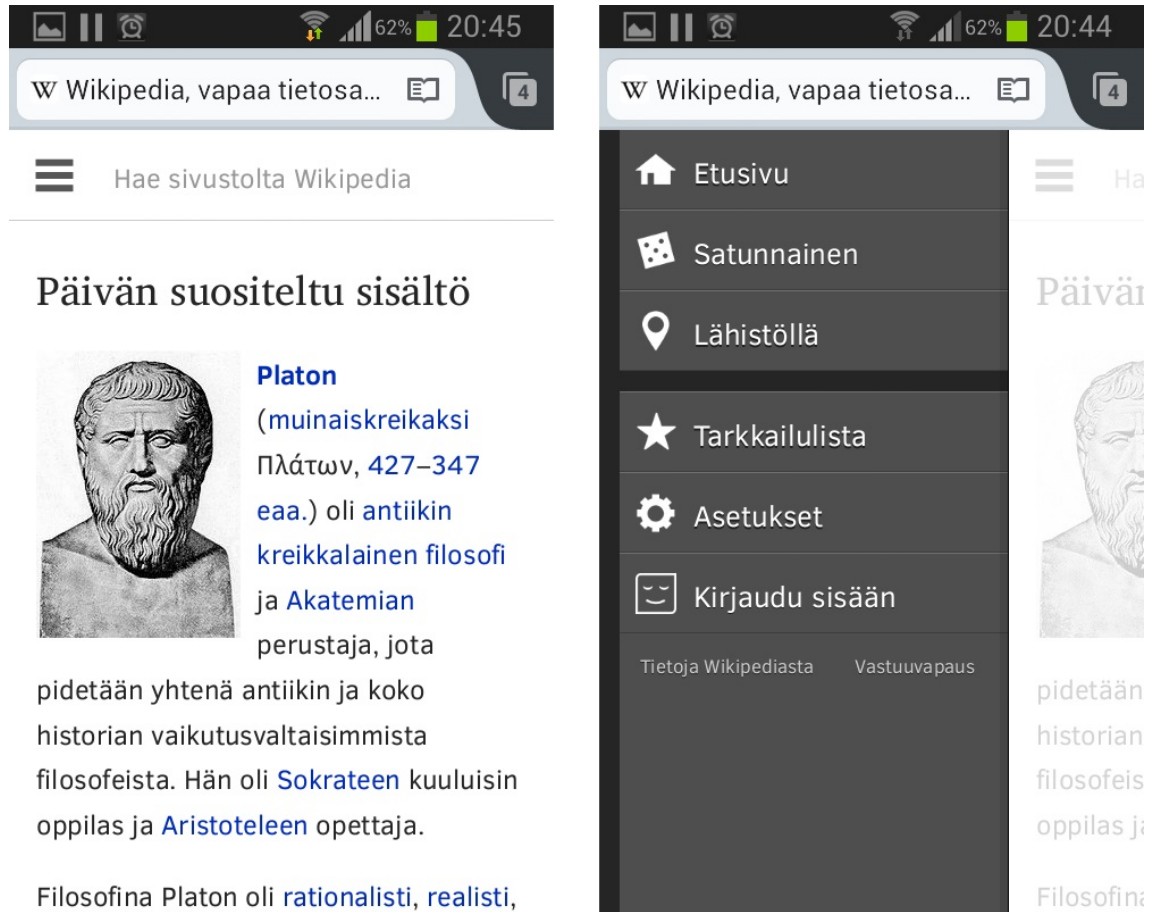
- näytön ominaisuudet (käyttöliittymän tulostaminen)
- syötetapa
- verkkoyhteys

Koska tietokone on historiallisesti ollut tähän päivään saakka oletusarvoinen verkon selailulaite, on verkkosivujen ja web-ohjelmien ominaisuudet mukautettu myös oletusarvoisesti tukemaan parhaiten tätä päätelaitetta. Verkkosivujen käyttöliittymät ja sivulla kerrallaan esitettävä sisältö ovat suunniteltu PC-tietokoneiden näyttöresoluutiota silmällä pitäen (kuva 3.3 sivulla 15). Myös syötekentät ja painikkeet ovat suunniteltu käytettäväksi näppäimistön ja hiiren avulla. Useimpien verkkoselainten määrittämän oletusfonttikoon (korkeus 16 pikseliä) mukaista tekstilinkkiä on helppo klikata hiirellä. Tietokoneita käytetään yleensä myös kohtalaisen hyvän verkkoyhteyden yli työpöydän ääressä.



Kuva 3.3: fi.wikipedia.org etusivu Google Chrome -selaimessa PC-tietokoneessa

Mobiililaitteiden ominaisuudet eroavat tietokoneiden asettamista ”oletusarvoista” kuitenkin huomattavasti. Jos verkkosivua selaava laite tunnistetaan mobiililaitteeksi, sille voidaan tarjota kokonaan eri toteutus verkkosivusta (kuva 3.4 sivulla 16). Tämä mahdollistaa laitteen erityispiirteiden huomioimisen hyvinkin kattavasti.



Kuva 3.4: fi.m.wikipedia.org etusivu Firefox for Android -selaimessa älypuhelimessa perustilassaan ja navigointivalikko avattuna

Mobiililaitteiden, erityisesti älypuhelimien näytöt ovat yleisesti huomattavasti pienempiä kuin tietokoneiden. Kuten luvussa 2.2.2 totesimme, useimmat mobiililaitteet pyrkivät ratkaisemaan tämän ongelman oletusarvoisesti emuloimalla selainikkunan leveydeksi 980 pikseliä. Tämä aiheuttaa kuitenkin useimmissa tapauksissa verkkosivun sisällön näkymisen liian pienenä käyttäjän tarpeisiin, edelleen aiheuttaen tarpeen sisällön zoomaamiseen ja horisontaaliseen vierittämiseen. Jotta pystyisimme esittämään nämä ei-toivotut asiat, on 980 pikseliä leveän ikkunan emuloinista päästävä eroon ja esitettävä verkkosivun sisältö pienemmällä pikselileveydellä, joka vastaa paremmin itse mobiililaitteen näytön kokoa.

Mobiililaitteen selainta voidaan kehottaa käyttämään manuaalisesti määritettyä pikselileveyttä tai oman ikkunansa leveyttä piirtäessään (*render*) verkkosivun sisältöä näytölle (kuva 3.5 sivulla 17). Tämä tehdään määrittelemällä sivun *HTML-*

merkkauksen *head*-osiossa *viewport*-ominaisuus *meta*-elementin avulla [7, s. 60]. Koodiesimerkki 3:ssa kehoitamme selainta käyttämään oman ikkunansa leveyttä (*content="width=device-width"*) piirtäessään verkkosivun sisältöä.

```
<html>
  <head>
    <meta name="viewport" content="width=device-width">
  </head>
  <body style="border: 5px solid; height: 300px">
    <h1>VIEWPORT META</h1>
    <p>viewport meta-elementti esitelty</p>
  </body>
</html>
```

Koodiesimerkki 3: *viewport*-ominaisuuden määrittäminen *meta*-elementillä



Kuva 3.5: Rakenteeltaan sama sisältö esitettynä selaimen oman ikkunan (*viewport*) leveydellä (koodiesimerkki 3) sekä 980 pikselin leveydelle (oletus) emuloituna Samsung Galaxy S3 Mini älypuhelimella

Kuten kuvasta 3.5 havaitsimme *Viewport*:n määrittely poistaa sisällön zoomaamisen tarpeen ja sisältö on paljon helpommin luettavissa. Selainikkunan leveys käsitellään *CSS*-pikselinä, jotka eivät vastaa välttämättä laitteen näytön pikseliä [26; 27; 28].

Useiden uusimpien älypuhelimien näytön resoluutio on huomattavasti suurempi kuin kyseisen laitteen selaimen ilmoittama resoluutio *CSS*-pikseleinä. Esimerkiksi *Apple iPhone 5c*:n näytön vaakaresoluutio on jopa kaksi kertaa suurempi, kuin mitä laitteen oletusselain ilmoittaa sen leveydeksi [29; 26]. Tämä on kuitenkin osa ratkaisua ja *CSS*-pikselien on tarkoitus tehdä sisällöstä luettavampaa suuren pikselitiheyden omaavilla laitteilla.

Toinen pienestä näytön koosta johtuva ongelma, horisontaalinen vierittäminen, on myös helposti vältettävissä *viewport*:in määrittelyn jälkeen. Ratkaisu on ns. *nestemäisen asettelun (fluid layout)* käyttö, eli prosentuaalisten arvojen käyttäminen määritettäessä verkkosivun leveyksiä *CSS*-koodissa [7, s. 25]. *Nestemäisessä asettelussa* sisällön yhteenlaskettu leveys on aina maksimissaan 100% selainikkunan leveydestä. Tällainen asettelu ei ole välttämättä esteettisesti paras vaihtoehto isokokoisilla näytöillä, mutta pienikokoisilla mobiililaitteiden näytöillä se on usein paras ratkaisu.

Asettelyn lisäksi verkkosivun sisältö on hyvä miettiä mobiililaitetta silmällä pitäen uudelleen. Koska mobiililaitteen näytöllä ei ole paljon tilaa, on tuo tila käytettävä mahdollisimman tehokkaasti hyödyksi. Verkkosivun tärkein osa, itse sisältö, tulisi olla mahdollisimman suuressa osassa ja ensimmäisenä näkyvillä (vertaa kuva 3.3 sivulla 15 ja kuva 3.4 sivulla 16). Myös keskimäärin verrattain heikot verkkoyhteydet mobiililaitteissa tuovat lisäsyyn sisällön yksinkertaistamiseen ja esimerkiksi eri kokoisten kuvien tarjoamiseen mobiilikäyttäjille [7, s. 97].

Luvussa 2.2.2 totesimme, että kosketussyötteen antaminen verkkosivulla ei oletusarvoisesti ole yhtä sujuvaa kuin hiirellä. Ilman erillistä kosketustapahtumien (*touch event*) huomioimista, selain joutuu tulkkamaan kyseiset tapahtumat erikseen hiiritapahtumiksi (*mouse event*), mikä aiheuttaa usein epätoivottua viivettä syötteen rekisteröinnissä. Tämä ongelma ratkeaa yksinkertaisesti lisäämällä tapahtumakäsittelijä (*event handler*) kosketustapahtumia varten syötepainikkeille ja linkeille.

Mobiilisivun sisältöä ja syötepainikkeita suunniteltaessa pyrkiminen niiden vähentämiseen on yleisesti hyvä lähtökohta. Vähentäminen auttaa käyttäjiä löytämään etsimänsä sisällön helpommin, suoriutumaan tehtävistään keskittyneesti ja vähentää heidän tekemien virheiden määrää. Näytöllä tulisi olla vain minimaalinen määrä sisältöä ja painikkeita, jotka vaaditaan käyttäjän tarpeiden tyydyttämiseen. Käyttäjystävällisyyden lisäksi vähentäminen helpottaa luonnollisesti myös itse verkkosivun toteuttamista. [15, s. 117-118] Kaikenkaikkiaan mobiililaitteiden erityispiirteiden huomioiminen vaatii varsin erilaista lähestymistapaa mobiilioptimoitua verkkosivua toteutettaessa. Brian Flingin sanoin [30, s. 66]:

Upeita mobiilituotteita ei koskaan tuoda muilta alustoilta, ne luodaan.

Yhteenvedona erillisten toteutusten vahvuudet ja heikkoudet on esitetty yksinkertaistettuna listana. Listausta hyödynnetään myöhemmin eri toteutustapojen arvioinnissa ja niiden keskinäisessä vertailussa.

**Toteutustavan vahvuudet:**

- Paras mahdollinen käyttökokemus ennalta rajatulla laitteella tai laiteryhmillä
- Uuden erillisen toteutuksen luominen ei vaadi muutoksia olemassa oleviin toteutuksiin
- Erilliset toteutukset ovat todennäköisesti kevyemmät kuin yksi mukautuva toteutus (palvelimelta ladataan vain tarvittu data)
- Sisältö on täysin optimoitavissa laitekohtaisesti

**Toteutustavan heikkoudet:**

- Kaikkien laitteiden kattaminen erillisillä optimoiduilla toteutuksilla on hankalaa
- Eri toteutusten erillisten koodikantojen ylläpito vaatii enemmän työvoimaa
- Laitteen tunnistaminen vaatii muutoksia palvelinpäässä
- Tehokas laitteen tunnistaminen vaatii käytännössä (maksullisen) kolmannen osapuolen laitetietokannan

### 3.2 Yksi, kaikkiin päätelaitteisiin mukautuva toteutus

Tässä luvussa esitellään ratkaisumalli, jossa verkkosivusta tehdään yksi toteutus, jota kaikki päätelaitteet käyttävät. Tätä ratkaisumallia kutsutaan mukautuvaksi verkkosuunnitteluksi (*responsive web design*). Mukautuvan verkkosuunnittelun periaatteita noudattava verkkosivu mukautuu nimensä mukaisesti käyttäjän päätelaitteeseen vaihtaen sivulla näytettävää sisältöä ja sen asettelua päätelaitteen näyttöresoluution mukaan. Mukautuminen tapahtuu kokonaan käyttäjän laitteessa, eikä siten erillistä (mobiili-)laitteen tunnistusta tarvitse suorittaa palvelinpäässä lainkaan.

Tämän luvun case-esimerkkinä toimii [www.cyber-duck.co.uk](http://www.cyber-duck.co.uk) verkkosivun siirtymä erillisistä toteutuksista mukautuvaan verkkosuunnitteluun. Case-esimerkin jälkeen käydään läpi mukautuvan verkkosuunnittelun perusteluita ja parhaita käytäntöjä. Luvussa annetaan myös koodiesimerkki, josta käy ilmi toteutustavan ydinajatus.



### 3.2.1 Case: Cyber-Duck

*Cyber-Duck* on yksi Iso-Britannian johtavista digitaalisia palveluita, kuten verkkosuunnittelua (*web design*), tarjoavista yrityksistä. Vuonna 2012 *Cyber-Duck* päätti, että heidän on aika luopua erillisestä mobiilisivustaan ja siirtyä mukautuvaan verkkosuunnitteluun pääasiallisella verkkosivullaan. Artikkelissaan *Adapting To A Responsive Design (Case Study)* [31] *Cyber-Duck*:in tuotantojohtaja *Matt Gibson* kertoo tästä muutoksesta.

*Cyber-Duck*:in aikaisempi mobiilisivu oli tehty mahdollisimman nopeasti vastaamaan mobiilikäyttäjien määrän nopeaan kasvuun. Mobiilisivu oli toteutettu erityisesti sivun latausnopeutta ja kosketuskäyttöä ajatellen. Mobiilisivun koko oli kutistettu työpöytäversion 2.2 megatavusta 700 kilotavuun ja sivun lataamiseen tarvittujen *HTTP*-pyyntöjen määrä oli miltei puolitetty 84:stä 46:een. Kosketusinteraktion parantamiseksi käyttöön oli otettu *jQuery Mobile* -mobiilikehys.

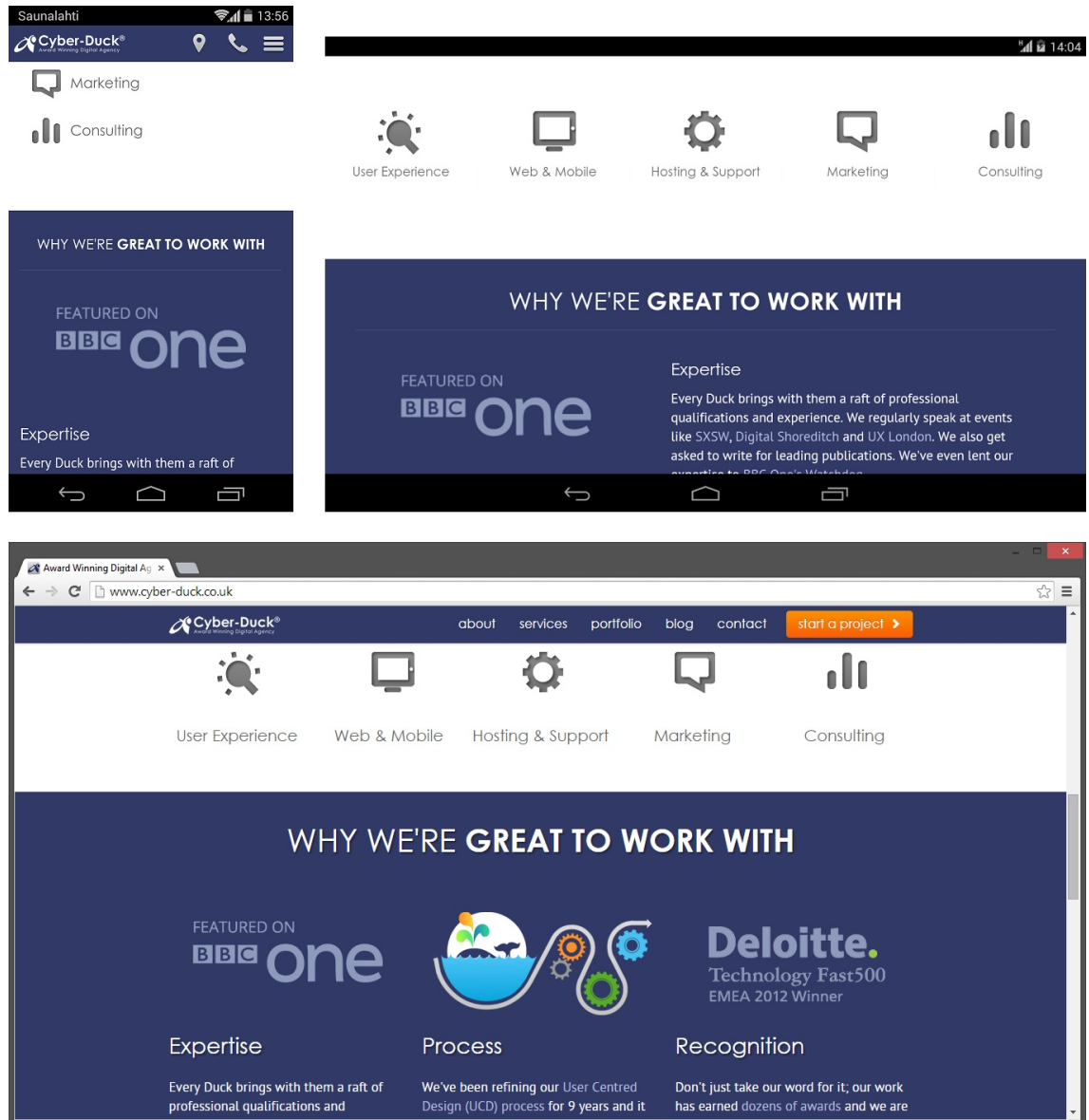
Ennen pitkää erillinen mobiilisivu osoittautui kuitenkin epäoptimaaliseksi vaihtoehdoksi useiden tekijöiden aiheuttaessa ongelmia. Usean eri toteutuksen lähdekoodin ylläpito osoittautui paljon henkilötyötunteja kuluttavaksi ja sisällönhallinta oli hankalaa, kun eri versioille piti luoda sisältö erikseen. Lisäksi yrityksen verkkosivut eivät toimineet optimaalisesti markkinoille tulleilla uuden kokoisilla laitteilla.

Viimekädessä markkinoille tulevien laitteiden eroavaisuudet oli tekijä, joka vaikutti eniten päätökseen siirtyä pois erillisistä toteutuksista mobiilille ja työpöydälle. Tavoitteeksi asetettiin luoda mukautuva verkkosivu, joka tarjoaa parhaan mahdollisen käyttökokemuksen laitteesta huolimatta. Uudelle verkkosivulle asetettiin seuraavat vaatimukset:

- Latausaikojen on oltava lyhyitä
- Sivun on toimittava ilman *CSS* muotoiluja ja *JavaScript* koodin suorittamista
- Sama sisältö ja toiminnallisuus tulisi olla saatavilla kaikilla alustoilla
- Sivun tulee toimia kaikilla alustoilla
- Sivun tulee olla ”tulevaisuusystävällinen” ylläpidon minimoimiseksi

Uuden sivun pohjana käytettiin jo olemassa olevaa mobiilisivua. Sivu rakennettiin luotettujen asiakkaiden antamaa palautetta ja *Google Analytics* analytiikkasovelluksesta saatuja tietoja hyödyntäen.

Yksi uuden sivun tärkeimmistä prioriteeteista oli tehdä siitä suorituskykyinen. Sivulatauksen on vaadittava maksimissaan 40 *HTTP*-pyyntöä ja ladattavaa dataa ei saa



Kuva 3.6: *Cyber-Duck*:in verkkosivun uusi toteutus (<http://www.cyber-duck.co.uk>) älypuhelimien, tabletin ja tietokoneen verkkoselaimessa

olla enempiä kuin 500 KT. Tämän tavoitteen saavuttamiseksi sivustolta poistettiin tarpeettomia osia, kuten kolmannen osapuolen skriptejä. Myös sivulla käytössä ollut sisällönhallintajärjestelmä päätettiin ottaa pois käytöstä, koska se koettiin tarpeettomaksi sivun harvan päivitystiheyden ja sivua päivittävien henkilöiden hyvän teknisen osaamisen johdosta. Kolmannen osapuolen *CSS*-kehymiä päätettiin myös välttää niiden tarpeettoman monimutkaisuuden takia. Itse alusta asti toteutettu *CSS* tulisi yksinkertaisuutensa lisäksi olemaan myös joustavampi sivun tarpeisiin.

Sivun käyttämä *JavaScript*-koodi ladattaisiin mahdollisesti vain jos asiakkaan selain tukee *JavaScript*:ä. Tässä tehtävässä apuna käytettiin *RequireJS JavaScript*-kirjastoa. Sama kirjasto mahdollisti myös *JavaScript* koodin tiivistämisen ja mo-

duulien lataamisen vain niitä tarvittaessa, mikä vähentää palvelimelle lähetettävien pyyntöjen ja sieltä ladattavan datan määrää.

Kaikki sivuston muotoilu tehtiin käyttäen yksinomaan *CSS*-muotoiluja tyylien suoran *JavaScript*-manipulaation sijaan. *JavaScript*:iä käytettiin ainoastaan lisäämään ja poistamaan *HTML* elementtien luokkia (*class*). *CSS* tyylien mukautuvat *murtumispisteet* (*breakpoint*) määriteltiin yksinomaan sivun sisällön, eikä oletettujen laitteiden näytön leveyksien mukaan. Lopputulos testattiin kuitenkin kattavasti erilaisilla laitteilla aina vanhemmasta *Nokia*:n puhelimesta *Playstation 3*:een.

Sivulle tehtyjen uudistuksien jälkeen *Google Analytics* tilastot näyttivät aiempaa huomattavasti paremmilta erityisesti sivun mobiilikävijöiden osalta. Mobiililaitteisiin kohdistuva liikenne nousi yli 200% ja mobiilikäyttäjien poistumistiheys aloitusivulta pieneni huikeat 4000%. Vaikka tilastot eivät kerro kaikkea, uuden mukautuvan toteutuksen (kuva 3.6) voitiin päätellä olevan parempi kuin aiemmin käytössä ollut erillinen mobiilisivu.

### 3.2.2 Mukautuva verkkosuunnittelu

Vaikka erilliset laitekohtaisesti optimoidut toteutukset voivat olla paras mahdollinen vaihtoehto niillä laitteilla joille optimoinnit on suunnattu, on ratkaisussa myös omat ongelmansa. Kuten myös *Cyber-Duck*:in mobiilisivusta luopumisen perusteista selvisi, kenties merkittävimmät ongelmat erillisissä toteutuksissa on niiden eroavien koodikantojen ylläpito sekä markkinoille tulevien laitteiden eroavaisuudet keskenään.

Laitteita on yhä vaikeampi luokitella mihinkään eristettyyn kategoriaan. Älypuhelimien näyttö voi olla mitä tahansa kolmen ja kuuden tuuman väliltä tai ympäristöstä. Vaikka älypuhelimien syötetapa useimmissa tapauksissa onkin kosketus, ei näppäimistöllä ja jonkinlaisella hiirellä (esimerkiksi kosketuslevy) varustettu malli ole missään nimessä tavaton ilmestys. Vastaavasti tablet-, kannettavan- tai työpöytätietokoneiden eroavaisuudet alkavat hämärtyä. Esimerkiksi *ASUS Transformer Pad* -perheen tuotteet ovat käytännössä perinteisiä, noin 10” kosketusnäytöllä varustettuja tablet-tietokoneita, jotka kuitenkin pystyy helposti telakoimaan täysikokoisella näppäimistöllä ja kosketuslevyllä varustettuun alarunkoon, tehden niistä käytännössä kannettavia tietokoneita [32]. Toisena esimerkkinä *Lenovo*:n *HORIZON* -tuoteperheen jopa 27” kosketusnäytöllä varustetut tietokoneet, joiden runko vastaa muotoilultaan perinteisiä tabletteja mutta joiden laitteisto ja mukana tulevat lisävarusteet ovat perinteisille tietokoneille tyypillisiä [33].

Koska laitteita on yhä vaikeampi luokitella mihinkään loogiseen kategoriaan, on

myös verkkosivun erillisten laitekohtaisten toteutusten määrittelyminen yhä hankalampaa. Erillisiä optimoituja toteutuksia tehtäessä on myös miltei mahdoton välttää tilannetta, jossa jokin laite tai laiteryhmä on huonosti tuettu. Lisäksi useamman erillisen toteutuksen ylläpitäminen tulisi yhä kalliimmaksi ja aikaakuluttavammaksi. [7, s. 9-11]

Nämä erillisten toteutusten ongelmakohdat voidaan pääosin välttää tekemällä verkkosivusta yksi, mukautuva toteutus (kuva 3.6 sivulla 21). Suunnittelumallin tarkoituksena on varautua erilaisiin päätelaitteen näyttötarkkuuksiin (*resoluutio*). Käytännössä tämä toteutetaan määrittelemällä verkkosivun *CSS*-koodissa *mediakyselyitä* (*media query*) (kuva 3.7, koodiesimerkki 4). Mediakyselyn avulla voidaan määrittää, millaisia *CSS*-tyylisääntöjä tulisi ottaa käyttöön, kun verkkosivua tarkastellaan tietynlaisella medialla. Toimiakseen tarkoituksenmukaisesti myös mobiililaitteissa, näytön resoluutioon liittyvät mediakyselyt tarvitsevat tuekseen koodiesimerkissä 3 (sivu 17) esitellyn *meta*-elementin, jossa selaimen *viewport* määritetään käyttämään leveytenään laitteen leveyttä.

Mediakyselyt ovatkin erityisen hyödyllisiä tarkasteltaessa verkkoselaimen ikkunan resoluutiota. Verkkosivun käyttöliittymäkomponenttien muotoilua, asetelua ja näkyvyyttä voidaan muuttaa, kun näytön pysty- tai vaakaresoluutio ylittää tietyn rajan. Näitä asetettuja rajoja kutsutaan murtumispisteiksi (*breakpoint*). Usein murtumispisteet asetetaan suosituimmiksi oletettujen laitteiden näyttöresoluutioiden mukaisesti kohtiin [12, s. 83, 113-114]. Näin tehdään oletusarvoisesti myös *Twitter*:in *Bootstrap*:issa, joka on yksi suosituimmista mukautuvan verkkosuunnittelun toteuttamista helpottavista (HTML, CSS ja JavaScript -) kehyksistä [34].

Koska mukautuvan verkkosuunnittelun perusajatus on luoda universaalisti kaikkiin näyttöihin ja resoluutioihin mukautuva yksittäinen toteutus, ei tiettyjä resoluutioita (oletettuja laitteita tai laitekategorioita) tulisi kuitenkaan asettaa erityisasemaan. Murtumispisteet onkin hyvä suunnitella ainoastaan verkkosivun sisällön asetteluun toimivuuden, eikä suosituimmiksi oletettujen resoluutioiden mukaan. Hyvä tapa löytää sisällölle sopivimmat murtumispisteet on tehdä verkkosivun asettelu ensin pienimmälle tuetulle resoluutiolle ja suurentaa selainikkunan kokoa, kunnes asettelu ei enää näytä hyvältä ja lisätä murtumispiste kyseiseen kohtaan. Prosessia jatketaan suurentaen selainikkunaa aina suurimpaan tuettuun resoluutioon saakka lisäten murtumispisteitä aina kun sivun asettelu alkaa näyttämään liian lavealta. [7, s. 75, 78-82]

Usein tilanteissa, joissa olemassa olevaa tietokoneille suunniteltua verkkosivua tahdotaan tehdä mobiiliystävällisemmäksi lähestytään murtumispisteitä isoimman tuetun resoluution suunnasta. Tässä tapauksessa vastaavasti murtumispisteiden kohdat

tulisi määrittää pienentämällä selainikkunaa, kunnes verkkosivun asettelu rikkoutuu tai käy liian ahtaaksi tai käyttöliittymään ilmestyy horisontaalisia vierityspalkkeja. Kummasta suunnasta murtumispisteitä lähestytään, eli kumpi ääripää on niin sanotusti oletusarvo, määrittää sen, miltä verkkosivu näyttää selaimissa jotka eivät tue *viewport* -meta-elementtiä tai mediakyselyitä. Mediakyselyitä voidaan emuloida myös erilaisilla JavaScript-kirjastoilla, joiden avulla mukautuva verkkosuunnittelu voidaan tuoda (jossain määrin) myös vanhempiin verkkoselaimiin [35; 36]. Mikäli mahdollista, kannattaa pienemmälle resoluutiolle (mobiililaitteille) suunniteltu asettelu asettaa oletusarvoksi. Mikäli verkkosivu avataan isolla näyttöresoluutiolla selaimella, joka ei tue mediakyselyitä, pystytään sivustoa todennäköisesti kuitenkin käyttämään vaikka verkkosivun käyttöliittymäelementit ovatkin isoja tai laeasti aseteltuja. Jos kuitenkin oletusarvoisesti isolle resoluutiolle suunniteltu verkkosivu avataan pieniresoluutioisella näytöllä, käy verkkosivun sisältö helposti pienikokoiseksi ja ahtaasti asetelluksi. Erityisesti liian pienet syötepainikkeet voivat osoittautua vaikeakäyttöisiksi pienellä näytöllä ja kosketussyötteellä. Toisaalta isommat syötepainikkeet perinteisillä tietokoneen syötelaitteilla käytettynä eivät tuota (ainakaan yhtä vakavia) ongelmia käytettävyyteen.

Yhteenvedona yhden, mukautuvan toteutuksen vahvuudet ja heikkoudet on esitetty yksinkertaistettuna listana. Listausta hyödynnetään myöhemmin eri toteutustapojen arvioinnissa ja niiden keskinäisessä vertailussa.

#### **Toteutustavan vahvuudet:**

- Laajan näyttöresoluutio- ja laiteskaalan tukeminen ilman erillisiä toteutuksia
- Markkinoille ilmestyvät uudet laitteet ja näyttöresoluutiot ovat todennäköisesti valmiiksi tuettuja
- Yksi koodikanta
- Laitteen ominaisuuksien tunnistaminen asiakaspäässä vähentää palvelimen kuormaa
- Sama sisältö on saatavilla kaikilla laitteilla

#### **Toteutustavan heikkoudet:**

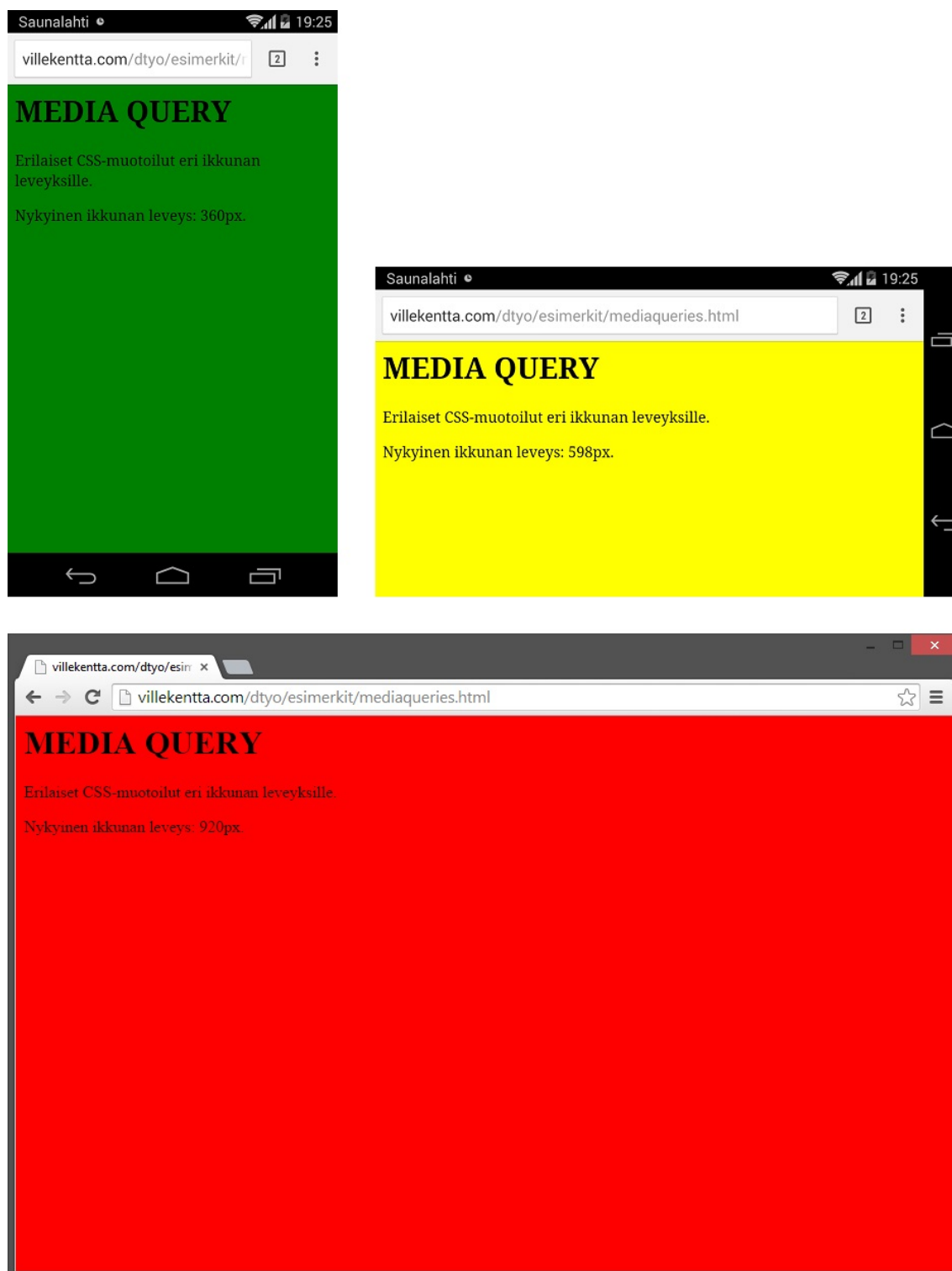
- Sama sisältö ladataan kaikilla laitteilla lisäten potentiaalisesti turhaa verkkoliikennettä
- Olemassa olevan (työpöytä-tietokone-) toteutuksen muokkaaminen mukautuvaksi saattaa tuottaa hyvin epäoptimaalisen lopputuloksen

- Laitekohtaisessa käyttökokemuksessa joudutaan tekemään pieniä kompromisseja

Kuten molempien toteutustapojen vahvuuksien ja heikkouksien listoista voidaan päätellä, ei kumpikaan toteutustapa ole automaattisesti parempi vaihtoehto. Täten toteutustavan valitseminen tapauskohtaisesti on tärkeää.

```
<html>
  <head>
    <meta name="viewport" content="width=device-width">
    <style>
      /*
        Sivun tausta asetetaan eri väriseksi
        erilaisilla selainikkunan leveyksillä.
        Oletus: vihreä.
        400 pikseliä tai leveämpi: keltainen
        900 pikseliä tai leveämpi: punainen
      */
      body {
        background: green;
      }
      @media screen and (min-width: 400px) {
        body{
          background: yellow;
        }
      }
      @media screen and (min-width: 900px) {
        body{
          background: red;
        }
      }
    </style>
    <script>
      /*
        Tämä funktio tunnistaa selainikkunan leveyden
        ja ilmoittaa sen sivulla
        (ei vaikuta tyylimuotoilujen toimintaan)
      */
      function ilmoitaLeveys()
      {
        var leveys = window.innerWidth;
        document.getElementById("leveys").innerHTML = leveys;
      }
    </script>
  </head>
  <body onload="ilmoitaLeveys()" onresize="ilmoitaLeveys()">
    <h1>MEDIA QUERY</h1>
    <p>Erilaiset CSS-muotoilut eri ikkunan leveyksille.</p>
    <p>Nykyinen ikkunan leveys: <span id="leveys"></span>px.</p>
  </body>
</html>
```

Koodiesimerkki 4: Erilaisten taustavärien asettaminen eri levyisille näytöille *CSS*-sääntöjen ja *mediakyselyiden* avulla



Kuva 3.7: Sama *HTML*-dokumentti (koodiesimerkki 4) Motorola Moto G älypuhelimella pysty- ja vaaka-asennossa sekä tietokoneella



## 4. AINEISTO JA MENETELMÄ

### 4.1 Aineisto: Surveypal-kyselytyökalu

Tämän diplomityön tapausesimerkkinä käytetään Surveypal-kyselytyökalua. Surveypal on kasvavassa suosiossa oleva kyselyiden tuottamiseen ja palautteen analysointiin keskittyvä kaupallinen web-sovellus. Kuten useimmat muut kaupalliset web-sovellukset, Surveypal-työkalu tarjotaan palveluna (*Software as a Service, SaaS*). Suurin osa Surveypal:in asiakkaista on suomalaisia yrityksiä tai järjestöjä, mutta sovelluksen voi ottaa käyttöön kuka tahansa. Surveypal perustettiin vuonna 2007 ja tuotiin markkinoille vuonna 2011.

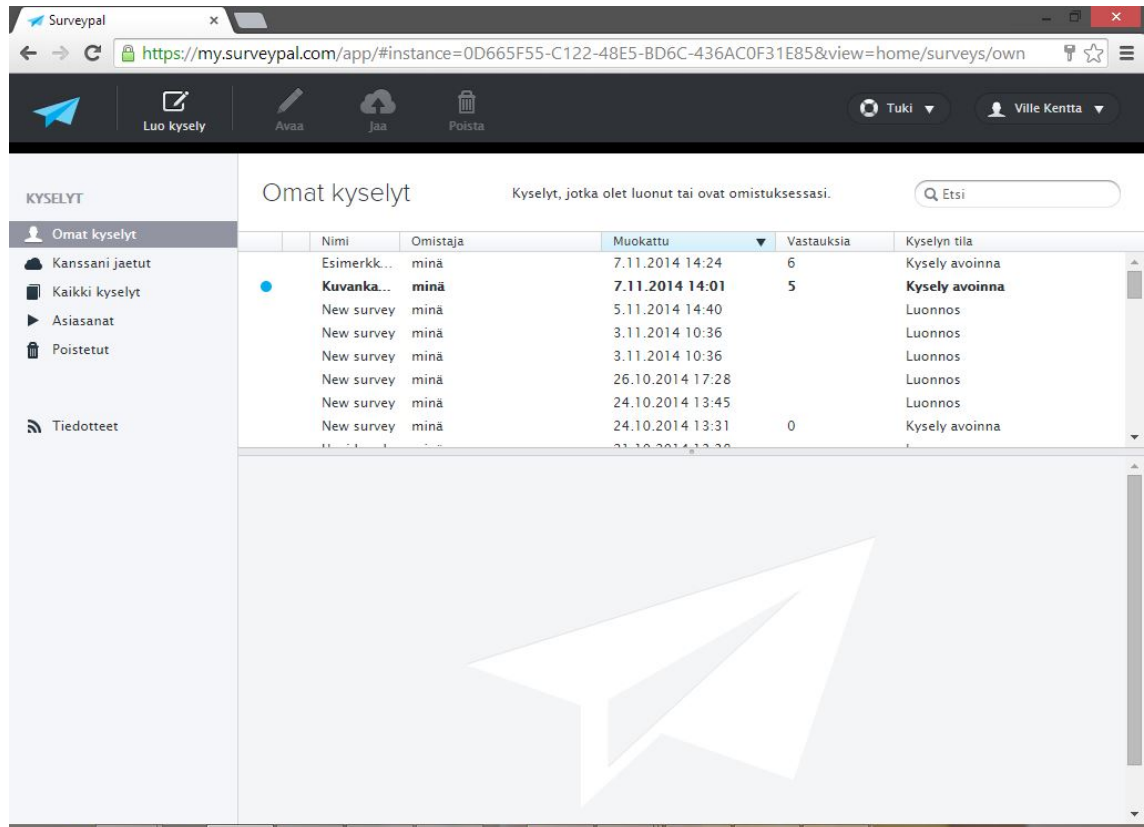
Surveypal-kyselytyökalulla voidaan luoda monipuolisia ja näyttäviä kyselyitä. Kyselyjen luominen on helppoa, mutta niihin voidaan liittää edistyneempiä ominaisuuksia kuten tietystä vastauksesta laukaistavia tapahtumia, eri kieliversioita tai täysin kustomoitu ulkoasu.

Tässä luvussa luodaan yleiskatsaus työkalun toiminnallisuuteen esittelemällä sen tärkeimmät näkymät. Lisäksi luvussa tarkastellaan työkalua myös siinä käytettyjen teknologioiden pohjalta.

#### 4.1.1 Yleiskuvaus

Käyttäkseen Surveypal-kyselytyökalua käyttäjä tarvitsee internet-yhteyden, verkkoselaimen sekä Surveypal-käyttäjätunnuksen. Käyttäjä aloittaa sovelluksen käytön navigoimalla palvelun URL-osoitteeseen (<https://my.surveypal.com>) verkkoselaimellaan. Ensimmäisenä käyttäjän täytyy kirjautua sisään sovellukseen kirjoittamalla käyttäjätunnuksensa ja salasanasensa kirjautumisnäkylässä (kuva A.1 sivulla 59).

Kirjautumisen jälkeen käyttäjä ohjataan aloitusnäkympään, jossa käyttäjä näkee kaikki kyselynsä (kuva 4.1). Tästä näkympään käyttäjä voi avata, kopioida, uudelleennimetä, jakaa ja poistaa olemassa olevia kyselyitään sekä aloittaa uuden kyselyn luomisen. Lisäksi käyttäjä voi avata tukipalveluita, kuten katsella ohjeita ja avata



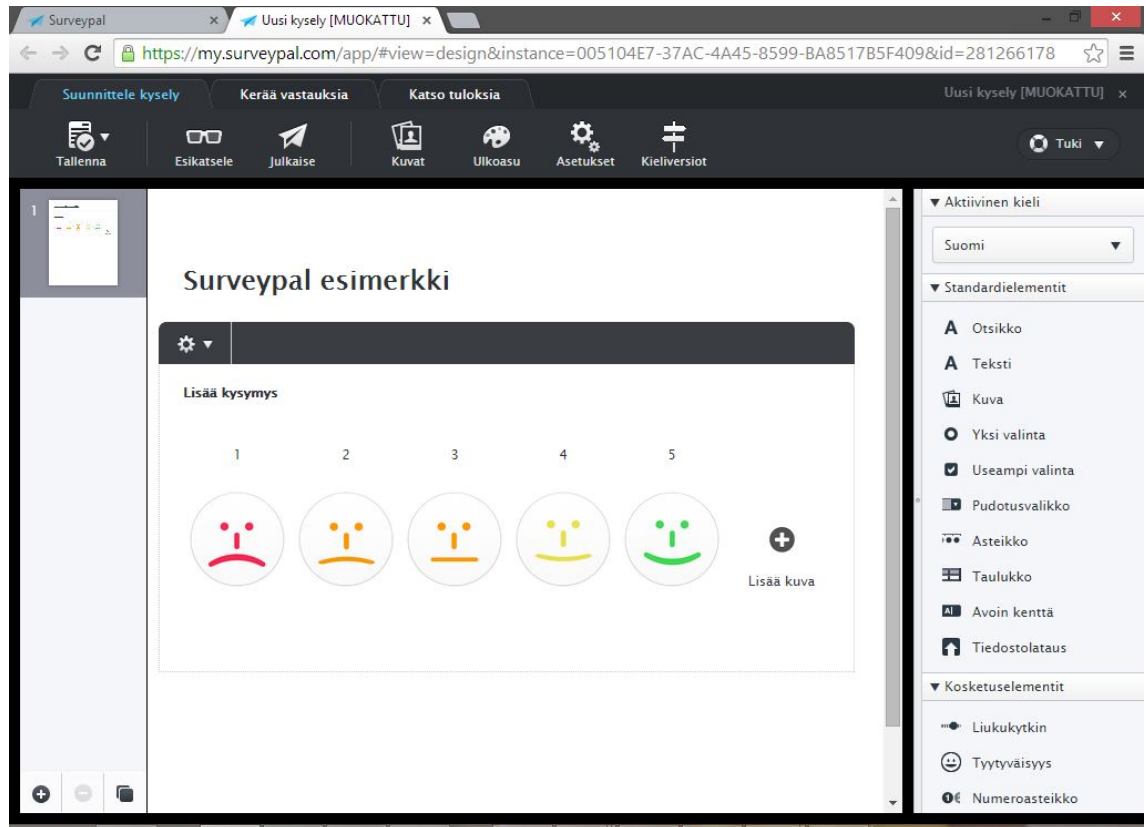
Kuva 4.1: Surveypal - aloitusnäky

tuki-chatin. Käyttäjä voi myös siirtyä tarkastelemaan omia tietojaan tai kirjautua ulos.

Uuden kyselyn luominen tapahtuu erillisessä näkymässä (kyselyeditori), joka avataan uuteen selainikkunaan (kuva 4.2). Myös olemassa olevien kyselyiden rakenteen muokkaaminen tapahtuu samassa näkymässä. Kyselyt muodostuvat kysymyselementeistä, joita voi raahata hiirellä näkymän oikeassa reunassa sijaitsevasta paneelista keskellä olevaan työtilaan. Itse kysymyselementtejä voidaan muokata kyselynäkymässä niiden lisäämisen jälkeen. Kysymyselementtejä voidaan muun muassa poistaa ja kopioida kysymyselementin vasemmasta yläkulmasta aukeavasta valikosta. Tässä näkymästä käyttäjä voi tallentaa ja julkaista kyselyn. Lisäksi käyttäjä voi siirtyä esikatsellemaan kyselyä tai muokkaamaan sen ulkoasua.

Kyselyn esikatselu tapahtuu uuteen ikkunaan avautuvassa näkymässä (kuva A.4). Näkymä vastaa yksi yhteen kyselyn vastausnäkyä ja on kokonaan irrallaan kyselyn muokkausnäkyästä.

Kyselyn ulkoasua muokataan myös erillisessä näkymässään (ulkoasueditori, kuva A.5). Ulkoasueditorissa käyttäjä voi asettaa kyselylle ennalta määritellyn teeman tai luoda kokonaan uuden teeman. Ulkoasua pystytään muokkaamaan hyvinkin yk-

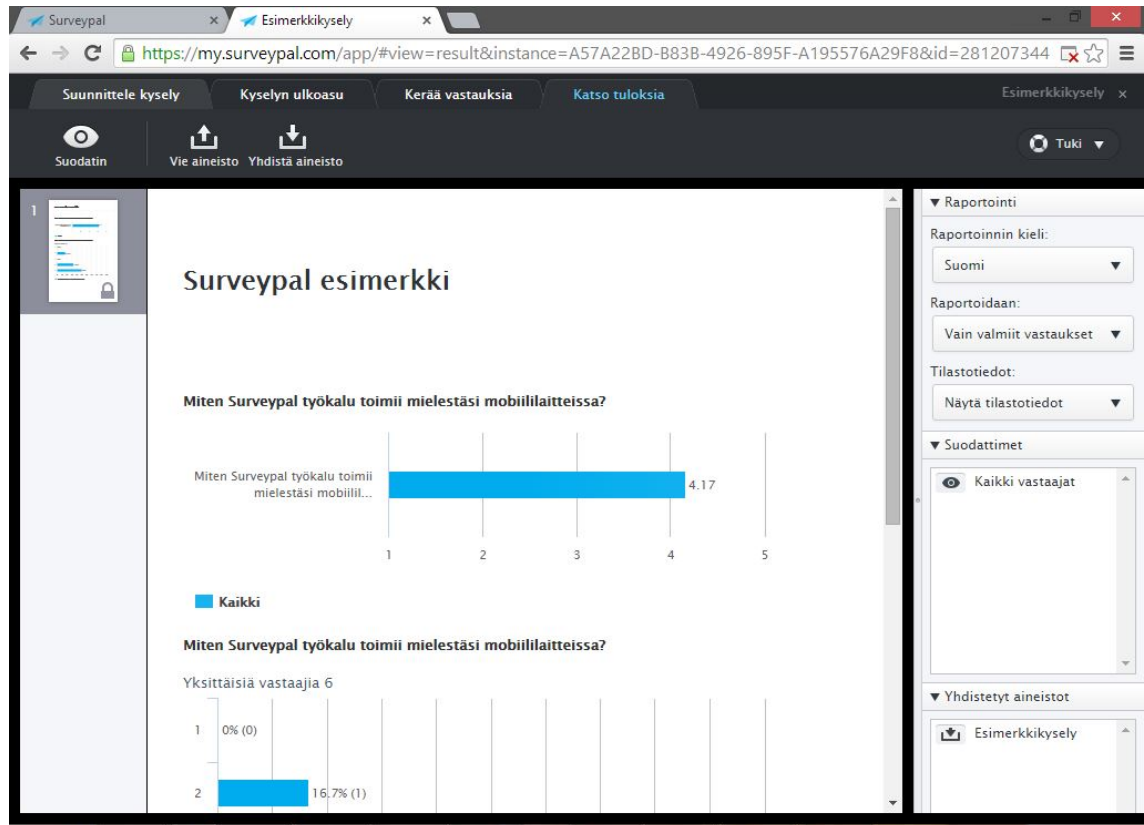


Kuva 4.2: Surveypal - kyselyn luominen ja muokkaus

sityiskohtaisesti ja esimerkiksi eri näyttöresoluutioille (oletetuille tablet-tietokoneille ja älypuhelimille) ja niiden orientaatioille pystytään asettamaan täysin erilaiset tyyliasetukset (kuva A.6).

Kysely voidaan julkaista joko kyselyeditorissa, ”kerää vastauksia” -näkyssä tai ”katso tuloksia” -näkyssä. Kyselyn julkaisemisen jälkeen käyttäjä voi kerätä vastauksia esimerkiksi lähettämällä yksityisiä sähköpostikutsuja tai jakamalla julkisen vastauslinkin (kuva A.7). ”Kerää vastauksia” -välilehdellä näkyy kaikki vastaajat, jotka ovat avanneet kyselyn tai joille on lähetetty yksityinen vastauslinkki (kuva A.8). Tällä välilehdellä käyttäjä voi muun muassa tarkastella ja poistaa yksittäisiä vastauksia, kerätä lisää vastauksia ja tarkastella kutsuhistoriaa.

Kun kyselyyn on saatu vastauksia, näkyy niiden lukumäärä aloitusnäkyssä (kuva A.9). Kyselyn vastauksia voidaan tarkastella kootusti ”katsele tuloksia” -näkyssä (kuva 4.3), johon päästään yksinkertaisesti avaamalla kysely, jossa on vastauksia. Vastaukset esitetään useimpien kysymystyyppien kohdalla palkkeina, jotka kuvaavat vastausten keskiarvoa tietyllä asteikolla tai tehtyjen valintojen prosenttiosuuksia kaikista valinnoista. Avoimet testivastaukset esitetään yksinkertaisesti listana tekstuaalisia vastauksia. Vastauksia voidaan myös suodattaa, jos tahdotaan nähdä vain tietynlaisia vastauksia antaneiden vastaajien vastaukset.



Kuva 4.3: SurveyPal - kyselyn tulokset

### 4.1.2 Teknologiat

SurveyPal on perinteinen web-sovellus, eli sitä käytetään verkkoselaimessa toimivan käyttöliittymän kautta. Sovelluksen *front end* muodostuu siis *HTML*-, *CSS*- ja *JavaScript*-kielillä toteutetuista tiedostoista, jotka tulkitaan ja suoritetaan selaimessa. Sovelluksen *back end* on toteutettu *Java*:lla.

JavaScript on erittäin merkittävässä osassa SurveyPal:in *front end*:ää. Sovellus rakennettiin alunperin *MooTools*[41] *JavaScript*-kirjaston avulla. Nykyisellään *MooTools* on kuitenkin korvattu lähes kokonaan *jQuery*:llä[39]. Sovelluksen JavaScript moduulit (koodiesimerkki 5 sivulla 32) on pääosin toteutettu *John Resig*:nin kehittämän periyttävän luokan pohjalta [38]. Nämä moduulit on pakattu *RequireJS*[40]-pakettien sisään, joka mahdollistaa niiden lataamisen asynkronisesti ja vain niitä tarvittaessa.

SurveyPal on suunniteltu alunperin yhden verkkosivun sovellukseksi (*Single Page Application, SPA*) (näkymien välinen siirtymä ei aiheuta sivun uudelleenlatausta) ja on sellainen pohjimmiltaan vielä nykyäänkin. Vaikka SurveyPal onkin yhden verkkosivun sovellus, ei kaikkia sovelluksen tarvitsemia resursseja kuitenkaan ladata ensimmäisellä sivun latauksella. Sovelluksen suorituksen aikana ladataan dynaamises-

```
define("home/view", function(require)
{
    var CSS = require("text!home/view.css");
    var HomeTemplate = require("text!home/view.html");
    var List = require("home/list");
    var Preview = require("home/preview");
    var View = require("ui/view");

    Surveypal.appendCSS(CSS);

    return View.extend(
    {
        init: function()
        {
            this.supreme(HomeTemplate);

            this._list = new List();
            this._preview = new Preview();

            Surveypal.swap(this);
        },

        getList: function()
        {
            return this._list;
        }

        getPreview: function()
        {
            return this._preview;
        }

        fit: function()
        {
            this.supreme();

            this.getList().fit();
            this.getPreview().fit();
        }
    });
});
```

Koodiesimerkki 5: RequireJS-moduuli (Surveypal aloitusnäkömännäkömänn yksinkartaistet-  
tu versio)

ti niin *HTML*-, *CSS*- kuin *JavaScript*-tiedostoja. *RequireJS* yhdessä periyttävien luokkien kanssa helpottaa koodin jakamista selkeisiin moduuleihin ja niiden lataamista selaimen vain niitä tarvittaessa. Käyttöliittymämoduuleihin liittyvät *HTML*- ja *CSS*-tiedostot ladataan myös vasta moduulin luonnin yhteydessä.

Käyttöliittymän muotoiluja hallinnoidaan pääasiassa *CSS*-tyylitiedostojen avulla. Osa elementtien tyylimäärittelyistä määritetään kuitenkin *jQuery*-kirjaston avulla suoraan *JavaScript*-koodissa manipuloimalla *HTML*-elementtien *style*-attribuutteja. *JavaScript*-koodilla määritellyt muotoilut ovat yleensä selainikkunan koon muuttuessa laukaistavia muutoksia (*fit*-funktio koodiesimerkissä 5), jotka muistuttavat toiminnaltaan hieman mukautuvassa verkkosuunnittelussa käytettäviä mediakyselyitä.

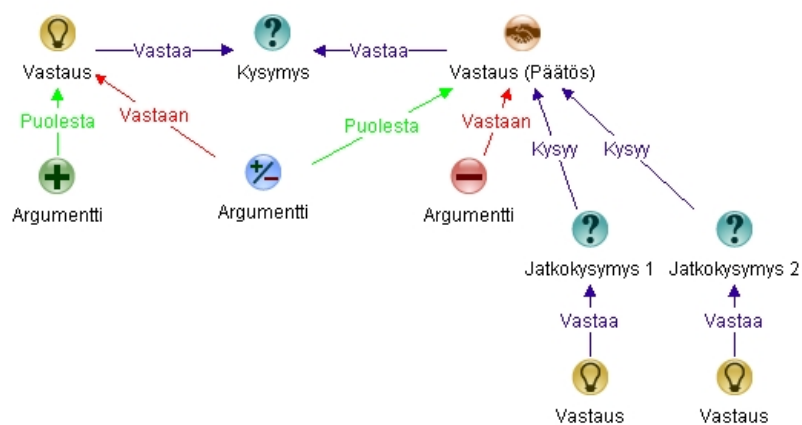
Kaikki sovellukseen suorituksen aikana ladattava koodi on tiivistetty (*minified*) *RequireJS*:n tarjoamalla optimointityökalulla. Esimerkiksi koodiesimerkki 5:ssä esitetyn moduulin (*home/view.js*) tiivistäminen tarkoittaisi, että *ui/view.js*, *home/preview.js*, *home/view.html* ja *home/view.css* -tiedostojen koodit upotetaan itse *home/view.js*:n sisään. Tämän lisäksi koodia lyhennetään muuttamalla muuttujanimiä lyhemmiksi ja poistamalla tyhjää tilaa koodista (välilyönnit ja rivinvaihdot).

## 4.2 Menetelmä

Surveypal-sovelluksen mobiilitoteutuksen suositellun toteutustavan perustelun apuna käytetään *Issue-Based Information System*:iä (*IBIS*), joka on Werner Kunzin ja Horst Rittelin kehittämä argumentointi-pohjainen metodi monimutkaisten ongelmien (*Wicked Problem*) ratkaisemiseen [46]. *IBIS*:iin perustuva ongelmakartta (*Issue Map*) [47] koostuu kysymyksistä (ongelma), niihin liittyvistä vastauksista (ratkaisu) sekä vastauksia kohtaan esitetyistä argumenteista (kuva 4.4). Vastauksiin voi liittyä myös uusia kysymyksiä. Kysymykseen liittyvä vastaus voidaan merkitä erikseen ongelman ratkaisevaksi päätökseksi. Ongelmakartan piirtämiseen käytettiin Compendium-sovellusta [48].

Ongelmakartan argumentit luodaan:

1. Eri toteutustapojen yleispiirteistä
2. Eri toteutustapojen suhtautumisesta W3C:n määrittelemiin web-sovelluksen parhaisiin käytäntöihin
3. Surveypal-käyttäjien mobiilikäyttötottumuksia käsittelevän kyselyn tuloksista



Kuva 4.4: *Issue-Based Information System (IBIS)* -metodin notaatio

## 5. ARVIOINTI

Kuten luvussa 3 todettiin, on mobiillin web-sovelluksen erilaisilla toteutustavoilla eri vahvuudet ja heikkoudet. Tässä luvussa näitä vahvuuksia ja heikkouksia punnitaan Surveypal-sovelluksen ja *World Wide Web Consortium*:in (*W3C*) asettamien parhaiden käytäntöjen näkökulmasta [43]. Lisäksi luvussa käydään läpi Surveypal-sovelluksen käyttäjille esitetyn, sovelluksen mobiilikäyttöön liittyvän kyselyn tulokset. Luvun tavoitteena on kerätä perusteluita sille, kumpi toteutustapa on parempi vaihtoehto Surveypal-sovelluksen tapauksessa.

### 5.1 Mobiilien web-sovellusten parhaat käytännöt

Tässä luvussa käydään läpi W3C:n kokoama lista parhaista käytännöistä toteutettaessa mobiilia web-sovellusta. Luvun tarkoituksena on arvioida, miten työlästä Surveypal-sovelluksen mobiilitoteutuksen luominen olisi eri toteutustavoilla. Jokaisen suosituksen kohdalla eri toteutusvaihtoehdoille annetaan 0-3 pistettä sen mukaan, miten työlästä kyseinen suositus olisi toteutustavalla, nykyisistä lähtökohdista täyttää (taulukko 5.0). Arvioinnissa 0 pistettä tarkoittaa, että suosituksen täyttäminen on erityisen työlästä ja 3 pistettä, että suositus on jo valmiiksi täytetty. Koska erillisellä mobiilitoteutuksella ei ole jo toteutettua pohjaa, voi se saada arvioinnissa parhaimmillaan 2 pistettä suositusta kohden.

Suositus	Erilliset toteutukset	Mukautuva toteutus
Esimerkkisuositus 1	1	3
Esimerkkisuositus 2	2	0
<b>Yhteensä</b>	<b>3</b>	<b>3</b>

Taulukko 5.0: Tässä luvussa esiintyvien taulukoiden rakenne

Pisteytyksen perusteina toimivat liitteessä C tehdyt arviot Surveypal-sovelluksen nykytilan, erillisten toteutusten, sekä mukautuvan toteutuksen suhtautumisesta suosituksiin. Erillisten toteutusten tapauksessa oletuksena on, että nykyisen sovelluksen rinnalle tehdään kokonaan uusi erillinen mobiililaitteille tarjottava web-sovellus.



Mukautuvan toteutuksen tapauksessa oletuksena on, että nykyistä sovellusta käytetään pohjana mukautuvan sovelluksen luomisessa. Suositukset on jaettu yhteisen teeman mukaisiin ryhmiin W3C:n alkuperäisen ryhmittelyn mukaan.

### 5.1.1 Sovellusdata

Suositus	Erilliset toteutukset	Mukautuva toteutus
1. Käytä evästeitä säästeliäästi	2	3
2. Käytä asiakaspään tietovarastoteknologioita lokaalille datalle	1	2
3. Toista lokaali data	1	3
<b>Yhteensä</b>	<b>4</b>	<b>8</b>

Taulukko 5.1: W3C-suositukset: Sovellusdata (liite C.1)

Surveypal-sovelluksen täyttäessä jo nykyisellään suositukset 1 ja 3, ei mukautuva toteutus vaatisi näiden suositusten täyttämiseksi lainkaan lisätyötä. Myös suositus 2 täyttyy jo kohtalaisen hyvin sovelluksen nykyisessä toteutuksessa, vaikka pientä parannettavaa tämän suosituksen kohdalla onkin. Molemmilla toteutustavoilla on mahdollista saavuttaa yhtä hyvä lopputulos kaikkien suositusten kohdalla. Mukautuva toteutus vie suuremman osan yhteispisteistä tämän suositusryhmän osalta pääasiassa Surveypal-sovelluksen hyvän nykytilan vuoksi (taulukko 5.1).

### 5.1.2 Turvallisuus ja yksityisyys

Suositus	Erilliset toteutukset	Mukautuva toteutus
4. Älä suorita suodattamattonta tai epäluotettua <i>JSON</i> -dataa	2	3
<b>Yhteensä</b>	<b>2</b>	<b>3</b>

Taulukko 5.2: W3C-suositukset: Turvallisuus ja yksityisyys (liite C.2)

Surveypal-sovelluksen nykytila täyttää jo ryhmän ainoan suosituksen (taulukko 5.2), mistä johtuen siihen pohjautuvassa mukautuvassa toteutuksessa suositus olisi jo valmiiksi täytetty. Molemmilla toteutustavoilla on mahdollista saavuttaa yhtä hyvä lopputulos, ja suosituksen noudattaminen on yksinkertaista.

### 5.1.3 Käyttäjän tietoisuus ja hallinta

Suositus	Erilliset toteutukset	Mukautuva toteutus
5. Varmista, että käyttäjää informoidaan henkilökohtaisen ja laitteeseen liittyvän informaation käytöstä	2	3
6. Mahdollista automaattinen sisäänkirjautuminen	1	2
<b>Yhteensä</b>	<b>3</b>	<b>5</b>

Taulukko 5.3: W3C-suositukset: Sovellusdata (liite C.3)

Kolmannessa suositusryhmässä (taulukko 5.3) kumpikaan toteutustapa ei osoittautunut toista selkeästi paremmaksi. Erillisten toteutusten kohdalla vaadittu pieni lisätyö käyttäjien informoinnin osalta olisi helposti toteutettavissa. Vaikka Surveypal-sovelluksessa ei nykyisellään tarjotakaan mahdollisuutta automaattiseen sisäänkirjautumiseen, olisi se silti helpompi toteuttaa nykyisen toteutuksen pohjalta, kuin tehdä koko sisäänkirjautumismekanismi alusta erillistä toteutusta varten.

### 5.1.4 Resurssien hillitty käyttö

Resurssien käyttöön keskittyvässä, neljännessä suositusryhmässä mukautuva toteutus sai hieman erillisiä toteutuksia paremmat yhteispisteet (taulukko 5.4). Mukautuvan toteutuksen pistevoitto oli jälleen pääasiassa sovelluksen hyvän nykytilan ansiota. Myös niiden suositusten täyttäminen, jotka eivät vielä sovelluksen nykytilassa täyty, olisi keskimäärin helpommin tai vähintään yhtä helposti toteutettavissa nykyiseen toteutukseen perustuvassa mukautuvassa toteutuksessa kuin kokonaan uudessa mobiilitoteutuksessa. Ainut poikkeus tähän sääntöön oli suositus numero 17. ”Pidä DOM:in koko kohtuullisena” johtuen mukautuvan toteutuksen luonteesta.

Resurssien käytön osalta erillisellä mobiilitoteutuksella voitaisiin saavuttaa jonkin verran parempi lopputulos mukautuvaan toteutukseen verrattuna, mikäli sovelluksesta tehtäisiin ominaisuuksiltaan karsitumpi versio. Tässä tapauksessa resurssien kokoa ja määrää saataisiin luonnollisesti pienennettyä. Suositusten täyttäminen erillisessä toteutuksessa vaatisi kuitenkin merkittävästi enemmän resursseja niin toteutus- kuin ylläpitovaiheessakin.

Suositus	Erilliset toteutukset	Mukautuva toteutus
7. Pakkaa siirrettävät resurssit	2	3
8. Minimoi sovelluksen ja datan koko	2	3
9. Vältä uudelleenohjauksia	1	3
10. Optimoi verkkopyynnöt	1	2
11. Minimoi ulkopuoliset resurssit	2	3
12. Yhdistä staattiset kuvat yhdeksi komposiittiresurssiksi ( <i>Sprite</i> )	2	2
13. Sisällytä taustakuvat CSS-tyylimäärittelyjen sisään	2	2
14. Välimuistita resurssit merkitsemällä resurssin viite	2	2
15. Välimuistita AJAX-data	1	2
16. Älä lähetä evästetietoja tarpeettomasti	2	3
17. Pidä DOM:in koko kohtuullisena	2	1
<b>Yhteensä</b>	<b>19</b>	<b>26</b>

Taulukko 5.4: W3C-suositukset: Sovellusdata (liite C.4)

### 5.1.5 Käyttökokemus

Suositus	Erilliset toteutukset	Mukautuva toteutus
18. Optimoi sovelluksen käynnistymisnopeus	1	3
19. Minimoi havaittu latenssi	2	3
20. Varaudu erilaisiin interaktiometodeihin	1	1
21. Säilytä fokus päivitettävissä sivua dynaamisesti	2	3
22. Käytä osiotunnisteita ( <i>fragment ID</i> ) ohjaamaan sovelluksen näkymiä	1	3
23. Tee puhelinnumeroista ”Klikkaa soittaaksesi” -painikkeita	2	2
24. Varmista tekstikappaleiden sujuvuus	1	1
25. Varmista, että laitteiden välinen tila on yhdenmukainen	1	2
26. Huomioi web-sovellusten käynnistämiseen soveltuvat mobiililaitteiden tukemat teknologiat	1	1
27. Käytä <i>meta viewport</i> -elementtiä toivotun näyttöresoluution tunnistamiseen	1	1
<b>Yhteensä</b>	<b>13</b>	<b>20</b>

Taulukko 5.5: W3C-suositukset: Käyttökokemus (liite C.5)

Käyttökokemusta koskevien suositusten (taulukko 5.5) täyttäminen arvioitiin olevan mobiilisiirtymän työläin osuus. Toteutustavasta riippumatta sovelluksen mobiilikäyttöliittymän suunnitteluun ja toteuttamiseen kuluu väistämättä merkittävä määrä aikaa. Suosituksen numero 27. (”käytä *meta viewport* -elementtiä toivotun näyttöresoluution tunnistamiseen”, taulukko C.27) täyttäminen ei itsessään ole työlästä, mutta siitä aiheutuva tarve toteuttaa erilaisilla näyttöresoluutioilla toimiva käyttöliittymä on. Myös kattava varautuminen erilaisiin interaktiometodeihin (taulukko C.20) on huomattavan työlästä.

Vaikka valtaosa suosituksista vaatii huomiota toteutustavasta riippumatta, sai mukautuva toteutus merkittävästi enemmän pisteitä johtuen nykyisen toteutuksen hyvästä tilasta suositusten 18, 19, 21 ja 22 osalta. Myös suosituksen 25 täyttyminen on huomattavasti luonnollisempaa mukautuvassa toteutuksessa, kuten taulukossa C.25 todettiin.

### 5.1.6 Erilaisten toimituskontekstien käsittely

Suositus	Erilliset toteutukset	Mukautuva toteutus
28. Suosi palvelinpään tunnistusta niiltä osin kuin se on mahdollista	1	3
29. Käytä asiakaspään tunnistusta kun tarpeellista	1	1
30. Käytä laitteiden luokittelua sisällön mukautumisen yksinkertaistamiseksi	1	2
31. Tue ilman JavaScript:iä toimivaa variaatiota soveltuvin osin	1	0
32. Tarjoa käyttäjälle mahdollisuus valita käyttöliittymien välillä	1	2
<b>Yhteensä</b>	<b>5</b>	<b>8</b>

Taulukko 5.6: W3C-suositukset: Erilaisten toimituskontekstien käsittely (liite C.6)

Suositusryhmässä kuusi, ”erilaisten toimituskontekstien käsittely” (taulukko 5.6), yksikään suosituksista ei varsinaisesti täyttynyt sovelluksen nykyisessä toteutuksessa. Nykyiseen toteutukseen pohjautuva mukautuva toteutus sai kuitenkin täydet kolme pistettä suosituksesta numero 28, koska sitä ei tarvitsisi huomioida kyseisessä tapauksessa lainkaan, eikä täten vaatisi lisätyötä. Mukautuvan toteutuksen luonteen takia, myös laitteiden luokittelua koskeva suositus (30) katsottiin olevan vähintäänkin helposti toteutettavissa. Ilman JavaScript:iä toimivan variaation toteuttamista nykyisen toteutuksen pohjalta ajateltiin olevan epärealistinen vaihtoehto, koska se tukeutuu JavaScriptiin erittäin voimakkaasti. Eri käyttöliittymien välillä siirtymisen olisi mukautuvan toteutuksen kohdalla epätodennäköisemmin tarpeellista asiakaspäässä tehtävien ominaisuustunnistuksien johdosta. Tarvittaessa eri käyttöliittymän valitseminen on kuitenkin myös helpommin toteutettavissa mukautuvassa toteutuksessa.

Kaikki suositusryhmän suositukset olisi täytettävissä erillisessä mobiilitoteutuksessa. Ne vaatisivat kuitenkin pääosin paljon toteutustyötä. Suosituksista 28, 29 ja 30 voidaan päätellä W3C:n suosittavan erillisen toteutuksen luomista mobiilikäyttäjille.

### 5.1.7 Muuta huomioitavaa

Suositus	Erilliset toteutukset	Mukautuva toteutus
33. Harkitse canvas-elementin tai <i>SVG</i> :n käyttöä dynaamiselle grafiikalle	2	3
34. Informoi käyttäjää automaattisesta verkon käytöstä	1	1
35. Tarjoa riittävät mahdollisuudet hallita automaattista verkon käyttöä	1	1
<b>Yhteensä</b>	<b>4</b>	<b>5</b>

Taulukko 5.7: W3C-suositukset: Muuta huomioitavaa (liite C.7)

Viimeisen suositusryhmän, ”muuta huomioitavaa” (taulukko 5.7), kohdalla Surveypal-sovelluksen nykyinen toteutus täytti dynaamista grafiikkaa käsittelevän suosituksen (33). Muiden suositusten ollessa yhtä työläitä täytettäviä toteutustavasta riippumatta, päättyi mukautuva toteutus saamaan hieman paremmat pisteet myös tässä suositusryhmässä.

### 5.1.8 Yhteenveto

Taulukoista C.28, C.29 ja C.30 voidaan havaita, että W3C suosittelee tekemään sovelluksesta erillisen version mobiililaitteille. Kuten luvussa 3.2.2 todettiin, on erillisissä toteutuksissa kuitenkin omat ongelmakohtansa. W3C:n parhaat käytännöt ovat kuitenkin pääosin täysin päteviä myös mukautuvaa (tai jopa pelkästään tietokoneelle suunnattua) toteutusta suunniteltaessa. Syy siihen, miksi mukautuvaa verkkosuunnittelua ei olla huomioitu W3C:n suosituksissa saattaa olla se, ettei kyseinen toteutusmetodi ollut vielä suositusten laatimishetkellä (14.12.2010) laajalti tiedossa, eikä suosituksia olla sen jälkeen päivitetty. Mukautuva verkkosuunnittelu (*Responsive Web Design*) mainittiin ensimmäisen kerran 25.5.2010 [44] ja W3C:n suosituksen työversio esiteltiin ensimmäisen kerran jo 29.7.2008 [45], jonka jälkeen

Suositusryhmä	Erilliset toteutukset	Mukautuva toteutus	Max
Sovellusdata	4	8	9
Turvallisuus ja yksityisyys	2	3	3
Käyttäjän tietoisuus ja hallinta	3	5	6
Resurssien hillitty käyttö	19	26	33
Käyttökokemus	13	20	30
Erilaisten toimituskontekstien käsittely	5	8	15
Muuta huomioitavaa	4	5	9
<b>Yhteensä</b>	<b>50</b>	<b>67</b>	<b>105</b>

Taulukko 5.8: Suositusryhmien yhteispisteet

se ei juurikaan muuttunut palvelinpään tunnistusta suosivan parhaan käytännön osalta.

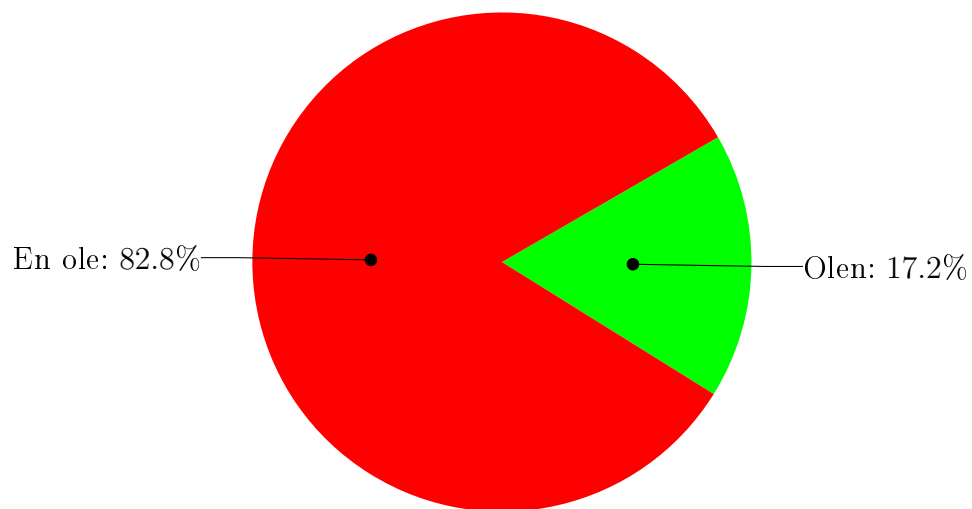
Suuri osa W3C:n esittämistä suosituksista toteutui kokonaan tai osittain jo Survey-pal-sovelluksen nykyisessä toteutuksessa. Suositukset, jotka eivät vielä nykyisessä toteutuksessa täyty, olisi pääosin toteutettavissa järkevinä ylläpito- ja kehitysprojekteina. Osa suosituksista (erityisesti taulukot C.17, C.20, C.24 ja C.31) pystyttäisiin toteuttamaan helpommin erillisessä, kokonaan alusta asti mobiiliin suunnitellussa toteutuksessa. Kaikki muut paitsi taulukko C.31 on käytännössä mahdollista ja järkevää toteuttaa myös nykyisen toteutuksen pohjalta.

Nykytilan ollessa pääosin hyvä lähtökohta mukautuvalle toteutukselle, erillisen mobiilitoteutuksen tekeminen vaikuttaa huonommalta vaihtoehdolta siitä aiheutuvan huomattavasti suuremman työmäärän vuoksi (taulukko 5.8). Vaikka jo arvio pelkkien suositusten täyttämisen työläydestä kertoo osittain siitä, miten paljon lisätyötä erillinen, kokonaan uusi toteutus tuo mukanaan, ei sen yleisen toteuttamisen ja ylläpidon tuomaa lisätyötä (muuten kuin W3C:n suositusten täyttämisen osalta) tule jättää huomiotta.

Mikäli mobiilikäyttäjille tahdottaisiin tarjota eri ominaisuuksilla varustettu (kevyempi) versio sovelluksesta, olisi erillinen mobiilitoteutus luonnollisempi vaihtoehto. Täten vältyttäisiin lähettämästä mobiililaitteille (sekä muille laitteille) merkittävää määrää yhteensopimatonta dataa ja sovelluskoodia.

## 5.2 Surveypal-käyttäjien kanta sovelluksen mobiilikäyttöön

Surveypal-sovelluksen käyttäjille lähetettiin sähköpostitse kysely, jossa tiedusteltiin heidän tottumuksiaan ja näkemyksiään sovelluksen mobiilikäytön suhteen (Liite D). Kysely luotiin ja lähetettiin vastaajille Surveypal-sovelluksen avulla. Surveypal-sovellus toimi apuna myös tuloksien analysoinnissa ja vastauksien luokittelussa (kuva 5.2). Kyselyyn vastasi 458 käyttäjää kaikista 1900 hiljattain aktiivisesta Surveypal-käyttäjistä (Liite E). Vastausprosentiksi muodostui 24%, joten kyselystä saatuja tuloksia voidaan pitää kohtalaisen luotettavana kuvana käyttäjien yleisestä mielipiteestä.



Kuva 5.1: Tulokset: Oletko käyttänyt Surveypal -sovellusta jollain mobiililaitteella, kuten tablet-tietokoneella tai älypuhelimella? (taulukko E.1)

Keskimääräinen vastaaja oli yli 30-vuotias nainen, joka suorittaa hieman yli puolet internetselailustaan mobiililaitteilla. Surveypal-sovellusta keskimääräinen vastaaja ei ole kuitenkaan mobiililaitteella käyttänyt. Vain 17,2% vastaajista oli joskus käyttänyt Surveypal-sovellusta mobiililaitteilla (Kuva 5.1). 49 vastaajaa (10,7% vastaajista) oli käyttänyt sovellusta tablet-tietokoneella ja 47 vastaajaa (10,3% vastaajista) älypuhelimella. 20 vastaajaa (4,4% vastaajista) oli käyttänyt sovellusta sekä tablet-tietokoneella että älypuhelimella. Henkilöt, jotka eivät olleet käyttäneet Surveypal-sovellusta koskaan mobiililaitteella sanoivat kuitenkin suorittavansa keskimäärin puolet internetselailustaan mobiililaitteilla.

Sovellusta mobiililaitteilla käyttäneet vastaajat antoivat mobiilikäytön sujuvuudelle keskimäärin arvosanan 6,5 asteikolla 0-10. Pyydettyessä vastaajia kuvailemaan sovelluksen käyttökokemustaan mobiililaitteillaan, esiintyi vastauksissa erityisesti seuraavanlaisia teemoja:



1. Vastaaaja oli tyytyväinen sovelluksen toimintaan
2. Sovellus toimi (latautui) vastaajan mielestä hitaasti, tai jokin sen näkymä ei toiminut lainkaan
3. Vastaaaja oli joutunut zoomaamaan sovelluksen näkymiä
4. Sovelluksen painikkeet ja käyttöliittymätekstit ovat vastaajan mukaan liian pieniä
5. Sovelluksen käyttö oli vastaajan mielestä hankalaa tietokonekäyttöön verrattuna

Osa vastaajista ajatteli heidän käyttämänsä laitteen tai verkkoyhteyden aiheuttaneen mainitun epätoivotun toiminnan sovelluksen käytössä.

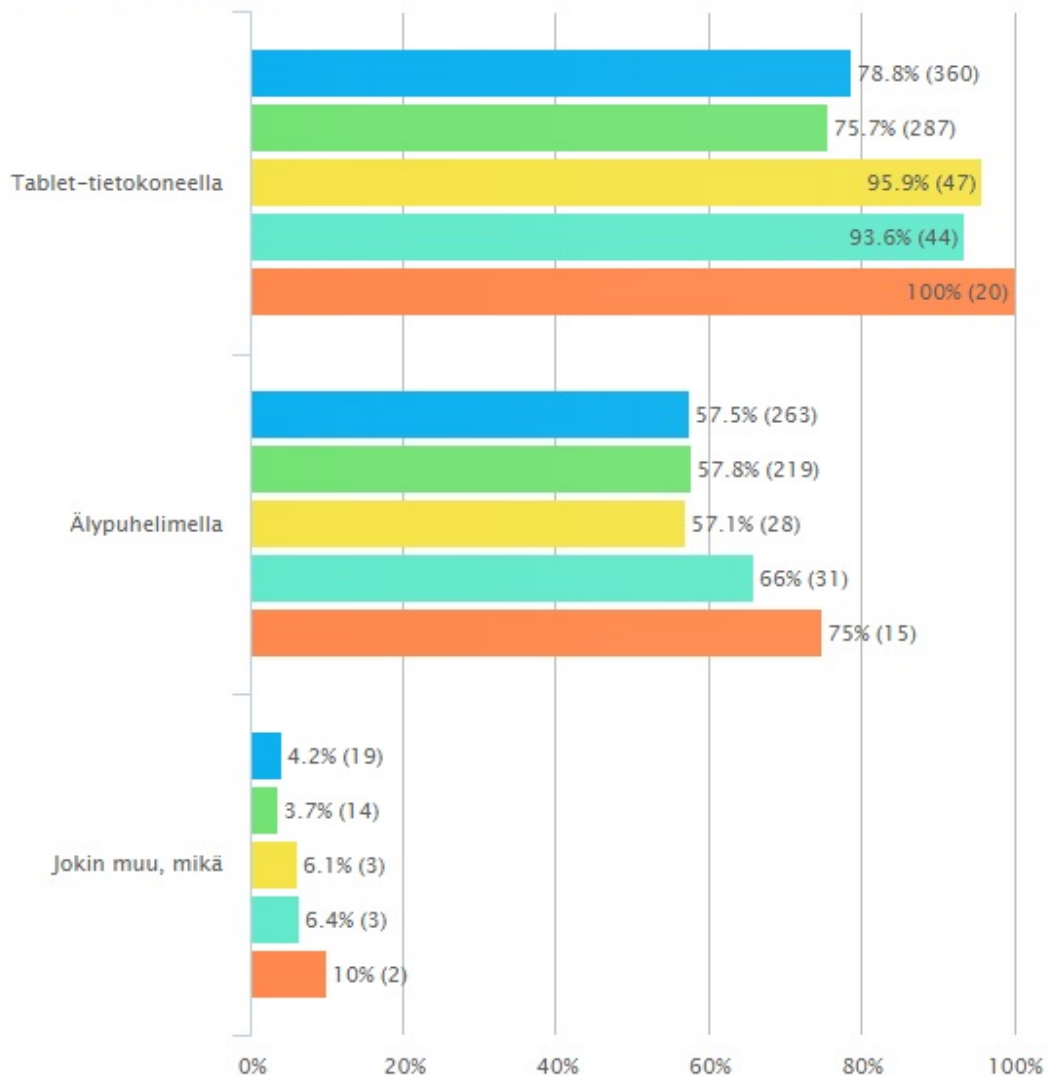
Kuten taulukossa C.27 sivulla 89 todettiin, sovelluksessa ei käytetä nykyisellään *meta viewport* -elementtiä. Tämä on syy tai osasy teemojen 3, 4 ja 5 mukaisiin ongelmiin. Teeman 5 mukaista ongelmaa aiheuttaa myös epätäydellinen varautuminen erilaisiin interaktiometodeihin, jota käsiteltiin taulukossa C.20 sivulla 84.

Sovelluksen latautuminen ja toimiminen hitaasti voi johtua lukuisista eri asioista. Kuten vastaajat itsekin olivat päätelleet, on laite ja varsinkin heikko verkkoyhteys saattanut hidastaa sovelluksen toimintaa. Muun muassa verkkopyyntöjen paremmalla optimoinnilla voitaisiin nopeuttaa sovelluksen toimintaa (taulukko C.10 sivulla 77).

Kaikilta vastaajilta kysyttiin, millä mobiililaitteella he voisivat kuvitella käyttävänsä Surveypal-sovellusta (kuva 5.2). Vastaajista 78,8% ajatteli voivansa käyttää sovellusta tablet-tietokoneella ja 57,5% älypuhelimella. Vastaajista, joilla oli kokemusta sovelluksen käytöstä mobiililaitteilla, jopa 93,6% voisi kuvitella käyttävänsä sovellusta tablet-tietokoneella. Vastaajien mielipiteet siitä, voisivatko he käyttää sovellusta älypuhelimella eivät juuri eronneet toisistaan riippumatta siitä, oliko vastaaja käyttänyt sovellusta mobiilissa vai ei.

### Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal-sovellusta?

Yksittäisiä vastaajia 457



**■ Kaikki (Keskiarvo: 1.47; Keskihajonta: 0.56)**

**■ Ei ole käyttänyt sovellusta mobiililaitteella (Keskiarvo: 1.48; Keskihajonta: 0.55)**

**■ On käyttänyt sovellusta (ainoastaan) tablet-tietokoneella (Keskiarvo: 1.44; Keskihajonta: 0.57)**

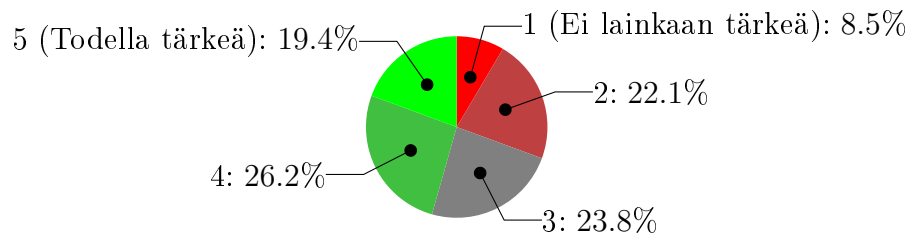
**■ On käyttänyt sovellusta (ainoastaan) älypuhelimella (Keskiarvo: 1.47; Keskihajonta: 0.57)**

**■ On käyttänyt sovellusta sekä tablet-tietokoneella että älypuhelimella (Keskiarvo: 1.51; Keskihajonta: 0.6)**

Kuva 5.2: Tulokset: Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal-sovellusta? (Taulukko E.5)

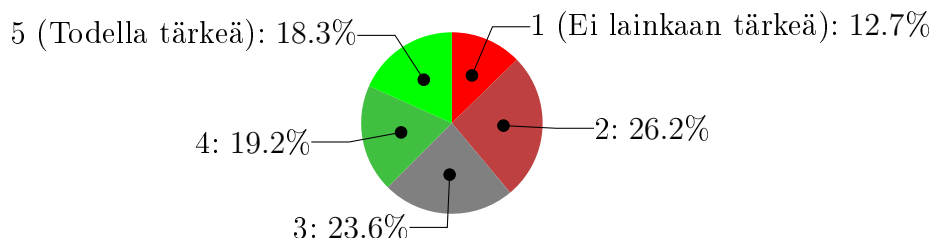
Vastaajia pyydettiin arvioimaan, miten tärkeitä Surveypal-sovelluksen eri ominaisuuudet olisivat heille mobiilissa kontekstissa. Jotkut ominaisuudet koettiin selkeästi toisia tärkeämmiksi, mutta yhtäkään ominaisuutta ei varsinaisesti pidetty täysin tarpeettomana mobiilissa kontekstissa. Ominaisuudet jaoteltiin seuraavasti:

- Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen) (kuvat A.2 sivulla 60 ja B.2 sivulla 65)
- Kyselyn luominen ja muokkaaminen (kuvat A.3 ja B.3)
- Kyselyn julkaiseminen (kuvat A.3 ja B.3)
- Vastausten kerääminen (kutsujen lähettäminen ja kyselylinkin jakaminen) (kuvat A.7 ja B.7)
- Tulosten katselu (kuvat A.10 ja B.10)



Kuva 5.3: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - **Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen)** (taulukko E.6)

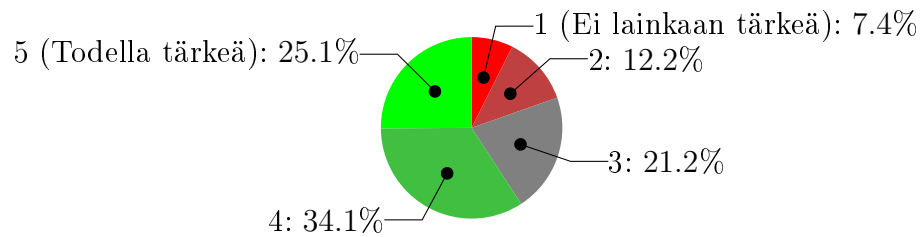
45,6% vastaajista piti kyselyiden hallintaa jossain määrin tärkeänä myös mobiilissa ympäristössä (kuva 5.3). Toisaalta jopa 30,6% vastaajista ei pitänyt ominaisuutta kovin tärkeänä. Kyseinen ominaisuuden tärkeys jakoi siis mielipiteet kohtalaisen voimakkaasti. Avoimissa vastauksissa kyselyiden hallinta mainittiin myös useaan otteeseen todennäköisenä käyttötapauksena mobiiliympäristössä.



Kuva 5.4: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - **Kyselyn luominen ja muokkaaminen** (taulukko E.7)

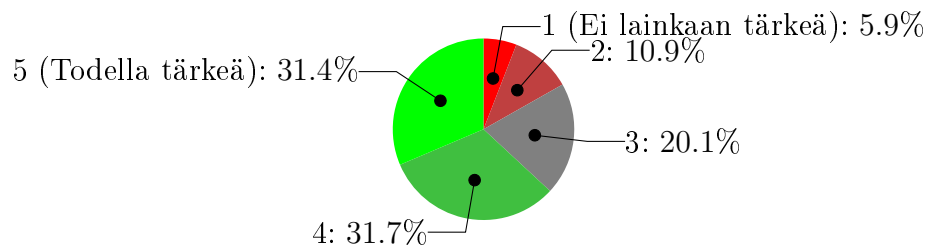
Kyselyiden luominen ja muokkaaminen oli vastaajien mielestä eritellyistä ominaisuuksista vähiten tärkeä (kuva 5.4). Jopa 38,9% vastaajista koki ominaisuuden vähemmän tärkeäksi mobiilissa kontekstissa. Mielipiteet jakautuivat myös tasaisesti

kyseisen ominaisuuden kohdalla. 37,5% vastaajista piti ominaisuutta jossain määrin tärkeänä. Mielenpitojen jakaantuminen näkyi selkeästi myös avoimissa vastauksissa. Useat vastaajat sanoivat, etteivät todennäköisesti loisi uutta kyselyä mobiilissa. Osa vastaajista eritteli erityisesti mobiililaitteiden rajoittuneisuuden (pieni näyttö ja fyysisen näppäimistön puute) syyksi, miksi he eivät kokeneet kyselyn luomista todennäköiseksi käyttötapaukseksi. Toisaalta kyselyn pienimuotoinen muokkaaminen, virheiden korjaaminen ja pienen pikakyselyn luominen koettiin todennäköiseksi käyttötapaukseksi useiden vastaajien toimesta.



Kuva 5.5: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - **Kyselyn julkaiseminen** (taulukko E.8)

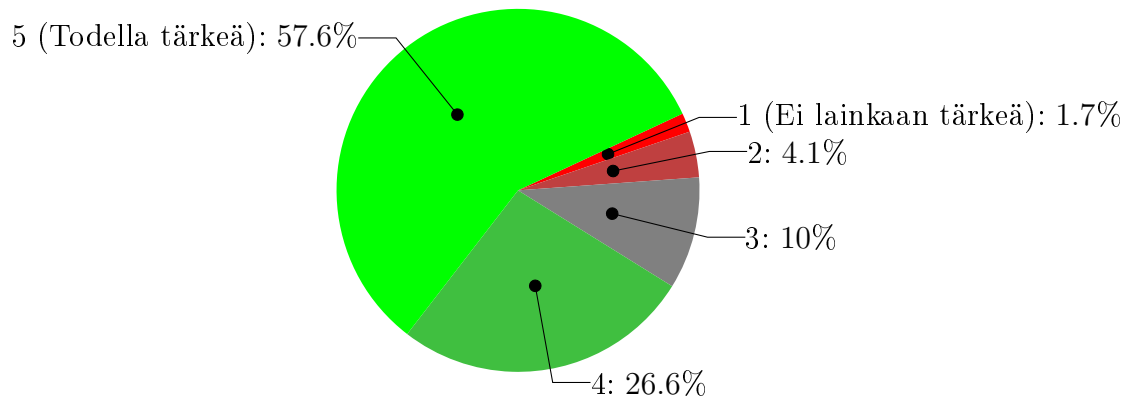
Yli puolet (59,2%) vastaajista piti kyselyiden julkaisemista tärkeänä ominaisuutena myös mobiiliympäristössä (kuva 5.5). Neljäsosa vastaajista piti ominaisuutta jopa erittäin tärkeänä. Avoimissa vastauksissa kyselyn julkaiseminen mainittiin useaan kertaan ja sitä pidettiin todennäköisenä käyttötapauksena yksinään (tietokoneella luodun kyselyn julkaiseminen matkalla, kokouksessa tai luennolla) ja pienen pikakyselyn luonnin yhteydessä.



Kuva 5.6: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - **Vastausten kerääminen** (kutsujen lähettäminen ja kyselylinkin jakaminen) (taulukko E.9)

Jopa 63,1% vastaajista koki vastausten keräämisen tärkeänä ominaisuutena myös mobiilipuolella (kuva 5.6). Useissa avoimissa vastauksissa mainittiin vastausten kerääminen todennäköisenä käyttötapauksena kutsujen ja muistutusten lähettämisen muodossa. Avoimista vastauksista kävi myös ilmi, että kysymyksen ominaisuus oli tiin jossain tapauksissa saatettu ymmärtää virheellisesti itse kyselyyn vastaamisena. Eräs vastaaja mainitsi käyttötapaukseksi vastausten keräämisen tapahtumassa, josta saattoi ymmärtää vastaajan tarkoittavan hänen tahtovansa käyttää mobiililaitetta itse kyselyn näyttämiseen ja siihen vastaamiseen, mikä ei kuitenkaan kuulunut

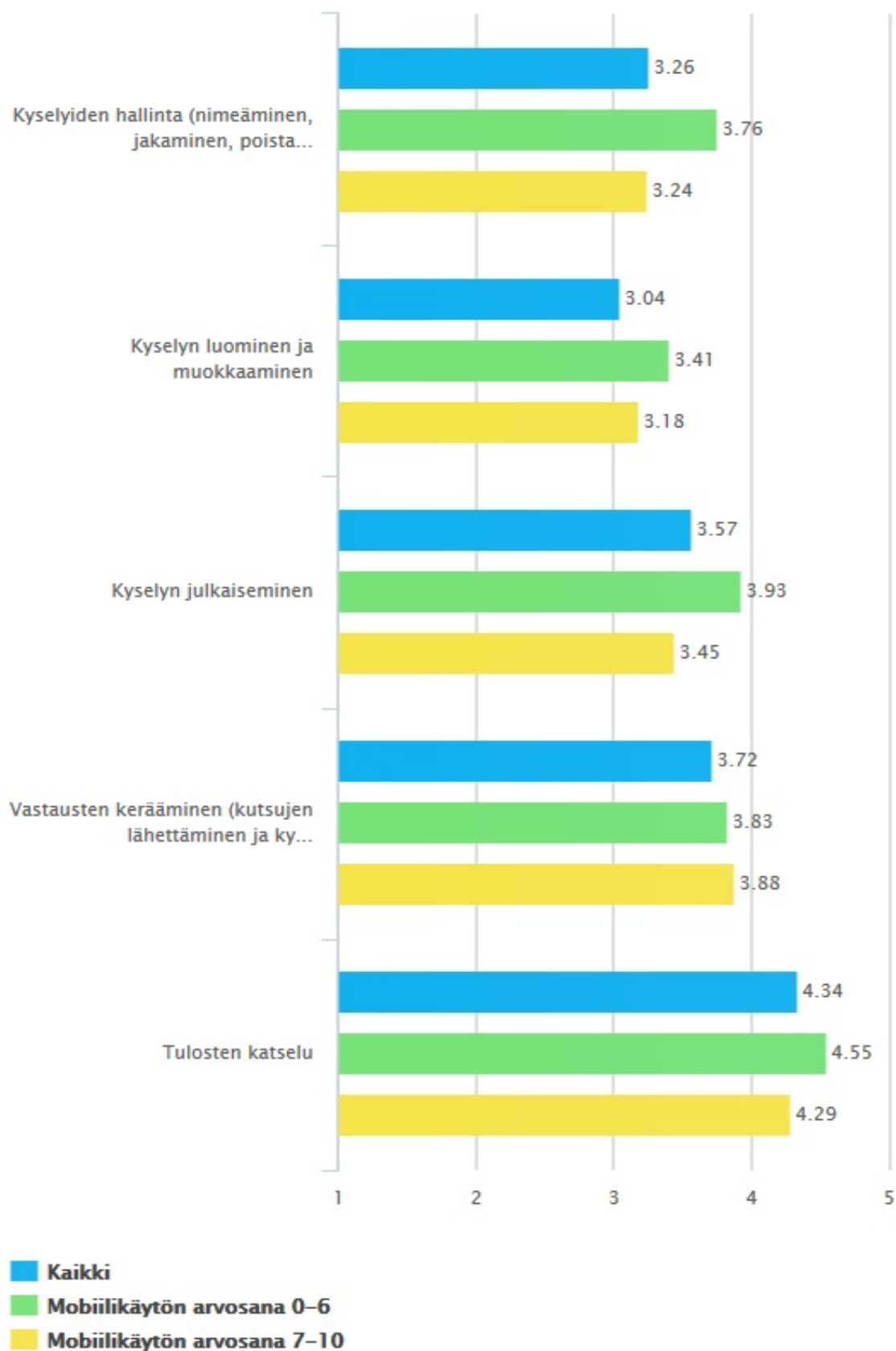
tämän kyselyn piiriin.



Kuva 5.7: Tulokset: Arvio, miten tärkeitä SurveyPal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - **Tulosten katselu** (taulukko E.10)

Tulosten katselu oli vastaajien mielestä ylivoimaisesti tärkein ominaisuus mobiilikäytössä. Suuri enemmistö (84,2% vastaajista) piti ominaisuutta tärkeänä ja yli puolet (57,6%) jopa erittäin tärkeänä (kuva 5.7). Tulosten katselu esiintyi myös suuressa osassa avoimia vastauksia todennäköisenä käyttötapauksena. Useat vastaajat kokivat myös hieman kevyemmän käyttötapauksen, vastausmäärän kehittymisen tarkastelun tärkeäksi ominaisuudeksi mobiilikäytössä.

Kyselyn tuloksista kävi kaikenkaikkiaan ilmi, että vaikka SurveyPal-sovelluksen käyttäjät eivät nykyisellään käytä sovellusta mobiililaitteilla kovin laajasti, olisi sovelluksella kuitenkin paljon potentiaalia mobiilikäytössä. Sovelluksen kaikkia ominaisuuksia ei koettu yhtä tärkeiksi mobiilikäytössä. Tästä johtuen erillisen, mobiiliin kontekstiin suunnitellun, eri ominaisuuksilla varustetun mobiiliversion toteuttaminen sovelluksesta olisi tältä osin perusteltua. Toisaalta kaikki eriteltyt ominaisuudet koettiin useammin ”todella tärkeiksi” kuin ”ei lainkaan tärkeiksi”, mistä voidaan päätellä sovelluksen kaikilla ominaisuuksilla olevan käyttöä myös mobiilissa. Henkilöt, jotka olivat käyttäneet sovellusta mobiililaitteella ja antaneet kokemalleen mobiilikäytön sujuvuudelle keskimääräistä huonomman arvosanan, pitivät sovelluksen ominaisuuksia tärkeämpänä mobiilissa kuin muut (kuva 5.8). Tästä näkökulmasta tutkittuna sovelluksen nykyisen toteutuksen muuttaminen mukautuvaksi olisi paremmin perusteltu ratkaisu. Mobiilikäyttäjille tarjottava käyttöliittymä tarvitsee joka tapauksessa merkittävän määrän huomiota, riippumatta siitä, kumpi kahdesta toteutustavasta valitaan. Sovelluksen käyttöliittymäelementit on suunniteltava pienelle mobiililaitteen ruudulle sopivaksi ja erityisesti kosketus syötetapana on otettava paremmin huomioon.



Kuva 5.8: Tulokset: Arvioi, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä (taulukot E.6 - E.10)

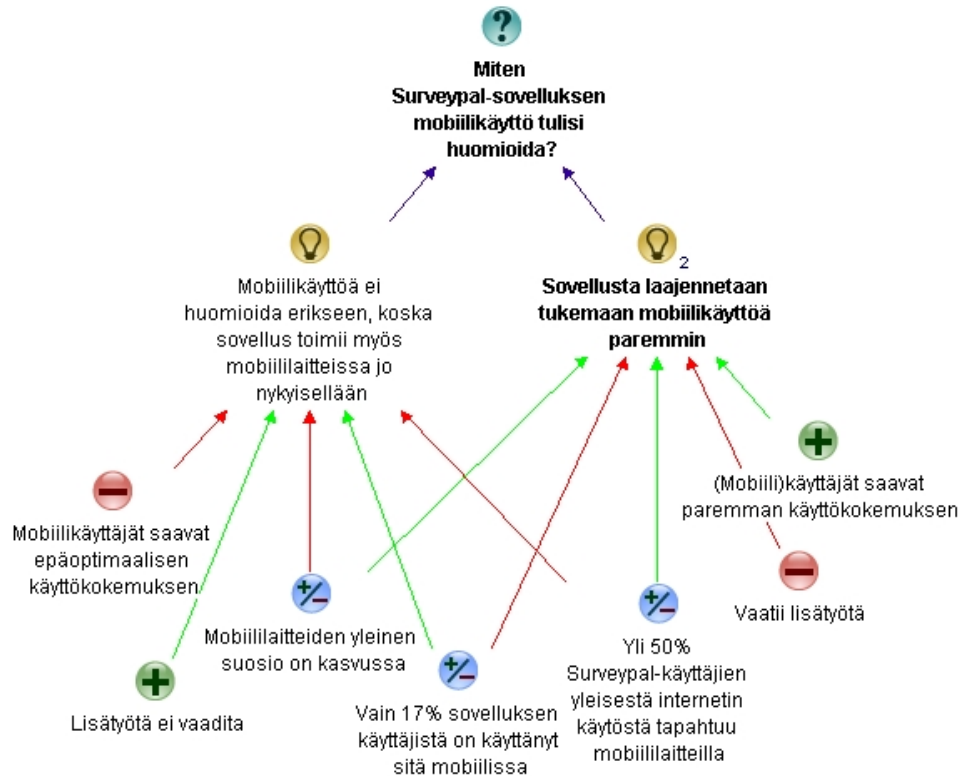
## 6. SUOSITUS SURVEYPAL WEB-OHJELMAN MOBIILITOTEUTUKSEN TOTEUTUSTAVASTA

Tässä luvussa esitetään suositus Surveypal web-ohjelman mobiilitoteutuksen toteutustavasta perusteluineen. Perustelu muodostetaan *IBIS*-metodiin pohjautuvan ongelmakartan avulla. Perustelun argumentteina toimivat luvuissa 3.1.4 (sivulla 19) ja 3.2.2 (sivulla 24) listatut eri toteutustapojen hyvät ja huonot puolet sekä toteutustapojen suhtautuminen Surveypal-sovelluksen nykytilaan (luku 5.1). Argumentoinnissa käytetään apuna myös Surveypal-käyttäjille esitetyn kyselyn tuloksia (luku 5.2).

Sovelluksesta suositellaan tekemään versio, joka toimii myös mobiililaitteilla (kuva 6.1 sivulla 51). Sovelluksen mobiilitoteutuksen toteutustavaksi suositellaan nykyisen toteutuksen muuttamista mukautuvaksi (kuva 6.2 sivulla 52). Hylättyjä ratkaisuvaihtoehtoja olivat mobiilikäytön huomiotta jättäminen sekä erillisen mobiilitoteutuksen luominen nykyisen tietokoneelle suunnatun toteutuksen rinnalle.

Mobiililaitteiden yleisen suosion ja Surveypal-käyttäjien mobiilin internetselailun suuren osuuden johdosta sovelluksen laajentamista mobiilikäyttöä paremmin tukeväksi voidaan pitää parhaana ratkaisuna. Tärkeimpinä perusteluita nykyisen toteutuksen mukautuvaksi muuttamisen puolesta olivat:

- Nykyinen toteutus täyttää suuren osan W3C:n esittämistä mobiilin web-sovelluksen parhaista käytännöistä
- Mukautuva, kaikki sovelluksen ominaisuudet kattava toteutus ei rajoita sovelluksen käyttömahdollisuuksia mobiililaitteilla
- Erillisen mobiilitoteutuksen tuoman erillisen koodikannan ja siitä johtuvan työmäärän lisääntymisen välttäminen
- Surveypal-käyttäjien arvio sovelluksen eri ominaisuuksien tärkeydestä
- Erilaisten laitteiden tukemisen yksinkertaisuus

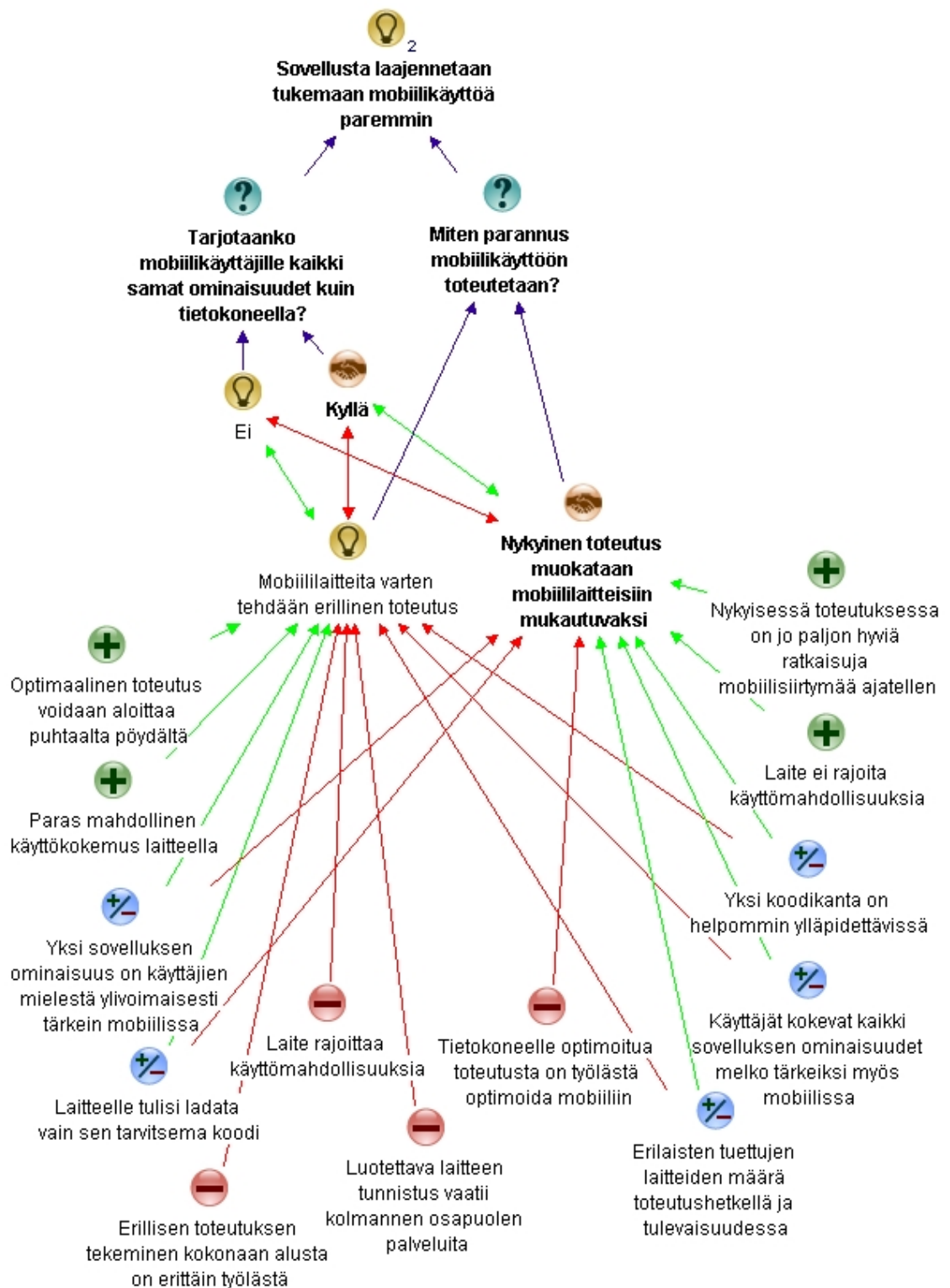


Kuva 6.1: Suositus Surveypal-sovelluksen mobiilituen toteutustavasta perusteltuna *IBIS*-metodin avulla 1/2

Tärkein erillisen mobiiliversion tekemistä vastaan olevista argumenteista on sen (kokoonaan uuden sovelluksen) vaatima huomattavasti suurempi työmäärä verrattaessa nykyisen toteutuksen pohjalta tehtyyn mukautuvaan toteutukseen. Koska käyttäjät arvioivat heille esitetyssä kyselyssä sovelluksen eri ominaisuudet keskimäärin tärkeiksi myös mobiilissa, pitäisi sovelluksesta tehdä joka tapauksessa sen ominaisuuksien osalta hyvin saman kaltainen versio. Tämä puolestaan johtaisi väistämättä toisteiseen työhön ja kevyemmästä versiosta saadun hyödyn menettämiseen ainakin osittain. Myös palvelinpään laitetunnistuksen epätäydellisyys ja maksullisuus toimivat argumentteina erillistä mobiilitoteutusta vastaan.

Suurin vaadittu muutos sovelluksen nykytilaan muutettaessa sitä mukautuvaksi on uudet käyttöliittymäasettelut eri kokoisilla näyttöresoluutiolla. Jotta asettelusta voitaisiin saada optimaalinen myös pienellä mobiililaitteen näytöllä, tarvitaan kohtalaisen isoja muutoksia sovelluksen HTML- ja CSS-koodiin. Myös kosketuksen kattava tukeminen syötetapana on kohtuullisen työmäärän vaativa projekti.





Kuva 6.2: Suositus Surveypal-sovelluksen mobiilituen toteutustavasta perusteltuna IBIS-metodin avulla 2/2

## 7. YHTEENVETO

Tässä diplomityössä tutkittiin, miten Surveypal web-sovelluksen mukautuminen mobiililaitteissa olisi paras toteuttaa. Web-ohjelmien erityispiirteitä ja niistä aiheutuvia ongelmia käytiin läpi yleisesti sekä mobiileihin web-sovelluksiin syvemmin pureutuen. Tietokoneilla käytettäväksi suunniteltujen web-sovellusten käytettävyys mobiililaitteilla todettiin myös olevan epäoptimaalinen.

Käytettävyyden parantamiseksi esitettiin kaksi eri tapaa toteuttaa web-sovelluksen mukautuminen erilaisilla laitteilla. Toteutustavoista ensimmäinen, erilliset toteutukset, todettiin parhaaksi mahdolliseksi vaihtoehdoksi tilanteissa, joissa mobiilikäyttäjien etsimä informaatio ja käyttötapaukset eroavat merkittävästi tietokokäyttäjien vastaavista. Tällaisissa tilanteissa mobiililaitteille tarjottava versio sovelluksesta voidaan tehdä sisällöltään erilaiseksi ja kevyemmäksi. Erillisten toteutusten huonoiksi puoliksi todettiin muun muassa kaikkien laitteiden kattamisen hankaluus ja erillisestä toteutuksesta aiheutuva erillinen koodikanta. Toinen esitelty toteutustapa, yksi mukautuva toteutus, todettiin paremmaksi vaihtoehdoksi tilanteessa, jossa tietokone- ja mobiilikäyttäjille halutaan tarjota sama sisältö. Mukautuvan toteutuksen selkeimpiä etuja verrattuna erillisiin toteutuksiin ovat yksittäisen koodikannan avulla saavutettu verrattain helppous ylläpidossa sekä tuettujen laitteiden skaala. Koska mukautuva toteutus sisältää aina koodia, jota ei voida hyödyntää käyttäjän päätelaitteella, ei sillä voida saavuttaa loppukäyttäjän kannalta yhtä optimoitua lopputulosta kuin erillisillä toteutuksilla.

Surveypal-sovelluksen nykytilan sekä mobiilitoteutuksen eri toteutusvaihtoehtojen suhtautumista W3C:n antamiin suosituksiin mobiileille web-sovelluksille käytiin läpi suosituskohtaisesti. Kävi ilmi, että Surveypal-sovelluksen nykyinen toteutus täyttää jo suuren osan suosituksista ja toimisi täten kohtalaisen hyvänä pohjana mukautuvalle toteutukselle. Osa suosituksista todettiin hieman hankalammiksi täyttää nykyisen toteutuksen pohjalta, mikä puolestaan osaltaan puoltaisi erillisen toteutuksen tekemistä mobiililaitteille. Toteutustavat pisteytettiin suosituskohtaisesti ja paremmaksi ratkaisuksi osoittautui mukautuva toteutus pistein 67/105. Erilliset toteutukset saivat 50/105 pistettä jääden 17 pisteellä jälkeen mukautuvasta toteutuksesta.

Surveypal-käyttäjien mielipiteitä sovelluksen nykytilasta ja sen potentiaalisista käyt-

tötapauksista mobiilissa tiedusteltiin kyselyn muodossa. Kyselyn tuloksista kävi ilmi, että vain 17 % sovelluksen käyttäjistä oli käyttänyt sovellusta myös mobiililaitteilla. Käyttäjät sanoivat kuitenkin suorittavansa yli puolet yleisestä internet-selailustaan mobiililaitteilla. Kaikkia sovelluksen ominaisuuksia pidettiin tärkeinä myös mobiilissa, mutta kyselyn tulosten tarkastelua pidettiin muita ominaisuuksia huomattavasti tärkeämpänä.

Surveypal-sovelluksen mobiilitoteutuksen suositelluksi toteutustavaksi perusteltiin yksi, mukautuva toteutus, joka toteutettaisiin sovelluksen nykyisen toteutuksen pohjalta. Perusteluina tälle suositukselle toimivat mukautuvan toteutuksen yleiset piirteet, Surveypal-sovelluksen hyvä nykytila mobiilisiirtymää ajatellen, sekä kyselyn tuloksista selvinnyt sovelluksen ominaisuuksien arvioitu tärkeys mobiilissa.

Surveypal-sovellukselle perusteltua suositusta ei voida soveltaa universaalisti mihin tahansa web-sovellukseen. Sekä erillisille laitekohtaisille toteutuksille, että yksittäiselle mukautuvalle toteutukselle on omat parhaat käyttökohteensa, eikä toinen toteutustapa ole aina automaattisesti parempi kuin toinen. Tässä diplomityössä käytettyjen perustelujen avulla voidaan perustella erilliset toteutukset parhaaksi vaihtoehdoksi jonkin muun web-sovelluksen tapauksessa. Yleisesti hyväksytyjen parhaiden käytäntöjen löytämiseen on kuitenkin vielä pitkä matka ja aiheessa on vielä paljon tutkittavaa.

## LÄHTEET

- [1] Vora, P. 2009. Web Application Design Patterns. Burlington, Morgan Kaufmann Publishers. 429 s.
- [2] Web Storage [WWW]. [Viitattu 7.9.2014]. Saatavissa <http://www.w3.org/TR/webstorage/>
- [3] 5.6 Offline Web applications - HTML5 [WWW]. [Viitattu 7.9.2014]. Saatavissa <http://www.w3.org/TR/2011/WD-html5-20110525/offline.html>
- [4] Can I use... Support tables for HTML5, CSS3, etc [WWW]. [Viitattu 7.9.2014]. Saatavissa <http://caniuse.com/>
- [5] Optimizing content for different browsers: the RIGHT way - Web Education Community Group [WWW]. [Viitattu 7.9.2014]. Saatavissa [https://www.w3.org/community/webed/wiki/Optimizing\\_content\\_for\\_different\\_browsers:\\_the\\_RIGHT\\_way](https://www.w3.org/community/webed/wiki/Optimizing_content_for_different_browsers:_the_RIGHT_way)
- [6] Report: Touchscreen demand to grow by 90-percent, led by mobile, tablet markets [WWW]. [Viitattu 10.9.2014]. Saatavissa <http://www.engadget.com/2011/07/13/report-touchscreen-demand-to-grow-by-90-percent-led-by-mobile/>
- [7] Kadlec, T. 2013. Implementing Responsive Design. Berkeley, New Riders. 271 s.
- [8] The Ultimate Guide to 4K and 8K Ultra HD [WWW]. [Viitattu 14.9.2014]. Saatavissa <http://www.ultrahdtv.net/the-ultimate-guide-to-4k-and-8k-ultra-hd/>
- [9] StatCounter Global Stats [WWW]. [Viitattu 3.1.2015]. Saatavissa <http://gs.statcounter.com/>
- [10] Google Nexus 5 [WWW]. [Viitattu 2.1.2015]. Saatavissa <http://www.google.com/nexus/5/>
- [11] Samsung 58" Smart Full HD LED TV H5204 [WWW]. [Viitattu 2.1.2015]. Saatavissa <http://www.samsung.com/fi/consumer/tv-home-theatre/tv/smart-tv/UE58H5204AKXXE>
- [12] Marcotte, E. 2011. Responsive Web Design. New York, A Book Apart. 150 s.
- [13] Scrolling and Scrollbars [WWW]. [Viitattu 14.9.2014]. Saatavissa <http://www.nngroup.com/articles/scrolling-and-scrollbars/>

- [14] What Exactly is... The 300ms Click Delay [WWW]. [Viitattu 14.9.2014]. Saatavissa <http://blogs.telerik.com/appbuilder/posts/13-11-21/what-exactly-is.....-the-300ms-click-delay>
- [15] Wroblewski, L. 2011. Mobile First. New York, A Book Apart. 130 s.
- [16] input - input control (CHANGED) - HTML5 [WWW]. [Viitattu 15.1.2015]. Saatavissa <http://www.w3.org/TR/html-markup/input.html>
- [17] Safari HTML Reference: Supported Meta Tags [WWW]. [Viitattu 8.8.2014]. Saatavissa <https://developer.apple.com/library/safari/documentation/AppleApplications/Reference/SafariHTMLRef/Articles/MetaTags.html>
- [18] Dev.Opera - An Introduction to Meta Viewport and @viewport [WWW]. [Viitattu 8.8.2014]. Saatavissa <http://dev.opera.com/articles/an-introduction-to-meta-viewport-and-viewport/>
- [19] Website Dimensions and Designing for the Web | Dare to Think [WWW]. [Viitattu 30.8.2014]. Saatavissa <http://www.daretothink.co.uk/blog/website-dimensions-and-designing-for-the-web/>
- [20] Case Study from Breastcancer.org: Choosing between responsive web design and a separate mobile site to improve mobile visitors experience [WWW]. [Viitattu 2.10.2014]. Saatavissa <http://www.nten.org/articles/2013/case-study-from-breastcancerorg-choosing-between-responsive-web-design-and-a-separate-mobile-site-to-improve-mob-0>
- [21] Detect Mobile Browsers - Open source mobile phone detection [WWW]. [Viitattu 1.10.2014]. Saatavissa <http://detectmobilebrowsers.com/>
- [22] Dev.Opera - Grid Design Basics: Grids for Web Page Layouts [WWW]. [Viitattu 30.8.2014]. Saatavissa <https://dev.opera.com/articles/grids-for-web-page-layouts/>
- [23] 51Degrees - Prices [WWW]. [Viitattu 2.10.2014]. Saatavissa <http://51degrees.com/Prices>
- [24] WURFL Java API 1.5 [WWW]. [Viitattu 2.10.2014]. Saatavissa [http://wurfl.sourceforge.net/njava\\_index.php](http://wurfl.sourceforge.net/njava_index.php)
- [25] WURFL Licensing [WWW]. [Viitattu 2.10.2014]. Saatavissa <http://wurfl.sourceforge.net/license.php>
- [26] Safari Web Content Guide: Configuring the Viewport [WWW]. [Viitattu 11.10.2014]. Saatavissa

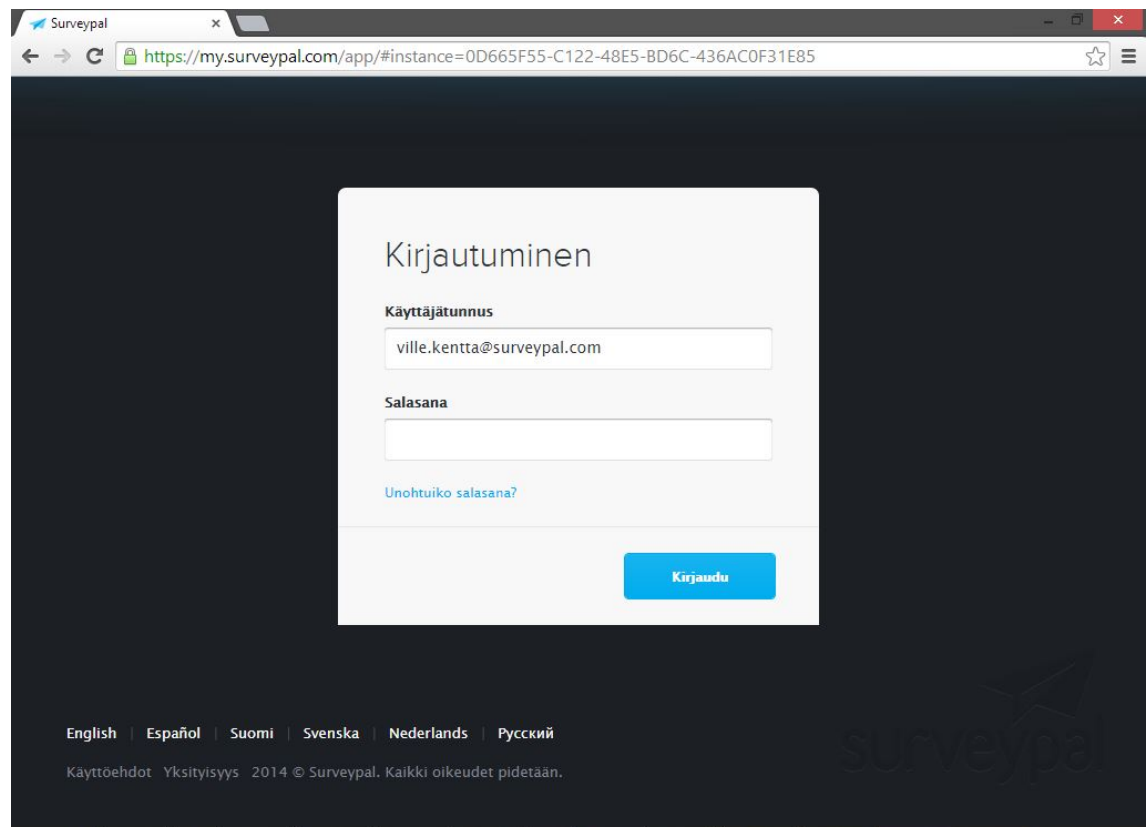
<https://developer.apple.com/library/ios/documentation/appleapplications/reference/safariwebcontent/usingtheviewport/usingtheviewport.html>

- [27] How to set WebBrowser control viewport dimensions on Windows Phone - Wiki [WWW]. [Viitattu 11.10.2014]. Saatavissa [http://developer.nokia.com/community/wiki/How\\_to\\_set\\_WebBrowser\\_control\\_viewport\\_dimensions\\_on\\_Windows\\_Phone](http://developer.nokia.com/community/wiki/How_to_set_WebBrowser_control_viewport_dimensions_on_Windows_Phone)
- [28] A Pixel Identity Crisis · An A List Apart Article [WWW]. [Viitattu 11.10.2014]. Saatavissa <http://alistapart.com/article/a-pixel-identity-crisis/>
- [29] Apple - iPhone 5s - Tekniset tiedot [WWW]. [Viitattu 11.10.2014]. Saatavissa <https://www.apple.com/fi/iphone-5s/specs/>
- [30] Fling, B. 2009. Mobile Design and Development. Sebastopol, O'Reilly. 309 s.
- [31] Adapting To A Responsive Design (Case Study) - Smashing Magazine [WWW]. [Viitattu 2.10.2014]. Saatavissa <http://www.smashingmagazine.com/2013/06/18/adapting-to-a-responsive-design-case-study/>
- [32] Tablet ja matkapuhelin - ASUS Transformer Pad - ASUS [WWW]. [Viitattu 20.10.2014]. Saatavissa [http://www.asus.com/fi/Tablets\\_Mobile/Transformer\\_Pad\\_Products/](http://www.asus.com/fi/Tablets_Mobile/Transformer_Pad_Products/)
- [33] Lenovo Touch Screen PC: Discover the Horizon 27 Table PC [WWW]. [Viitattu 20.10.2014]. Saatavissa <http://shop.lenovo.com/fi/fi/desktops/lenovo/horizon-series/horizon-27/>
- [34] Scaffolding - Bootstrap [WWW]. [Viitattu 26.10.2014]. Saatavissa <http://getbootstrap.com/2.3.2/scaffolding.html#responsive>
- [35] Responsive.js [WWW]. [Viitattu 31.10.2014]. Saatavissa <http://www.responsivejs.com/>
- [36] Respond.js [WWW]. [Viitattu 31.10.2014]. Saatavissa <https://github.com/scottjehl/Respond>
- [37] 960 Grid System [WWW]. [Viitattu 30.8.2014]. Saatavissa <http://960.gs/>
- [38] John Resig - Simple JavaScript Inheritance [WWW]. [Viitattu 16.11.2014]. Saatavissa <http://ejohn.org/blog/simple-javascript-inheritance/>
- [39] jQuery [WWW]. [Viitattu 8.8.2014]. Saatavissa <http://jquery.com/>
- [40] RequireJS [WWW]. [Viitattu 8.8.2014]. Saatavissa <http://requirejs.org/>

- [41] MooTools - a compact javascript framework [WWW]. [Viitattu 8.8.2014]. Saatavissa <http://mootools.net/>
- [42] Highcharts [WWW]. [Viitattu 8.8.2014]. Saatavissa <http://www.highcharts.com/products/highcharts>
- [43] Mobile Web Application Best Practices, W3C Recommendation 14 December 2010 [WWW]. [Viitattu 28.11.2014]. Saatavissa <http://www.w3.org/TR/mwabp/>
- [44] Responsive Web Design - An A List Apart Article [WWW]. [Viitattu 6.1.2015]. Saatavissa <http://alistapart.com/article/responsive-web-design>
- [45] Mobile Web Application Best Practices, W3C Working Draft 29 July 2008 [WWW]. [Viitattu 6.1.2015]. Saatavissa: <http://www.w3.org/TR/2008/WD-mwabp-20080729/#bp-devcap-detection>
- [46] Werner, K., Rittel, H. 1970. Issues as Elements of Information Systems, Working paper No. 131. Heidelberg, Studiengruppe für Systemforschung
- [47] Okada, A., Shum, S.J.B. ja Sherborne, T. 2008, Knowledge Cartography: software tools and mapping techniques. London, Springer, 400 s.
- [48] Compendium [WWW]. [Viitattu 2.1.2015]. Saatavissa <http://compendiuminstitute.net/>

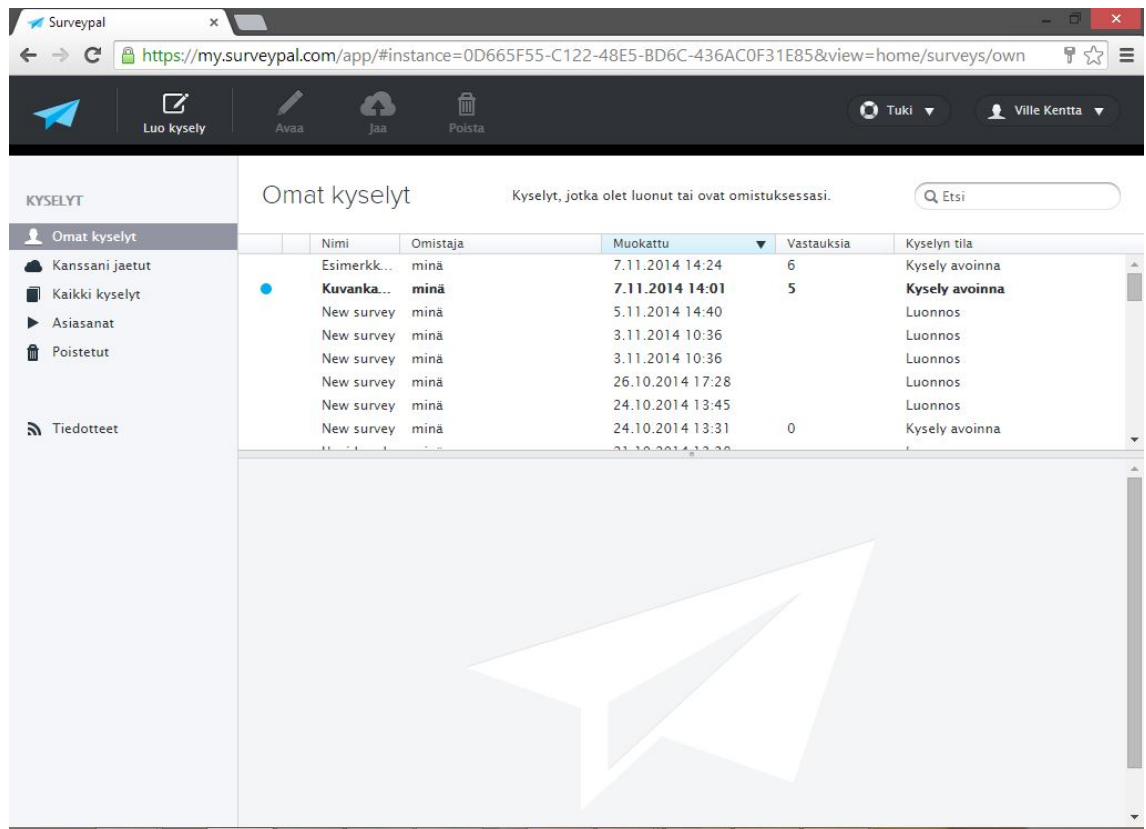
## A. LIITE - SURVEYPAL - KUVANKAAPPAUKSET (TYÖPÖYTÄTIETOKONE)

Tässä liitteessä esitellään Surveypal - kyselytyökalun tärkeimmät näkymät työpöytätielokoneella avattuna. Kuvankaappauksissa käytettiin *Google Chrome* selainta ja *Windows 8.1* käyttöjärjestelmää näyttöresoluution ollessa 1024 \* 768 pikseliä.

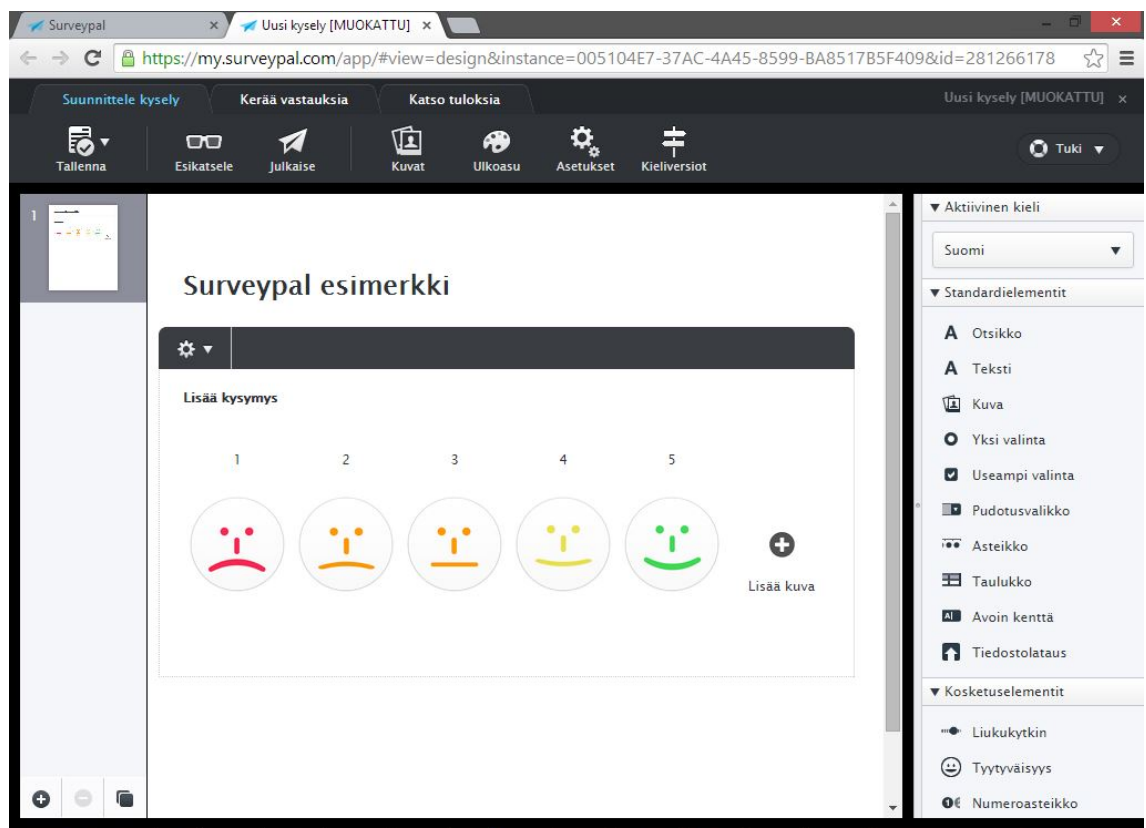


Kuva A.1: Surveypal - kirjautuminen

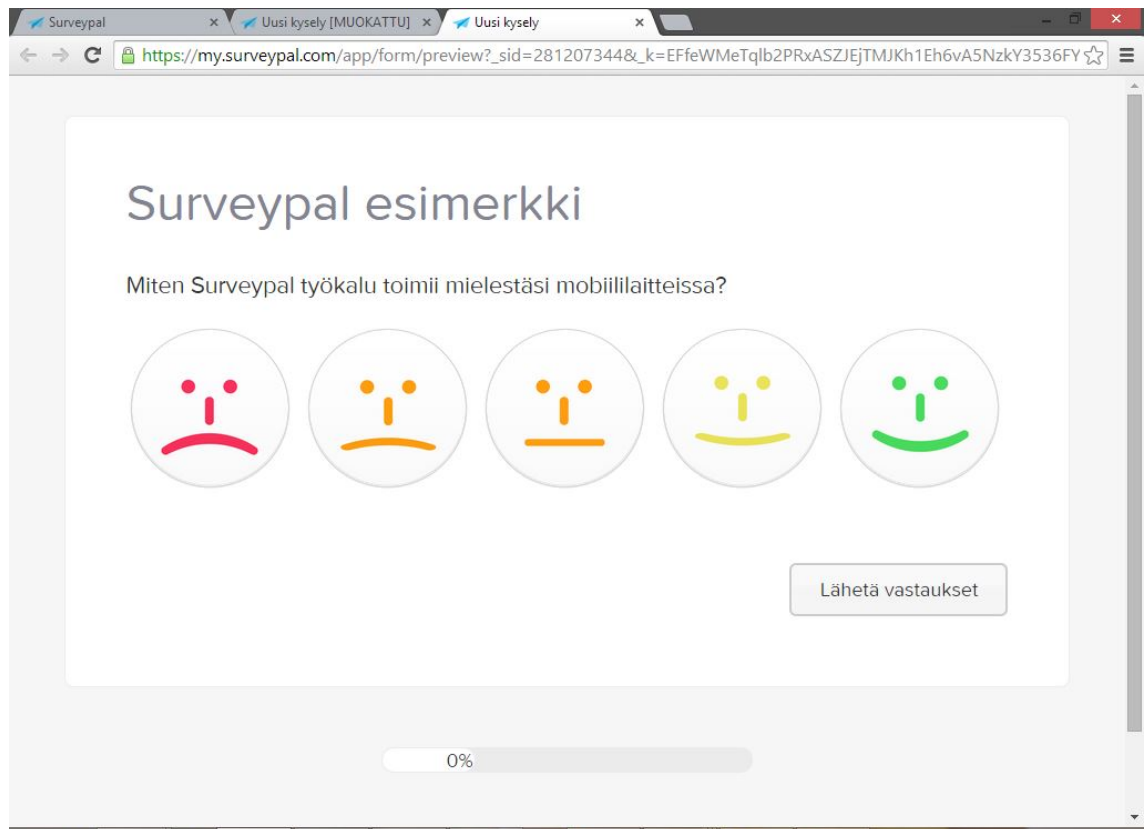




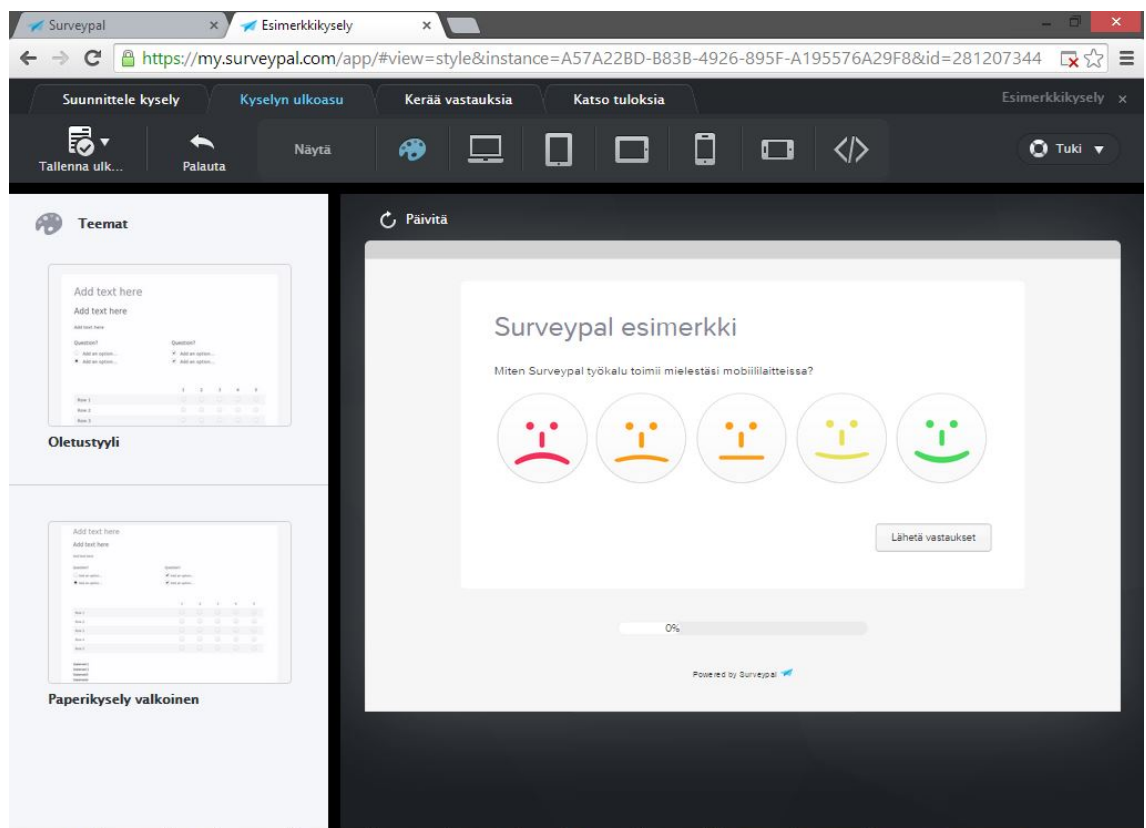
Kuva A.2: Surveypal - aloitusnäkö



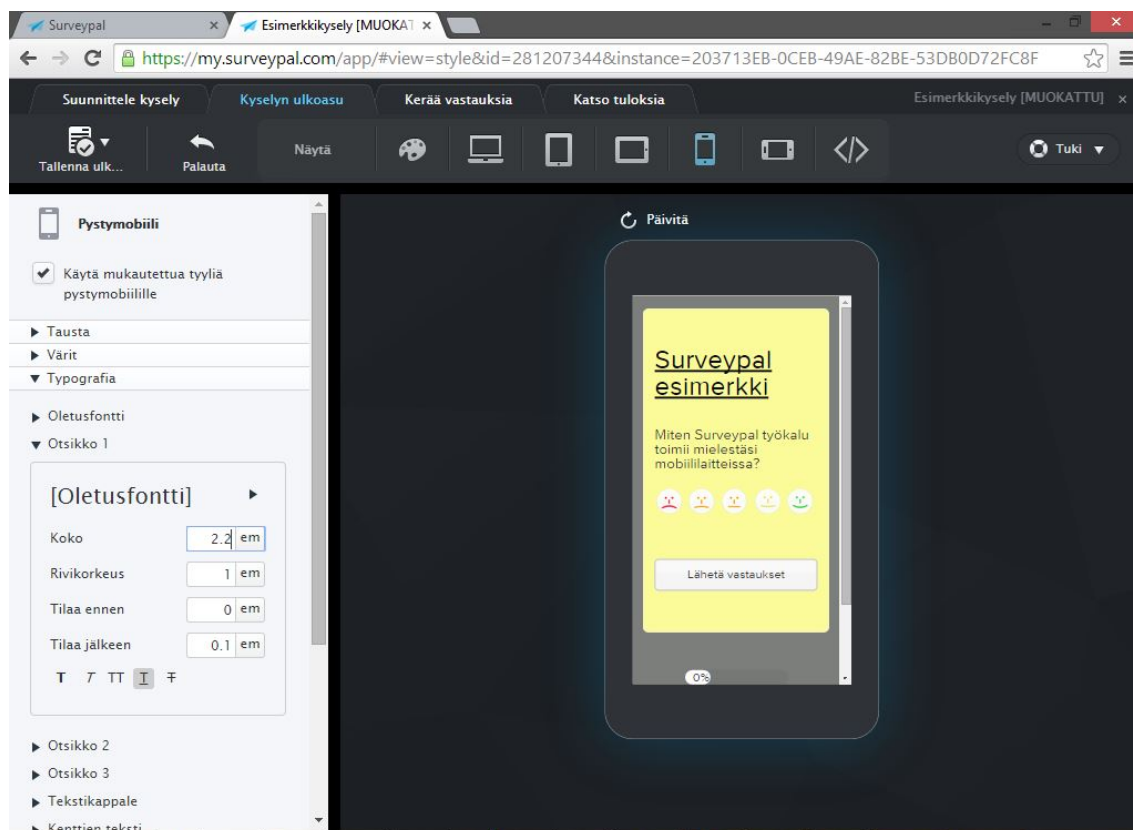
Kuva A.3: Surveypal - kyselyn luominen ja muokkaus



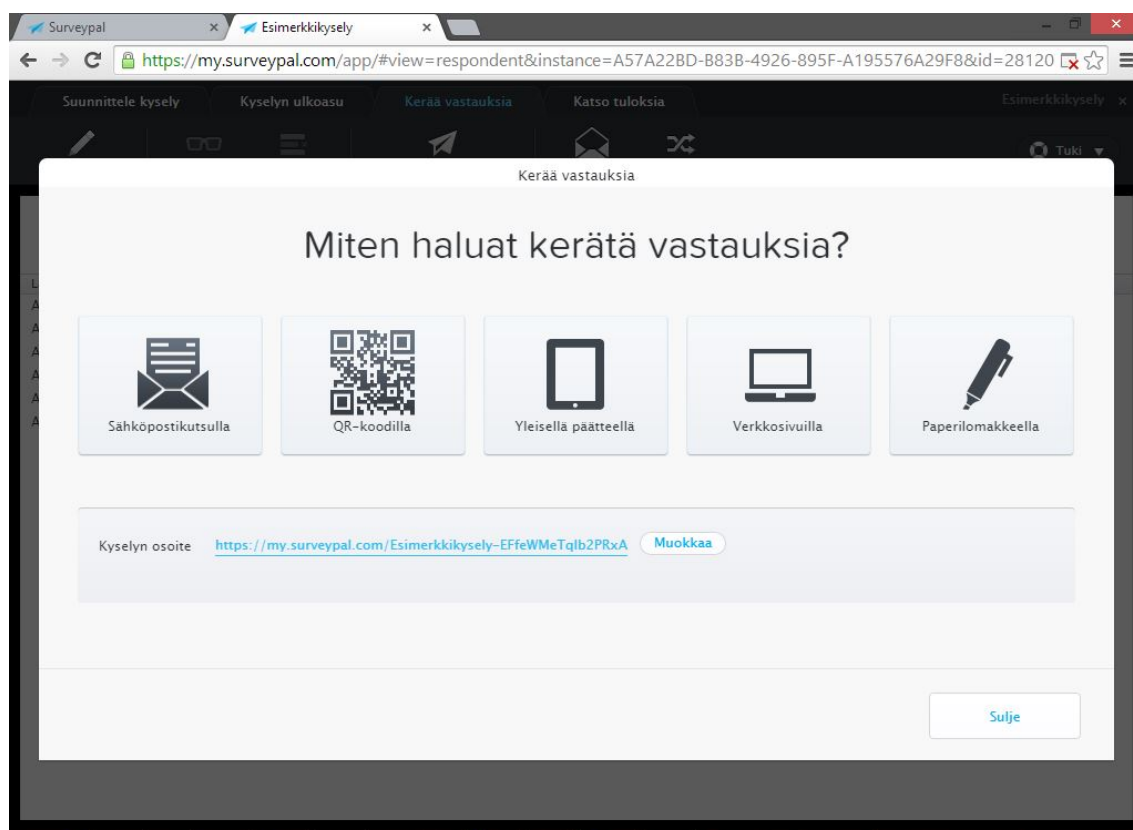
Kuva A.4: Surveypal - kyselyn esikatselu



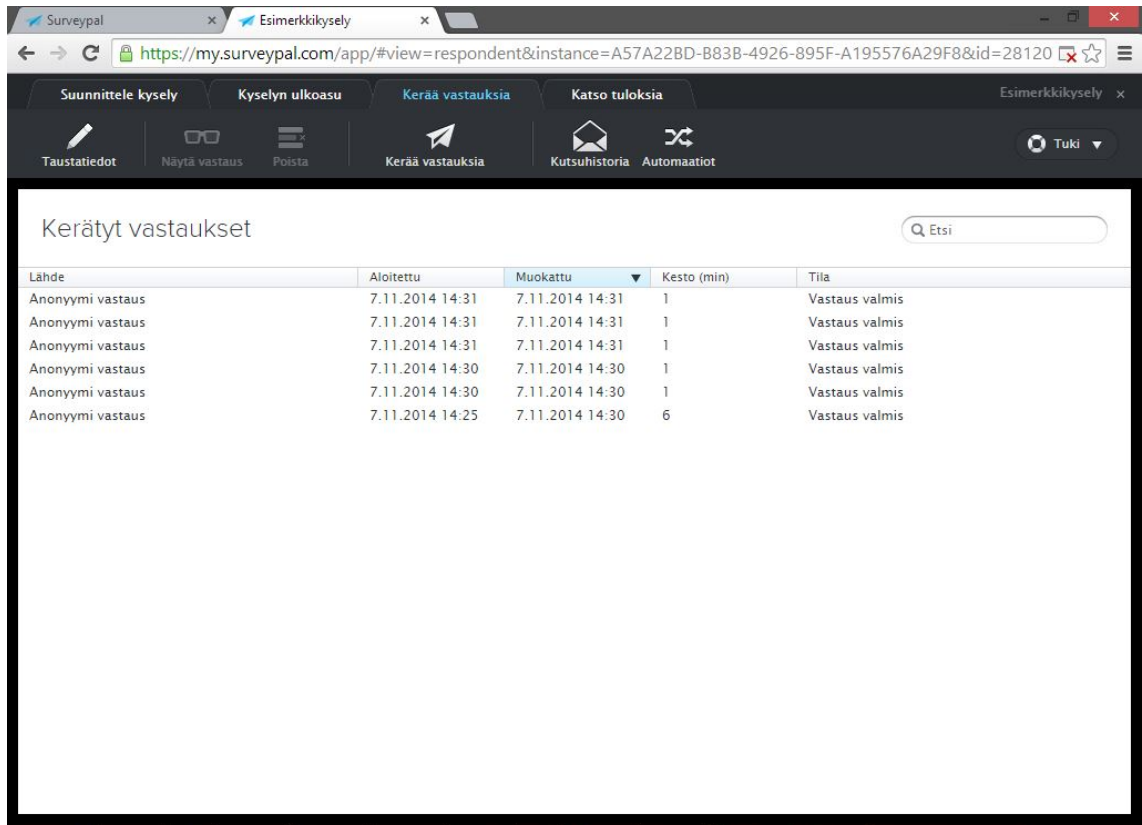
Kuva A.5: Surveypal - kyselyn tema



Kuva A.6: Surveypal - kyselyn mobiili teema

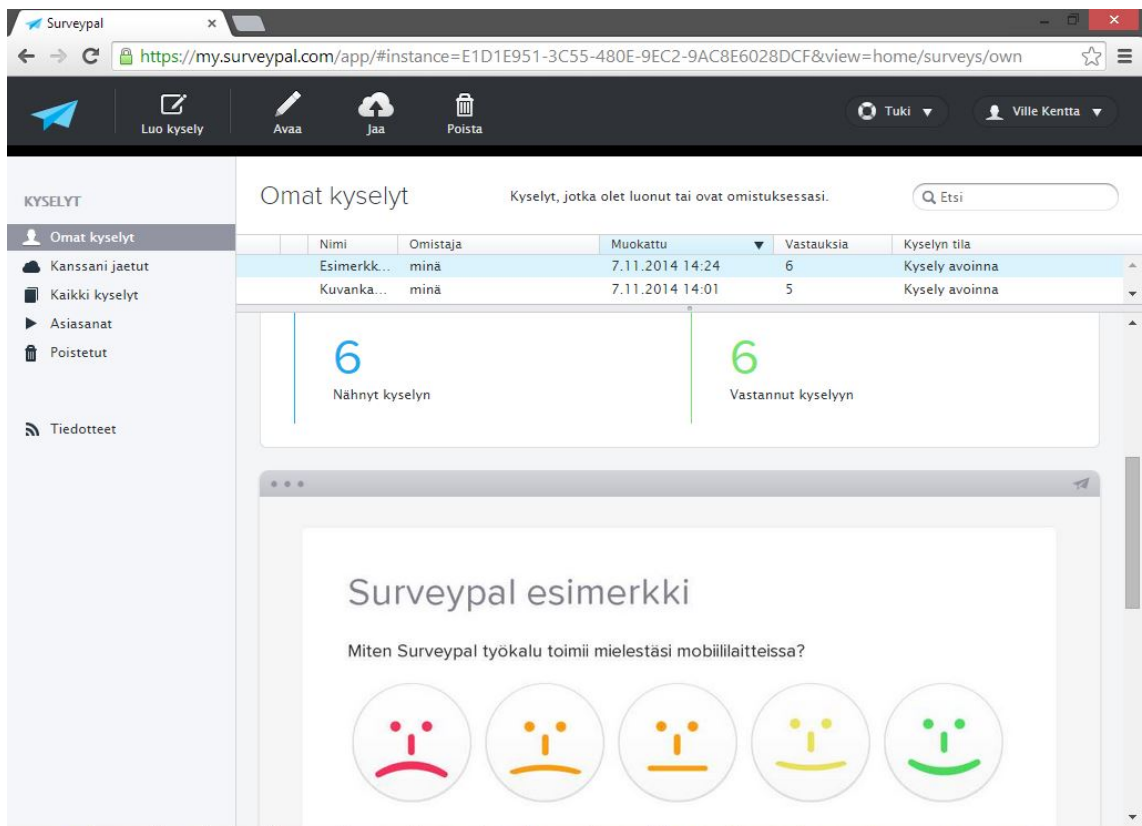


Kuva A.7: Surveypal - vastausten kerääminen



Lähde	Aloitettu	Muokattu	Kesto (min)	Tila
Anonyymi vastaus	7.11.2014 14:31	7.11.2014 14:31	1	Vastaus valmis
Anonyymi vastaus	7.11.2014 14:31	7.11.2014 14:31	1	Vastaus valmis
Anonyymi vastaus	7.11.2014 14:31	7.11.2014 14:31	1	Vastaus valmis
Anonyymi vastaus	7.11.2014 14:30	7.11.2014 14:30	1	Vastaus valmis
Anonyymi vastaus	7.11.2014 14:30	7.11.2014 14:30	1	Vastaus valmis
Anonyymi vastaus	7.11.2014 14:25	7.11.2014 14:30	6	Vastaus valmis

Kuva A.8: Surveypal - kyselyyn saadut vastaukset



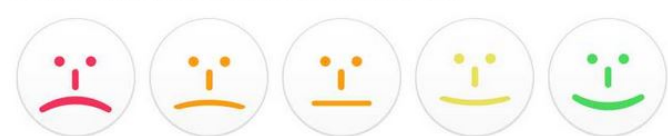
Nimi	Omistaja	Muokattu	Vastauksia	Kyselyn tila
Esimerkk...	minä	7.11.2014 14:24	6	Kysely avoinna
Kuvanka...	minä	7.11.2014 14:01	5	Kysely avoinna

6 Nähty kyselyyn

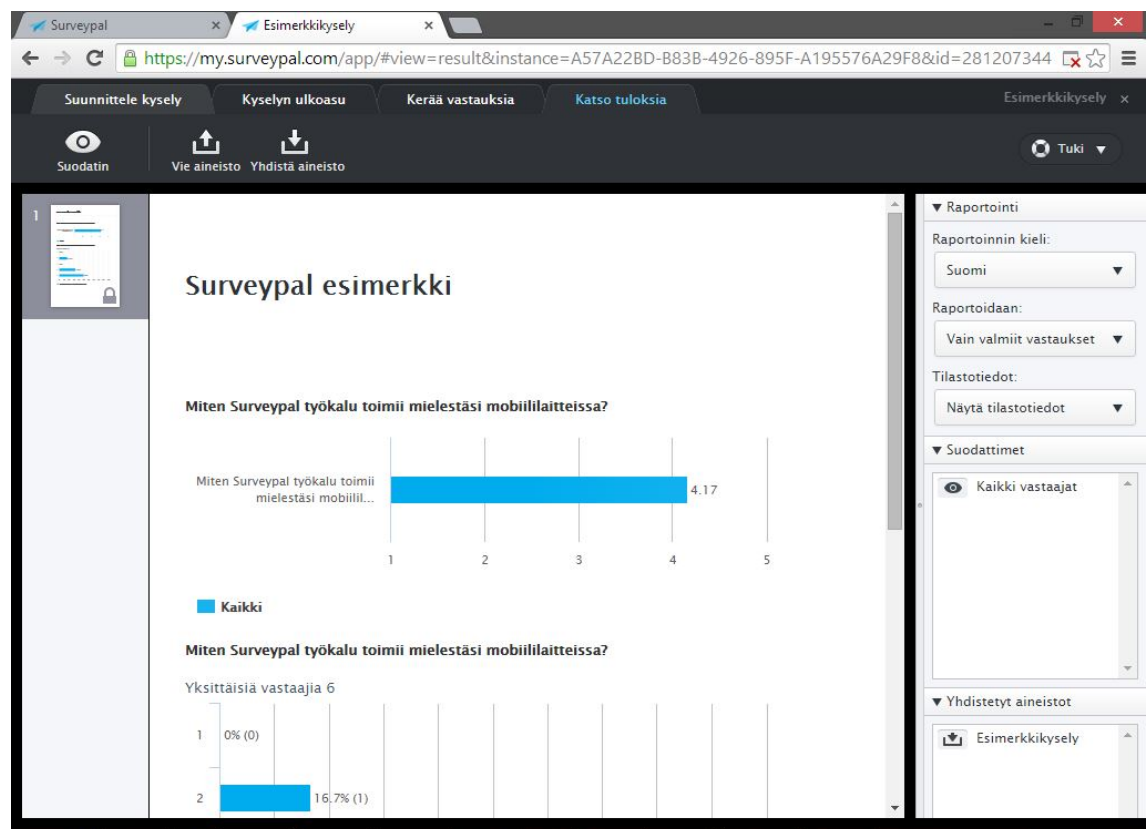
6 Vastannut kyselyyn

SurveyPal esimerkki

Miten SurveyPal työkalu toimii mielestäsi mobiililaitteissa?



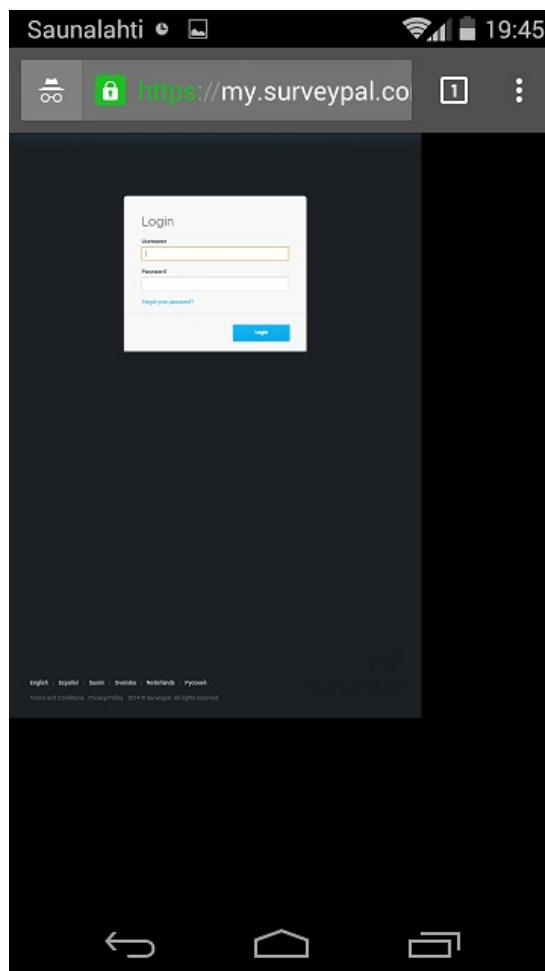
Kuva A.9: Surveypal - kyselyn tulokset aloitusnäkyssä



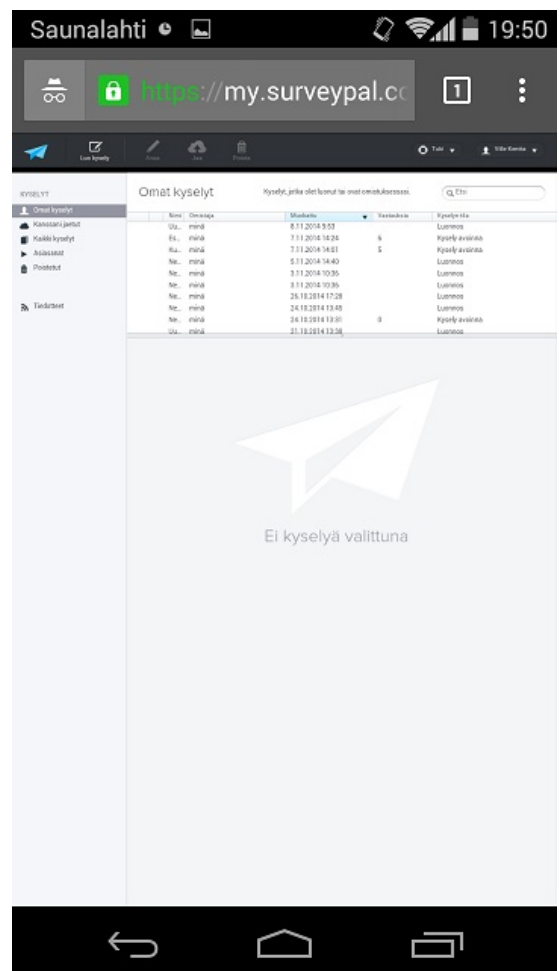
Kuva A.10: Surveypal - kyselyn tulokset

## B. LIITE - SURVEYPAL - KUVANKAAPPAUKSET (ÄLYPUHELIN)

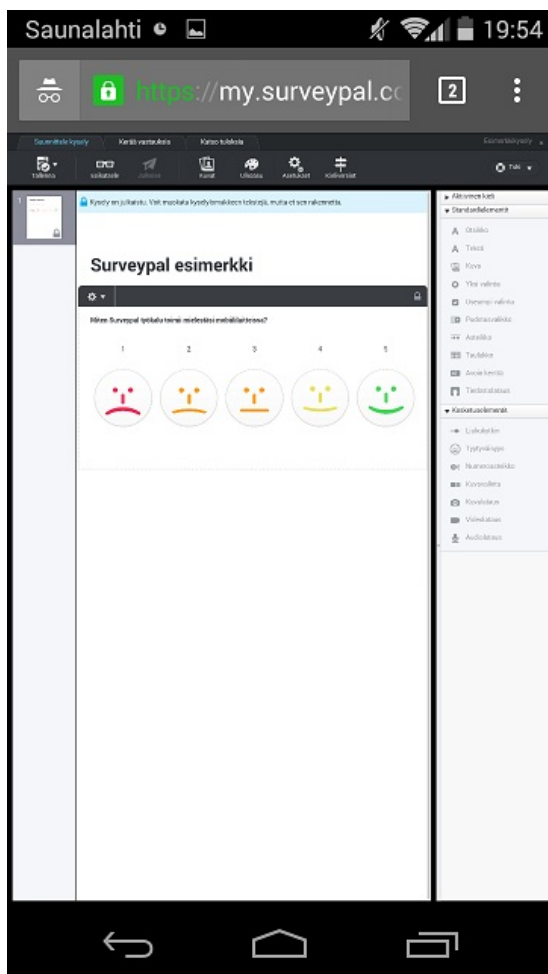
Tässä liitteessä esitellään Surveypal - kyselytyökalun tärkeimmät näkymät älypuhelimella avattuna. Kuvankaappaukset ovat otettu sellaisenaan välittömästi sivunlatauksen jälkeen, zoomaamatta käyttöliittymää manuaalisesti. Kuvankaappauksissa käytettiin *Motorola Moto G* älypuhelinia (*Android 4.4.4*) ja sen vakioselainta, *Google Chrome*:a.



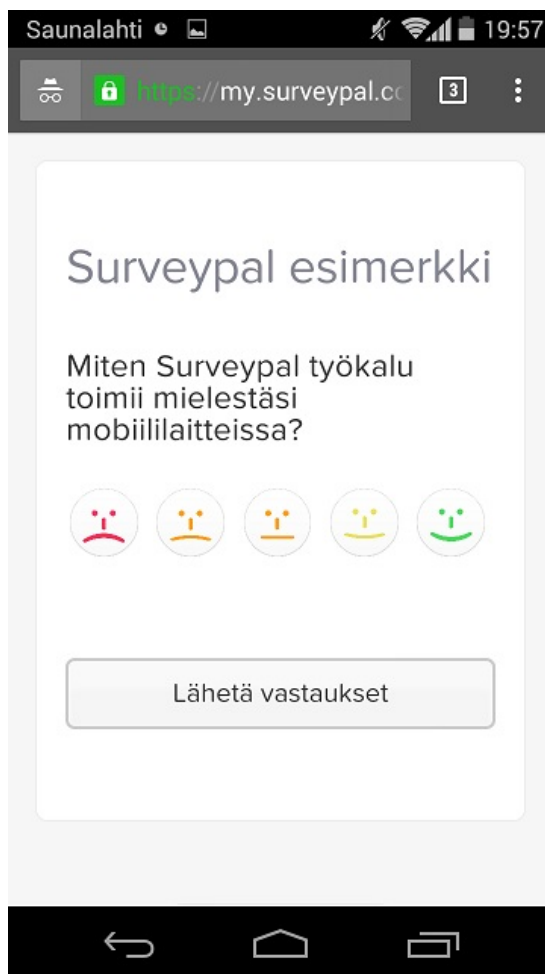
Kuva B.1: Surveypal - kirjautuminen



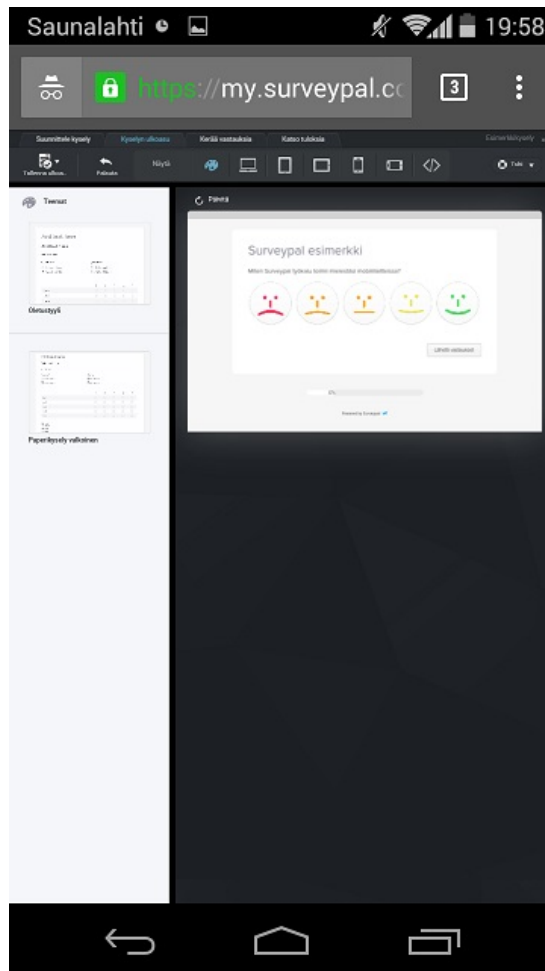
Kuva B.2: Surveypal - aloitusnäkö



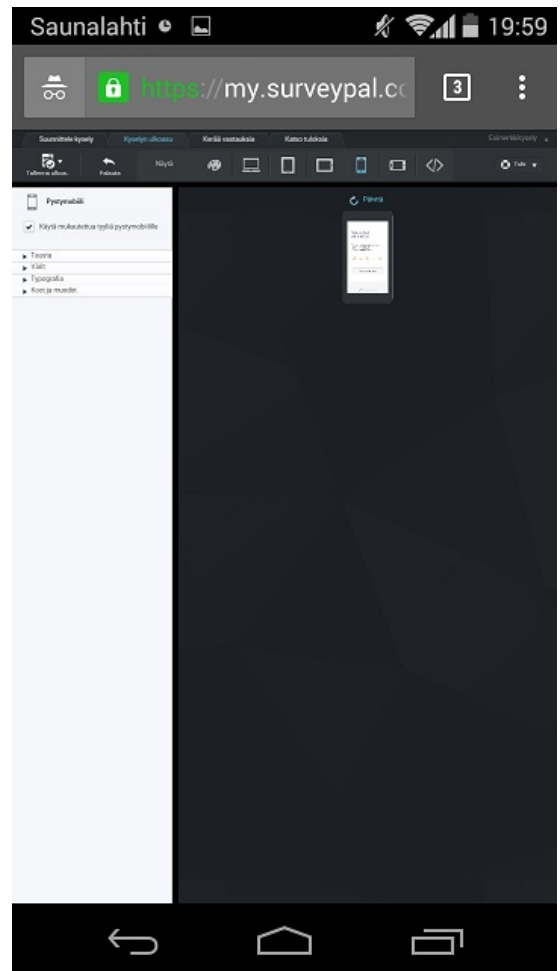
Kuva B.3: Surveypal - kyselyn luominen ja muokkaus



Kuva B.4: Surveypal - kyselyn esikatselu

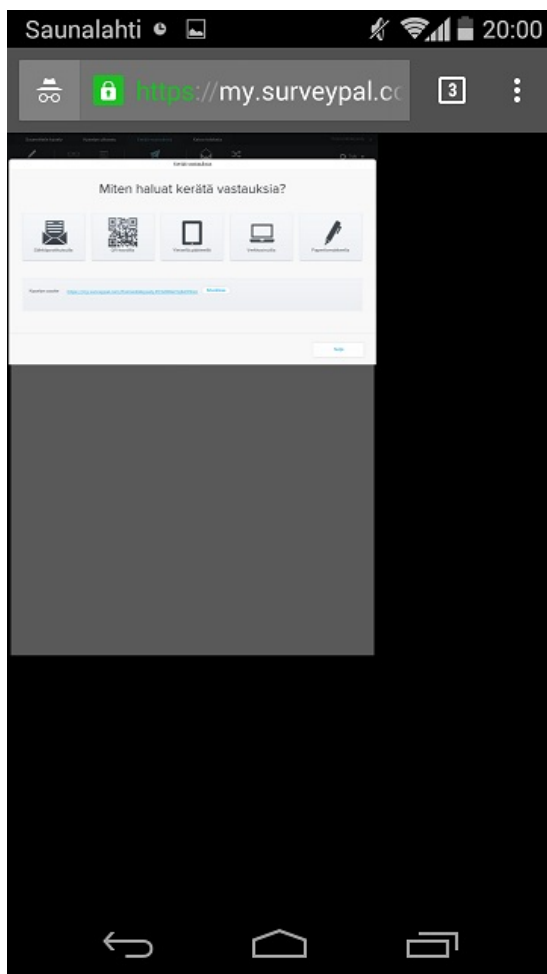


Kuva B.5: Surveypal - kyselyn teema

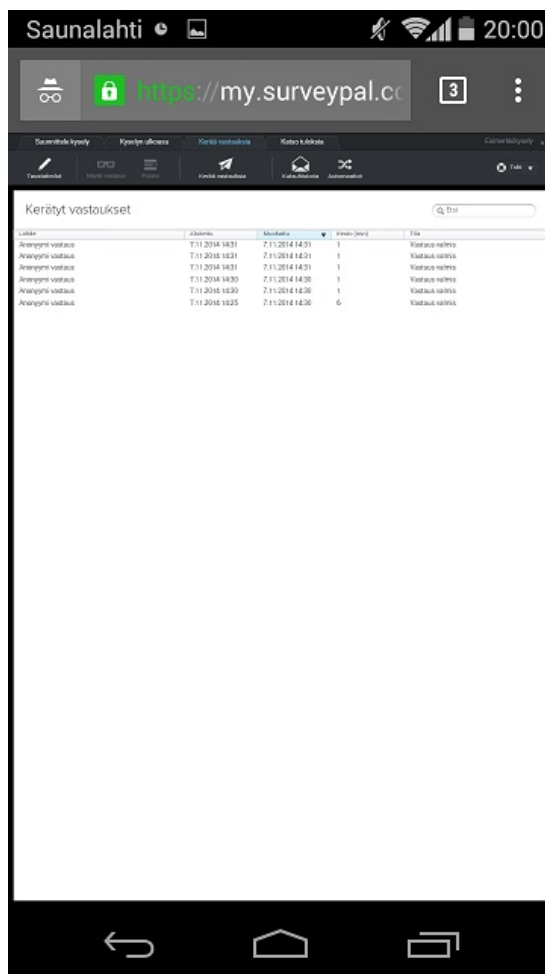


Kuva B.6: Surveypal - kyselyn mobiili teema

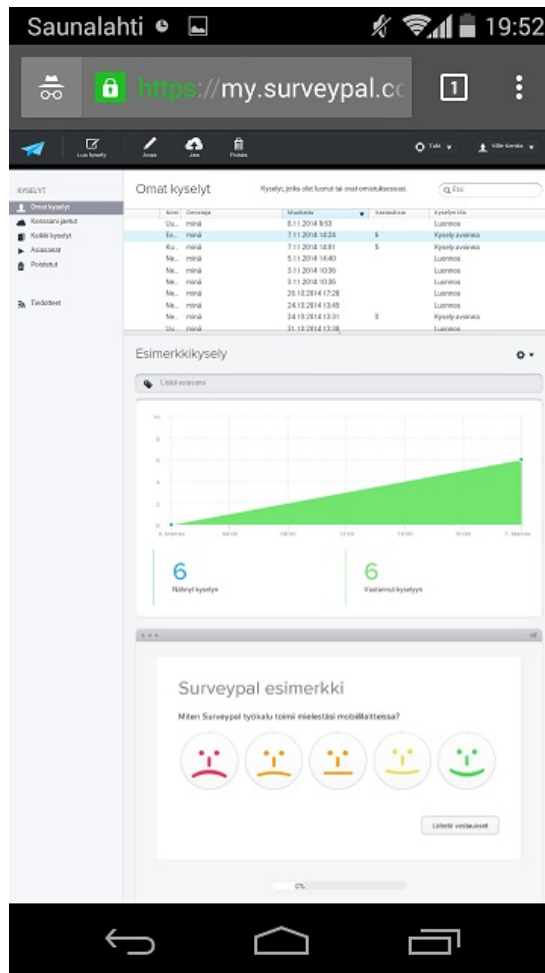




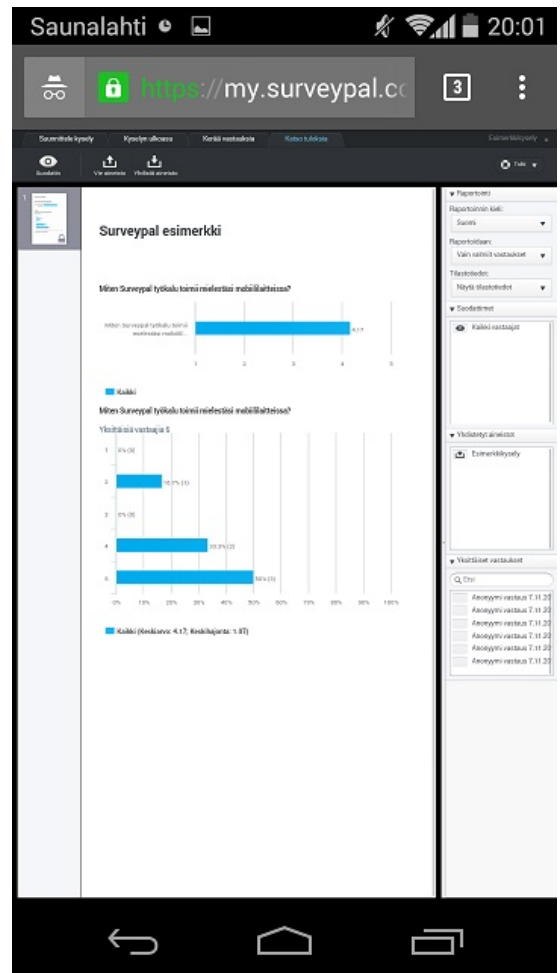
Kuva B.7: Surveypal - vastausten kerääminen



Kuva B.8: Surveypal - kyselyyn saadut vastaukset



Kuva B.9: Surveypal - kyselyn tulokset aloitustähtinä



Kuva B.10: Surveypal - kyselyn tulokset

## C. W3C:N SUOSITUKSET

Tässä liitteessä käydään läpi W3C:n kokoama lista parhaista käytännöistä toteutettaessa mobiilia web-sovellusta. Jokainen suosituksen kohdalla arvioidaan, miten Surveypal-sovelluksen nykytila toteuttaa kyseisen kohdan sekä miten eri mobiilisovelluksen toteutustavat suhtautuvat suosituksen täyttämiseen (taulukko C.0). Erillisten toteutusten tapauksessa oletuksena on, että nykyisen sovelluksen rinnalle tehdään kokonaan uusi erillinen mobiililaitteille tarjottava web-sovellus. Mukautuvan toteutuksen tapauksessa oletuksena on, että nykyistä sovellusta käytetään pohjana mukautuvan sovelluksen luomisessa.

### Esimerkkisuositus

Suositus	W3C:n esittämän suosituksen kuvaus.
<b>Surveypal nykytila</b>	Miten Surveypal-sovelluksen nykytila suhtautuu suositukseen.
<b>Erilliset toteutukset</b>	Miten erillisen mobiilin web-sovelluksen toteuttaminen suhtautuu suosituksen toteutumiseen.
<b>Mukautuva toteutus</b>	Miten nykyisen web-sovelluksen muuttaminen mobiililaitteilla mukautuvaksi suhtautuu suosituksen toteutumiseen.

Taulukko C.0: Tässä liitteessä esiintyvien taulukoiden rakenne

## C.1 Sovellusdata

### Käytä evästeitä säästeliäästi

<b>Suositus</b>	Evästeet ovat suosittu ja toimiva tapa ylläpitää sovelluksen tilaa asiakaspäässä. Koska evästeissä säilötty data lähetetään palvelimelle jokaisen kutsun yhteydessä, tulisi niihin säilöttävän datan määrä pitää minimaalisena hyvän suorituskyvyn ylläpitämiseksi heikommissa (mobiili-)verkoissa.
<b>Surveyspal nykytila</b>	Sovelluksen tilaa asiakaspäässä ei ylläpidetä evästeiden avulla, eikä evästeiden mukana kuljeteta ylimääräistä dataa.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.1: W3C-suositus: Käytä evästeitä säästeliäästi

### Käytä asiakaspään tietovarastoteknologioita lokaalille datalle

<b>Suositus</b>	Asiakaspään tietovarastoteknologioiden käyttö mahdollistaa sovelluksen nopeamman käynnistymisen, kun kaikkea dataa ei jokaisella sivulatauksella tarvitse noutaa palvelimelta. Sovelluksen käyttöliittymän yleistä reagointinopeutta voidaan myös parantaa tallentamalla data ensin lokaalisti ja tekemällä itse palvelinpään tallennus taustalla.
<b>Surveyspal nykytila</b>	Sovellus hyödyntää <i>HTML5:n local storage</i> -teknologiaa, mutta sen käyttöä olisi mahdollista laajentaa erityisesti sessiotietojen säilyttämiseen.
<b>Erilliset toteutukset</b>	Mobiilitoteutus voitaisiin toteuttaa alusta alkaen käyttämään lokaaleja tietovarastoteknologioita paremmin.
<b>Mukautuva toteutus</b>	Lokaalien tietovarastojen laajempi hyödyntäminen nykyisen toteutuksen pohjalta olisi helposti järjestettävissä ja siitä hyödyttäisiin kaikilla laitteilla.

Taulukko C.2: W3C-suositus: Käytä asiakaspään tietovarastoteknologioita lokaalille datalle

### Toista lokaali data

<b>Suositus</b>	Mikäli sovelluksessa käytetään asiakaspään tietovaraas- toteknologioita, on data lähetettävä myös palvelimelle, jotta se olisi käytettävissä myös muilla laitteilla.
<b>Surveyspal nykytila</b>	Asiakaspään tietovarastoissa säilöttävä data lähetetään pääasiassa myös palvelimelle sitä muokattaessa. Joitain käyttöliittymäasetuksia säilötään vain laitekohtaisesti.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytän- töä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.3: W3C-suositus: Toista lokaali data

## C.2 Turvallisuus ja yksityisyys

### Älä suorita suodattamatonta tai epäluotettua *JSON*-dataa

<b>Suositus</b>	<i>JSON</i> -datan jäsentämiseen ei tulisi käyttää <i>JavaSc-</i> <i>ript</i> :in <i>eval()</i> -funktioita, vaikka se saattaakin olla no- pein vaihtoehto rajoittuneilla laitteilla. Puhdistamatto- man ( <i>unescaped</i> ) <i>JSON</i> -datan suorittaminen aiheuttaa huomattavan tietoturvariskin varsinkin mobiililaitteis- sa, joista voidaan saada selville käyttäjän henkilökoh- taisia tietoja kuten yhteystiedot tai nykyinen sijainti.
<b>Surveyspal nykytila</b>	Sovelluksen koodissa ei käytetä <i>eval()</i> -funktioita <i>JSON</i> - datan parsimiseen lainkaan.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytän- töä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

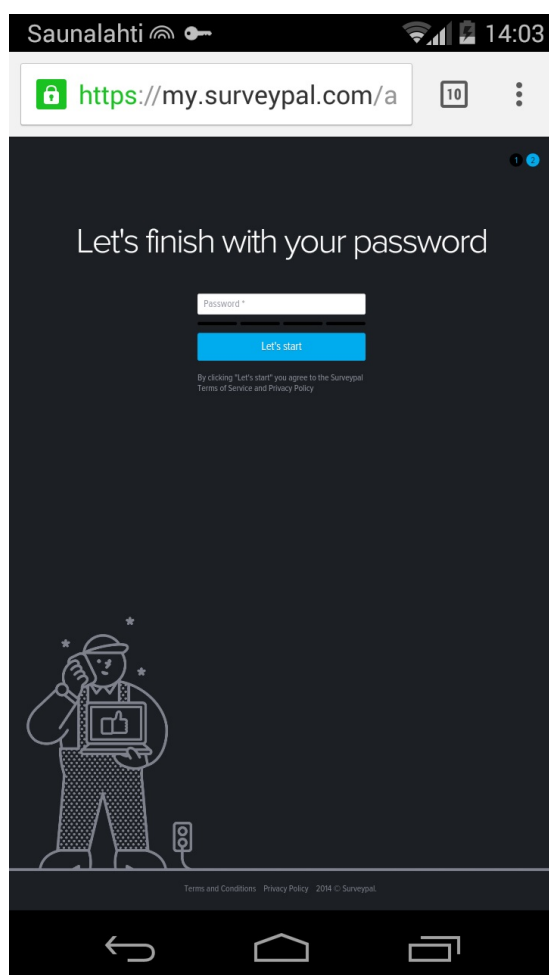
Taulukko C.4: W3C-suositus: Älä suorita suodattamatonta tai epäluotettua *JSON*-dataa

### C.3 Käyttäjän tietoisuus ja hallinta

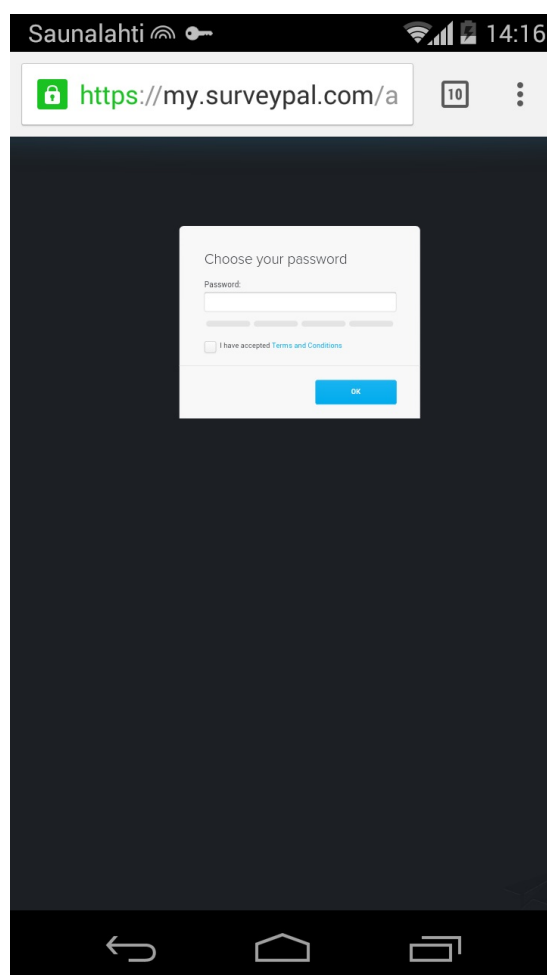
#### Varmista, että käyttäjää informoidaan henkilökohtaisen ja laitteeseen liittyvän informaation käytöstä

<b>Suositus</b>	Mikäli sovellus tarvitsee käyttäjän henkilökohtaista tai laitteeseen liittyvää informaatiota, tulisi sovelluksen ilmoittaa siitä käyttäjälle. Käyttäjää olisi informoitava siitä, millaista dataa sovellus tarvitsee ja siirretäänkö dataa palvelinpäähän. Ilmoitus tulisi antaa joko ennen sovelluksen ensimmäistä käyttökertaa tai ennen informaation ensimmäistä käyttökertaa ja käyttäjällä tulisi olla mahdollisuus hylätä sovelluksen pyyntö.
<b>Surveyspal nykytila</b>	Surveyspal käyttäjien on hyväksyttävä sovelluksen käyttöehdot ja rekisteriseloste ennen kuin he voivat käyttää sovellusta (Kuvat C.1 ja C.2 sivulla 74). Näistä dokumenteista käy ilmi, miten käyttäjään liittyvää informaatiota käytetään. Kyseiset dokumentit sijaitsevat Surveyspal:in kotisivuilla, itse sovelluksen ulkopuolella.
<b>Erilliset toteutukset</b>	Itse informoivan dokumentin rakenne ei vaadi muutoksia erillisten toteutusten tapauksessa. Kuten luvussa 3.1.3 sivulla 14 todettiin, eri laitteiden tunnistaminen vaatii kuitenkin laitteen selaintunnisteen tutkimista palvelinpäässä, mikä saattaa aiheuttaa muutostarpeita dokumentin sisältöön.
<b>Mukautuva toteutus</b>	Informoivaan dokumenttiin ei tarvita muutoksia lainkaan. Laitteen ominaisuudet tunnistetaan asiakaspäässä, eikä siihen liittyvää informaatiota tarvitse siirtää palvelimelle lainkaan.

Taulukko C.5: W3C-suositus: Varmista, että käyttäjää informoidaan henkilökohtaisen ja laitteeseen liittyvän informaation käytöstä



Kuva C.1: Surveypal - Tilin luonti itsepalveluna (käyttäjä hyväksyy käyttöehdot ja rekisteriselosteen viimeistelemällä tilinsä luonnin)



Kuva C.2: Surveypal - CRM:n kautta luodun tilin aktivointi (Käyttäjän on hyväksyttävä käyttöehdot ja rekisteriseloste tilin aktivoinnin yhteydessä)

### Mahdollista automaattinen sisäänkirjautuminen

<b>Suositus</b>	Mikäli sovellus vaatii sisäänkirjautumista (käyttäjätunnus ja salasana), tulisi käyttäjälle tarjota mahdollisuus myös automaattiseen sisäänkirjautumiseen. Tämä on erityisen tärkeää mobiililaitteilla, joilla tekstuaalisen syötteen antaminen on hankalampaa kuin tietokoneen näppäimistöllä.
<b>Surveyspal nykytila</b>	Sovelluksessa ei toistaiseksi ole mahdollisuutta automaattiseen sisäänkirjautumiseen.
<b>Erilliset toteutukset</b>	Erillisten toteutusten tapauksessa automaattinen sisäänkirjautuminen voitaisiin toteuttaa pelkästään mobiililaitteille suunnattuun versioon. Mikäli toiminnallisuus toteutettaisiin myös työpöytäversioon, vaatisi se lisätyötä.
<b>Mukautuva toteutus</b>	Toteuttamalla automaattinen sisäänkirjautuminen, tulee se saataville kaikilla laitteilla ilman erillistä lisätyötä.

Taulukko C.6: W3C-suositus: Mahdollista automaattinen sisäänkirjautuminen

## C.4 Resurssien hillitty käyttö

### Pakkaa siirrettävät resurssit

<b>Suositus</b>	Sovelluksen käyttämien resurssien sisältö tulisi pakata, mahdollisimman pienen verkon käytön mahdollistamiseksi. On suositeltavaa käyttää laajalti tuettua HTTP 1.1 pakkausta.
<b>Surveyspal nykytila</b>	Siirrettävät tekstipohjaiset resurssit ( <i>HTML</i> , <i>CSS</i> , <i>JavaScript</i> ) tiivistetään gzip-paketeiksi. Kuvatiedostoja ( <i>JPG</i> , <i>PNG</i> ...) ei tiivistetä, koska ne eivät hyödy siitä.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.7: W3C-suositus: Pakkaa siirrettävät resurssit



### Minimoi sovelluksen ja datan koko

<b>Suositus</b>	Sovelluksen <i>HTML</i> , <i>CSS</i> ja <i>JavaScript</i> -tiedostot tulisi prosessoida poistaen niistä tyhjä tila (välilyönnit ja rivinvaihdot) sekä lyhentäen ilmaisia (muuttujien, funktioiden ja valitsimien nimet), sikäli kuin se on mahdollista.
<b>Surveyspal nykytila</b>	Kuten luvussa 4.1.2 sivulla 33 totesimme, kaikki sovelluksen koodi tiivistetään jo nykyisessä toteutuksessa.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.8: W3C-suositus: Minimoi sovelluksen ja datan koko

### Vältä uudelleenohjauksia

<b>Suositus</b>	Uudelleenohjauksia tulisi välttää niiden aiheuttaman viiveen johdosta erityisesti mobiiliverkoissa.
<b>Surveyspal nykytila</b>	Sovelluksessa ei ole lainkaan käytössä uudelleenohjauksia. Koska Surveyspal on yhden verkkosivun sovellus (sovelluksen verkko-osoitteen polku on aina sama näkymästä riippumatta), perinteisiltä verkko-osoiteuudelleenohjauksilta vältytään luonnollisesti. Sovelluksen näkymäsiirtymissä käytetään hyväksi verkko-osoitteen osiotunnistetta (#-merkin jälkeinen osa):
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä. Suosituksen täyttäminen on kuitenkin hankalampaa, koska uudelleenohjauksilta ei voida välttyä heikentämättä yleistä käytettävyyttä esimerkiksi tilanteessa, jossa mobiilikäyttäjä avaa tietokoneelle suunnattuun versioon johtavan linkin.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta. Koska kaikki laitteet käyttävät samaa toteutusta, ei uudelleenohjauksia laitekohtaiseen toteutukseen myöskään tarvita.

Taulukko C.9: W3C-suositus: Vältä uudelleenohjauksia

### Optimoi verkkopyynnöt

<b>Suositus</b>	Yhteyden muodostaminen <i>HTTP</i> -pyynnön suorittamiseksi voi kestää huomattavasti kauemman mobiiliverkossa kuin kiinteässä verkossa. Vaikka myös verkkokais-ta on usein mobiiliverkoissa pienempi, on kuitenkin suo-siteltavaa tehdä isompia pyyntöjä harvemmin.
<b>Surveyspal nykytila</b>	Valtaosa sovelluksen tekemistä pyynnöistä on harvoin tehtäviä, isompia pyyntöjä. Joitain pyyntöjä, kuten is-tunnon tietoja kuljettavaa pyyntöä pystyttäisiin opti-moimaan huomattavasti.
<b>Erilliset toteutukset</b>	Verkkopyynnöt voitaisiin optimoida alusta alkaen pa-remmin mobiilitoteutukseen.
<b>Mukautuva toteutus</b>	Verkkopyyntöjen optimoiminen nykyisen toteutuksen pohjalta auttaisi tilannetta kaikilla laitteilla.

Taulukko C.10: W3C-suositus: Optimoi verkkopyynnöt

### Minimoi ulkopuoliset resurssit

<b>Suositus</b>	Kuten edellä todettiin, verkkopyyntöjen määrä tulisi pitää mahdollisimman alhaisena. Ulkoisten resurssien käyttö ei ole suositeltavaa, koska niitä ei voida tiivistää yhteen tiedostoon sovelluksen muiden resurssien kanssa.
<b>Surveyspal nykytila</b>	Sovelluksessa on nykyisellään käytössä kolme ulkopuo-lista resurssia ( <i>JavaScript</i> -tiedostoa): <ul style="list-style-type: none"> <li>• Typekit</li> <li>• Google Analytics</li> <li>• Kissmetrics</li> </ul> Kaikki näistä resursseista ladataan asynkronisesti, ei-vätkä ne ole välttämättömiä sovelluksen toiminnan kan-nalta. Tästä johtuen niiden pitäminen erillään sovelluk-sen omasta koodista on perusteltua, eikä heikennä so-velluksen suorituskykyä.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytän-tää.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.11: W3C-suositus: Minimoi ulkopuoliset resurssit

### Yhdistä staattiset kuvat yhdeksi komposiittiresurssiksi (*Sprite*)

<b>Suositus</b>	<i>HTTP</i> -pyyntöjen minimoimiseksi ja hyvän suorituskyvyn säilyttämiseksi staattisia kuvia (esimerkiksi ikonit) tulisi yhdistää komposiittiresurssiksi.
<b>Surveyspal nykytila</b>	Kaikki sovelluksessa käytettävät ikonit ovat yhdistetty yhdeksi komposiittiresurssiksi. Kaikkia käyttöliittymässä käytettäviä kuvia ei kuitenkaan olla yhdistetty isommiksi kuviksi. Esimerkiksi kaikki kyselypohjien esimerkkikuvat ovat erillisiä ja ne voitaisiin yhdistää yhteen kuvaan.
<b>Erilliset toteutukset</b>	Staattisten kuvien yhdistäminen isommiksi komposiittikuviksi voitaisiin toteuttaa alusta asti paremmin erillisessä mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	Nykyisen toteutuksen kuvien yhdistäminen olisi yksinkertainen prosessi ja se parantaisi tilannetta kaikilla laitteilla.

Taulukko C.12: W3C-suositus: Yhdistä staattiset kuvat yhdeksi komposiittiresurssiksi (*Sprite*)

### Sisällytä taustakuvat CSS-tyylimäärittelyjen sisään

<b>Suositus</b>	Sovelluksen käyttämät taustakuvat tulisi upottaa <i>CSS</i> -tyylimäärittelyiden sisään käyttäen <i>base64</i> -koodausta. Täten taustakuvat eivät vaadi erillistä <i>HTTP</i> -pyyntöä lainkaan.
<b>Surveyspal nykytila</b>	<i>Base64</i> -koodattujen taustakuvien käyttö sovelluksessa on hyvin vähäistä.
<b>Erilliset toteutukset</b>	<i>Base64</i> -koodatut taustakuvat voitaisiin ottaa alusta asti paremmin käyttöön mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	<i>Base64</i> -koodattujen taustakuvien laajempi käyttöönotto olisi helposti toteutettavissa ja se parantaisi tilannetta kaikilla laitteilla.

Taulukko C.13: W3C-suositus: Sisällytä taustakuvat CSS-tyylimäärittelyjen sisään

### Välimuistita resurssit merkitsemällä resurssin viite

<b>Suositus</b>	Toisinaan muuttuvat dynaamiset resurssit voidaan välimuistittaa asettamalla niille URI, joka sisältää sisällön tiivisteeseen ( <i>hash</i> ). Tiivisteeseen johdosta resurssin sisällön muuttuessa myös resurssin URI muuttuu. Täten resurssien välimuistituspolitiikka voidaan asettaa olemaan voimassa ikuisesti ( <i>never expire</i> ).
<b>Surveyspal nykytila</b>	Sovelluksessa ei nykyisellään käytetä resurssien viitteiden merkitsemistä.
<b>Erilliset toteutukset</b>	Tämä käytäntö voitaisiin ottaa käyttöön alusta alkaen erillisessä mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	Käytännön käyttöönotto nykyisessä toteutuksessa olisi vähintään yhtä helppoa kuin erillisten toteutusten tapauksessa ja se parantaisi tilannetta kaikilla laitteilla.

Taulukko C.14: W3C-suositus: Välimuistita resurssit merkitsemällä resurssin viite

### Välimuistita AJAX-data

<b>Suositus</b>	Kuten pääasiallinen sisältö, tulisi myös <i>AJAX</i> -pyynnöillä haettava data välimuistittaa mahdollisuuksien mukaan.
<b>Surveyspal nykytila</b>	Sovelluksen <i>AJAX</i> -pyynnöillä haettavaa dataa (resurssseja) ei välimuistiteta merkitsemällä resurssien viitteitä, mutta ne tallennetaan <i>HTML5</i> :n <i>AppCache</i> :en.
<b>Erilliset toteutukset</b>	Erillisessä mobiilitoteutuksessa <i>AJAX</i> -pyynnöillä haettava data voitaisiin välimuistittaa paremmin alusta lähtien.
<b>Mukautuva toteutus</b>	Nykyinen toteutus ja <i>AppCache</i> :n käyttö tarjoavat hyvän lähtökohdan <i>AJAX</i> -pyynnöillä haettavan datan välimuistituksen suhteen. Kuten taulukossa C.14 todettiin, resurssien merkitsemisen käyttöönotto nykyiseen toteutukseen olisi tehtävissä ja se parantaisi tilannetta entisestään kaikilla laitteilla.

Taulukko C.15: W3C-suositus: Välimuistita AJAX-data

### Älä lähetä evästetietoja tarpeettomasti

<b>Suositus</b>	Staattiset resurssit eivät tarvitse evästetietoja. Tätä johtuen sovelluksen suorituskykyä voidaan parantaa asettamalla nämä resurssit polkuun tai aliverkkotunnukseen, jossa sovelluksen evästeet eivät ole voimassa.
<b>Surveyspal nykytila</b>	Sovelluksen staattiset resurssit palvelevat hajautetun sisällönjakeluverkoston kautta ( <i>CDN</i> ), joten ne ovat lähtökohtaisesti eri verkkotunnuksen alla, missä sovelluksen evästeet eivät ole voimassa.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.16: W3C-suositus: Älä lähetä evästetietoja tarpeettomasti

### Pidä DOM:in koko kohtuullisena

<b>Suositus</b>	Muistissa pidettävän <i>DOM</i> -merkkauksen koko voi olla rajoitettu mobiililaitteilla. Isot ja monimutkaiset sivut voivat ylittää tämän rajan ja aiheuttaa odottamattomia virheitä.
<b>Surveyspal nykytila</b>	Sovelluksen <i>DOM</i> -merkkauksen koko voi olla huomattavan suuri näkymästä riippuen. Esimerkiksi kyselylistaus sovelluksen aloitusnäkyssä voi sisältää käytännössä rajattoman määrän kyselyitä kerrallaan, eikä listan sisältöä ole jaettu sivuille.
<b>Erilliset toteutukset</b>	Tekemällä erillinen mobiilitoteutus, olisi <i>DOM</i> -merkkkaus helpompi suunnitella kokonaan uudelleen mobiilioptimointia ajatellen.
<b>Mukautuva toteutus</b>	Nykyisen toteutuksen <i>DOM</i> -merkkaukseen olisi muokattava paikoittain sen liiallisen paisumisen estämiseksi. Jotta nykyinen toteutus voitaisiin muotoilla mobiililaitteen näytölle sopivaksi, vaaditaan <i>DOM</i> -merkkaukseen myös muita muutoksia. Mukautuvalle verkkosuunnittelulle piirteenoimaista on myös hieman suurempi <i>DOM</i> -merkkauksen koko, kuin mitä on mahdollista saavuttaa erillisillä toteutuksilla.

Taulukko C.17: W3C-suositus: Pidä DOM:in koko kohtuullisena

## C.5 Käyttökokemus

### Optimoi sovelluksen käynnistymisnopeus

---

<b>Suositus</b>	<p>Sovelluksen käynnistymisnopeus vaikuttaa huomattavasti käyttökokemukseen. Seuraavat tekniikat voivat auttaa minimoimaan sovelluksen käynnistymiseen kuluva aikaa:</p> <ul style="list-style-type: none"> <li>• Offline teknologioiden (kuten <i>HTML5:n AppCache:n</i>) käyttäminen</li> <li>• Isojen skriptien pilkkominen osiin ja skriptien lataaminen vasta niitä tarvittaessa</li> <li>• Lokaalien tietovarastoteknologioiden käyttäminen</li> <li>• Lokaaleihin tietovarastoihin tehtävien kyselyjen lukumäärän minimoiminen</li> </ul>
-----------------	--

---

#### Surveypal nykytila

- *AppCache* on käytössä sovelluksessa.
- Isot skriptit pilkotaan osiin ja skriptit ladataan vasta niitä tarvittaessa *RequireJS*-kirjastoa käyttäen
- Sovellus käyttää lokaaleja tietovarastoja
- Sovellus tekee kyselyitä lokaaleihin tietovarastoihin vain käynnistyessään, eikä tehtyjen kyselyiden lukumäärä ole korkea

---

<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
------------------------------	--

---

<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.
---------------------------	---

---

Taulukko C.18: W3C-suositus: Optimoi sovelluksen käynnistymisnopeus

<https://my.surveypal.com/app/#view=home/surveys/own&instance=8DB4T...>

Koodiesimerkki 6: Surveypal - Aloitusnäkyvän *URI* (omat kyselyt)

---

### Minimoi havaittu latenssi

---

- Suositus** Käyttäjän havaitsema latenssi sovelluksessa on merkittävä käyttökokemukseen vaikuttava tekijä. Seuraavia tekniikoita voidaan käyttää käyttäjän havaitseman latenssin vähentämiseksi:
- Inkrementaalisen piirtämisen mahdollistaminen asettamalla sovelluksen *JavaScript*-koodi sivun loppuun ja käyttäjälle hyödyllisen informaation näyttämisen kun sovelluksen sisältöä vielä ladataan
  - Spinnerien ja latauspalkkien käyttäminen verkkoa tai laitteen *API*:a kutsuttaessa auttamaan käyttäjää ymmärtämään, ettei sovellus ole jumittunut
  - Sivun sisällön päivittäminen dynaamisesti sivun uudelleenlatauksen sijaan
  - Todennäköisesti seuraavaksi näytettävien sivujen esilataaminen

---

**Surveyspal nykytila** Sovelluksessa käytetään inkrementaalista piirtämistä ja käyttäjälle näytetään sovelluksen käynnistyessä latauksen edistymistä kuvaava spinneri (kuva C.3), kunnes sovellus on ladattu loppuun. Spinneriä käytetään myös muiden raskaiden tai aikaavievien operaatioiden yhteydessä (esimerkiksi näkymän vaihto tai käyttöliittymään vaikuttavan verkkopyynnön tekeminen). Sivun sisältöä päivitetään ainoastaan dynaamisesti, eikä sivuja uudelleenladata sisällön päivittämisen vuoksi. Kun käyttäjän avaama näkymä ollaan saatu ladattua valmiiksi, alkaa sovellus lataamaan valmiiksi komponentteja, joita tarvitaan kyseisessä kontekstissa todennäköisesti seuraavaksi.

---

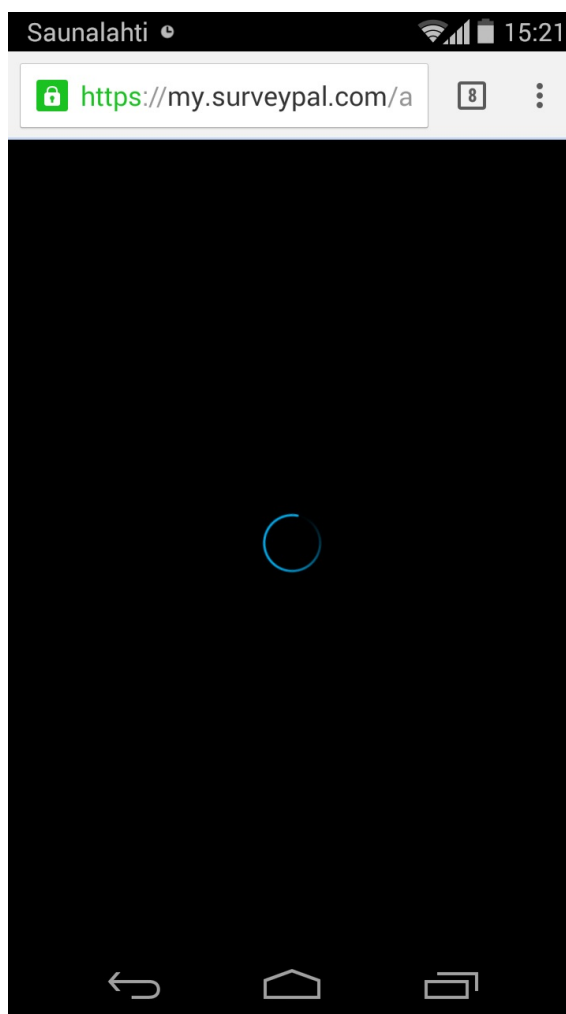
**Erilliset toteutukset** Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.

---

**Mukautuva toteutus** Ei erityistä huomioitavaa tai muutostarvetta.

---

Taulukko C.19: W3C-suositus: Minimoi havaittu latenssi



Kuva C.3: Spinner Surveypal-sovelluksessa



---

### Varaudu erilaisiin interaktiometodeihin

---

**Suositus** Kolme pääasiallista syötetapaa tulisi huomioida käyttöliittymäsuunnittelussa:

- Valintapohjainen: Selaimen valinta *focus* ”hyp-pää” elementistä toiseen
- Osoitinpohjainen: Osoitin voidaan siirtää mihin kohtaan näyttöä tahansa
- Kosketuspohjainen: Tapahtumat (*event*) liittyvät suoraan näyttöön kohdistuneeseen kosketukseen

Optimaalisesti käyttöliittymän tulisi mukautua sen mukaan, mikä syötetapa on käytössä. Mikäli tämä ei ole mahdollista, pitäisi käyttöliittymä suunnitella siten, että kaikkia edellä mainittuja syötetapoja tuetaan.

---

#### Surveypal nykytila

- Valintapohjainen: Kaikki sovelluksen käyttöliittymäelementit eivät ole käytettävissä tätä interaktiometodia hyödyntäen
  - Osoitinpohjainen: Sovellus on optimoitu tälle interaktiometodille
  - Kosketuspohjainen: Tämä interaktiometriodi on katettu osittain
- 

**Erilliset toteutukset** Erillinen mobiilitoteutus voitaisiin toteuttaa alusta alkaen tukemaan paremmin erityisesti kosketuspohjaista interaktiota.

---

**Mukautuva toteutus** Nykyisen toteutuksen interaktiometodien tuen laajentaminen olisi kohtalaisen työläs tehtävä. Ottaen huomioon kosketusnäyttöjen lisääntyneen määrän myös (kannettavissa) tietokoneissa, olisi erityisesti kosketustuki hyödyllinen monilla laitteilla.

---

### Säilytä fokus päivitettäessä sivua dynaamisesti

<b>Suositus</b>	<i>JavaScript</i> :in <i>focus</i> -metodin avulla selaimen fokus voidaan siirtää siihen verkkosivun osaan, joka on päivitetty tai muuttunut. Jos tämä tapahtuu odottamattomasti, voi se kuitenkin hämmentää tai ärsyttää käyttäjää, erityisesti mikäli valinnan palauttamista aiemmin valittuna olleeseen verkkosivun osaan on hankalaa.
<b>Surveyspal nykytila</b>	<i>JavaScript</i> :in <i>focus</i> -metodia käytetään sovelluksessa ainoastaan käyttäjän avaamien ponnahdusikkunoiden yhteydessä, jolloin fokuksen siirtymistä voidaan pitää luonnollisena.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.21: W3C-suositus: Säilytä fokus päivitettäessä sivua dynaamisesti

### Käytä osiotunnisteita (*fragment ID*) ohjaamaan sovelluksen näkymiä

<b>Suositus</b>	Web-sovellus voi vaihtaa näkymää dynaamisesti ilman täyttä sivunlatausta näyttämällä ja piilottamalla sisältöä. Tämä estää oletusarvoisesti kuitenkin edellisiin näkymiin siirtymisen verkkoselaimen ”Siirry taaksepäin” -painikkeella ja tietyn näkymään siirtymisen linkin avulla. Näkymien väliset siirtymät tulisi toteuttaa tarkastelemalla <i>URI</i> :n osiotunnistetta (esimerkiksi: <a href="http://esimerkki.com/sovellus#etusivu">http://esimerkki.com/sovellus#etusivu</a> ) <i>JavaScript</i> :in avulla.
<b>Surveyspal nykytila</b>	Sovelluksen näkymien ohjaamiseen käytetään <i>URI</i> :n osiotunnistetta ja sen <i>view=näkymätunniste</i> -lohkoa (Koodiesimerkki 6 sivulla 81).
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.22: W3C-suositus: Käytä osiotunnisteita (*fragment ID*) ohjaamaan sovelluksen näkymiä

### Tee puhelinnumeroista ”Klikkaa soittaaksesi” -painikkeita

<b>Suositus</b>	Soveltuvia <i>URI</i> -tunnisteita tulisi käyttää puheluiden aloittamisen ja tekstiviestien ( <i>SMS</i> ) lähettämisen helpottamiseksi.
<b>Surveyspal nykytila</b>	Sovelluksissa esiintyvät puhelinnumerot eivät toimi ”Klikkaa soittaaksesi” -painikkeina.
<b>Erilliset toteutukset</b>	Puhelinnumeroiden huomioiminen voitaisiin toteuttaa alusta alkaen paremmin erillisessä mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	Puhelinnumeroiden muuttaminen ”Klikkaa soittaaksesi” -painikkeiksi olisi hyvin helposti toteutettavissa.

Taulukko C.23: W3C-suositus: Tee puhelinnumeroista ”Klikkaa soittaaksesi” -painikkeita

### Varmista tekstikappaleiden sujuvuus

<b>Suositus</b>	Tekstikappaleiden tulisi mahtua näytölle ilman vierityspalkkien tarvetta ja tekstin pitäisi asettua uudelleen mikäli laitteen orientaatio muuttuu.
<b>Surveyspal nykytila</b>	Sovellus on oletettu käytettäväksi pääasiassa tietokoneilla, joiden näyttöjen resoluutio on minimissään 1024 * 768 pikseliä. Tästä johtuen sovelluksen käyttöliittymää joudutaan zoomaamaan ja vierittämään, kuten luvussa 2.2.2 todettiin.
<b>Erilliset toteutukset</b>	Tekstikappaleet, kuten kaikki muukin sisältö, voitaisiin ottaa alusta alkaen paremmin huomioon erillisessä mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	Nykyisen toteutuksen muokkaaminen tältä osin optimaaliseen tilaan olisi huomattavan työmäärän vaativa projekti. Tällä tavoin toteutettuna sovelluksen käyttöliittymä voitaisiin saada toimivaksi hyvin laajalla skaalalla erilaisia laitteita.

Taulukko C.24: W3C-suositus: Varmista tekstikappaleiden sujuvuus

---

**Varmista, että laitteiden välinen tila on yhdenmukainen**

---

<b>Suositus</b>	Kaikki data, joka kerätään tai näytetään toisella laitteella tulisi olla validia ja saatavilla myös toisella laitteella. Esimerkiksi mobiiliversion ja tietokoneversion tarjoama kokemus tulisi olla yhdenmukainen.
<b>Surveypal nykytila</b>	Nykyisellään tietokoneilla ja mobiililaitteilla käytettävä versio on täysin sama ja täten myös luonnollisesti yhdenmukainen.
<b>Erilliset toteutukset</b>	Erillisten toteutusten tapauksessa tämä käytäntö on huomioitava erityisen tarkasti. Koska sovelluksen eri versioiden koodikannat ovat erillään toisistaan, myös eroavaisuuksien muodostuminen on huomattavasti todennäköisempää.
<b>Mukautuva toteutus</b>	Mukautuvassa toteutuksessa yhdenmukaisuuden säilyttäminen on huomattavasti helpompaa, koska sovelluksesta on olemassa vain yksi versio.

---

Taulukko C.25: W3C-suositus: Varmista, että laitteiden välinen tila on yhdenmukainen

### Huomioi web-sovellusten käynnistämiseen soveltuvat mobiililaitteiden tukemat teknologiat

---

<b>Suositus</b>	Verkkolähtöinen sisällön tuominen ( <i>push</i> ) mahdollistaa ilmoitusten ja päivitysten lähettämisen käyttäjille, kun he ovat sovelluksen kontekstin ulkopuolella. Päätelaitteesta riippuen toteutusmahdollisuuksia ovat muunmuassa: <ul style="list-style-type: none"><li>• OMA Push</li><li>• Tekstiviesti</li><li>• QR-koodi</li><li>• Muut valmistajakohtaiset toteutukset</li></ul>
<b>Surveyspal nykytila</b>	Sovellus on nykyisellään suunniteltu käytettäväksi pääasiassa tietokoneella, joten web-sovelluksen käynnistämistä mobiililaitteissa ei olla erikseen huomioitu.
<b>Erilliset toteutukset</b>	Sovelluksen käynnistäminen voitaisiin tehdä helpomaksi alusta alkaen erillisessä mobiilitoteutuksessa.
<b>Mukautuva toteutus</b>	Sovelluksen käynnistämisen helpottaminen mobiililaitteissa olisi yhtä helppo toteuttaa mukautuvaan toteutukseen kuin erilliseen toteutukseenkin. Tässä tapauksessa hyöty olisi kuitenkin suurempi, koska mikä tahansa laite voisi toimia apuna sovelluksen käynnistämiseksi helpommin mobiililaitteessa. Esimerkiksi sovelluksen ollessa auki tietokoneella voitaisiin näyttää <i>QR</i> -koodi, joka voitaisiin lukea mobiililaitteella.

---

Taulukko C.26: W3C-suositus: Huomioi web-sovellusten käynnistämiseen soveltuvat mobiili-spesifit teknologiat

**Käytä *meta viewport* -elementtiä toivotun näyttöresoluution tunnistamiseen**

<b>Suositus</b>	Jotkut verkkoselaimet pyrkivät näyttämään tietokoneille suunnatut verkkosivut pienellä näytöllä zoomaamalla ne automaattisesti. Tämä voi olla kuitenkin ongelmallista, mikäli sovellus on jo optimoitu pientä näyttöä ajatellen. <i>Meta viewport</i> -elementin avulla verkkosivun automaattinen zoomaaminen voidaan estää.
<b>Surveyspal nykytila</b>	Sovelluksessa ei käytetä nykyisellään <i>meta viewport</i> -elementtiä.
<b>Erilliset toteutukset</b>	Erillisessä mobiilitoteutuksessa <i>meta viewport</i> -elementti voitaisiin ottaa alusta alkaen käyttöön ja suunnitella toteutuksen käyttöliittymäasettelu pienille näyttöresoluutioille sopivaksi.
<b>Mukautuva toteutus</b>	Kuten luvussa 3.2.2 totesimme, <i>meta viewport</i> -elementti vaaditaan mukautuvan toteutuksen luomiseksi. Nykyisen toteutuksen ollessa optimoitu tietokoneille, vaaditaan käyttöliittymän asetteluun pienemmillä resoluutioilla muutoksia tämän muutoksen myötä.

Taulukko C.27: W3C-suositus: Käytä *meta viewport* -elementtiä toivotun näyttöresoluution tunnistamiseen

## C.6 Erilaisten toimituskontekstien käsittely

### Suosi palvelinpään tunnistusta niiltä osin kuin se on mahdollista

- Suositus** Laitteesta saatua tietoa tulisi käyttää määrittelemään sen konteksti ja mahdollistamaan vastauksen mukauttamisen. Tavoitteena on olla lähettämättä laitteelle tarpeetonta tai yhteensopimatonta dataa. Seuraavat *HTML*-pyynnön *header*-osion kentät voivat antaa vihjeitä laitteen ominaisuuksista:
- *Accept*: Tämä selaimen hyväksymien *MIME*-tyyppien lista voi auttaa laitteelle lähetettävän sisällön valitsemisessa tai luomisessa. Se ei ole kuitenkaan aina luotettava ja useat selaimet lupaaavat pystyvänsä käsittelemään kaikkia *MIME*-tyyppejä.
  - *User-Agent*: Tätä yleisesti uniikkia tekstitunnistetta voidaan käyttää avaimena laitteen tietojen hakemiseksi laitekuvauslistauksesta, joita käsiteltiin luvussa 3.1.3, sivulla 14.
  - *X-Wap-Profile*: Tätä viittausta laitteen *User Agent Profile*:en voidaan käyttää laitekuvauslistauksia käytettäessä tapauksissa, joissa laitteen profiili ei ole saatavilla, validi tai ajan tasalla.

---

**Surveyspal nykytila** Sovellus ei nykyisellään yritä tunnistaa käyttäjän laitetta lainkaan.

---

**Erilliset toteutukset** Kuten luvussa 3.1.3 sivulla 14 todettiin, on laitteen tunnistaminen tärkeä osa tätä toteutusmallia. Palvelinpään tunnistuksen avulla voidaan päättää, tarjotaanko pyynnön tekevälle laitteelle mobiili- vai työpöytäversio sovelluksesta.

---

**Mukautuva toteutus** Palvelinpään tunnistus taistelee toteutusmallin periaatteita vastaan.

---

Taulukko C.28: W3C-suositus: Suosi palvelinpään tunnistusta niiltä osin kuin se on mahdollista

```
<meta name="viewport" content="width=800">
```

Koodiesimerkki 7: Selainikkunan emuloiman leveyden määrittäminen kiinteään pikseliarvoon

---

### Käytä asiakaspään tunnistusta kun tarpeellista

---

**Suositus** Tilanteissa joissa tiettyjä ominaisuuksia ei voida tunnistaa palvelinpäässä, voidaan ne pyrkiä tunnistamaan asiakaspäässä. Ominaisuuden tunnistamisen jälkeen informaatiota voidaan hyödyntää mukauttamalla sisältöä suoraan tai hakemalla uusi sisältö palvelimelta. Asiakaspään tunnistuksessa auttavat muun muassa:

- *JavaScript*: Skripti tunnistaa laitteen tai selaimen ominaisuuden ja muuttaa sovelluksen sisältöä tai toimintaa havaintojen mukaan.
- *CSS-mediatyypit*: CSS mediatyypit mahdollistavat eri tyylimäärittelyiden asettamisen eri tyyppisille medioille kuten mobiililaitteille ja tulosteelle.
- *CSS-mediakyselyt*: Mediakyselyt mahdollistavat eri tyylimäärittelyiden asettamisen voimaan erilaisille näytön ominaisuuksille kuten sen leveydelle, orientaatiolle tai resoluutiolle.

---

**Surveypal nykytila** Sovelluksessa tehdään jo nykyisellään jonkin verran asiakaspään tunnistusta, pääasiassa CSS mediakyselyiden muodossa.

---

**Erilliset toteutukset** Mikäli erillisissä toteutuksissa joudutaan tekemään paljon asiakaspään tunnistusta, alkaa toteutusmalli menettämään etujaan verrattuna mukautuvaan toteutukseen.

---

**Mukautuva toteutus** Kuten luvussa 3.2.2 sivulla 22 todettiin, perustuu tämä toteutusmalli käytännössä kokonaan asiakaspään tunnistukseen. Sovellus tunnistaa laitteen ominaisuudet ja mukautuu niihin ajon aikana.

---

Taulukko C.29: W3C-suositus: Käytä asiakaspään tunnistusta kun tarpeellista



### Käytä laitteiden luokittelua sisällön mukautumisen yksinkertaistamiseksi

---

<b>Suositus</b>	<p>Laitteita tulisi luokitella yleisien ominaisuuksien mukaan ryhmiä koodikannan helpomman hallittavuuden mahdollistamiseksi.</p> <p>Esimerkki 1: Sovelluksen luokittelu laitteen piirtokäytön ja ohjelmointirajapintojen mukaan</p> <ul style="list-style-type: none"> <li>• <i>Luokka 1</i>: Perus <i>XHTML</i>-tuki, tuki vain hyvin yksinkertaisille skripteille tai ei skriptitukea lainkaan. Ei <i>XHR</i>-tukea.</li> <li>• <i>Luokka 2</i>: Täysi <i>Ajax</i> ja <i>JavaScript</i>-tuki.</li> <li>• <i>Luokka 3</i>: Kehittyneet ohjelmointirajapinnat kuten paikannustiedot tai <i>AppCache</i>.</li> </ul> <p>Esimerkki 2: Sovelluksen luokittelu tuettujen syötetapojen mukaan</p> <ul style="list-style-type: none"> <li>• <i>Luokka 1</i>: Osoitinpohjainen.</li> <li>• <i>Luokka 2</i>: Kosketuspohjainen.</li> </ul>
<b>Surveysal nykytila</b>	Laitteita ei luokitella nykyisessä toteutuksessa lainkaan. Kaikki laitteet käyttävät samaa versiota sovelluksesta.
<b>Erilliset toteutukset</b>	Erillisten toteutusten tapauksessa esimerkki 2:n mukainen jako kosketuspohjaiseen (mobiili) ja osoitinpohjaiseen (nykyinen, tietokoneille suunnattu) toteutukseen olisi jossain määrin loogista. Kuten luvussa 3.2.2 sivulla 22 todettiin, voi laitteiden luokittelu eri kategorioihin olla kuitenkin hyvin hankalaa.
<b>Mukautuva toteutus</b>	Mukautuvan toteutuksen perusideana on tarjota kaikille laitteille sama toteutus, joten laitteiden luokittelu ei ole luonnollista. Tuki eri ominaisuuksille voidaan kuitenkin tarkastaa asiakaspäässä ja sopeutua jokaisen ominaisuuden osalta erikseen, mikä tekee ratkaisusta joustavamman.

---

Taulukko C.30: W3C-suositus: Käytä laitteiden luokittelua sisällön mukautumisen yksinkertaistamiseksi

---

**Tue ilman *JavaScript*:iä toimivaa variaatiota soveltuvin osin**

---

<b>Suositus</b>	Skripteihin ja <i>XHR</i> :ään pohjautuvat sovellukset eivät ole tuettuja kaikissa selaimissa. Mikäli laaja saavutettavuus on tärkeä huomioitava asia, sovelluksesta voidaan tarjota variaatio, joka käyttää synkronisia lomakkeen lähetyksiä ( <i>form post</i> ) <i>XHR</i> :n sijaan.
<b>Surveyspal nykytila</b>	Kuten luvussa 4.1.2 sivulla 31 todettiin, tukeutuu sovellus hyvin voimakkaasti <i>JavaScript</i> -koodiin. Ilman <i>JavaScript</i> :iä toimivaa variaatiota ei ole tarjolla.
<b>Erilliset toteutukset</b>	Erillisestä mobiilitoteutuksesta voitaisiin tehdä myös ilman <i>JavaScript</i> :iä toimiva versio.
<b>Mukautuva toteutus</b>	Sovelluksen nykyisen toteutuksen muokkaaminen toimivaksi ilman <i>JavaScript</i> :iä ei ole realistisesti saavutettavissa.

---

Taulukko C.31: W3C-suositus: Tue ilman *JavaScript*:iä toimivaa variaatiota soveltuvin osin

---

### Tarjota käyttäjälle mahdollisuus valita käyttöliittymien välillä

---

<b>Suositus</b>	Laitetunnistus ei voi koskaan olla täydellinen eikä kaikkia käyttötapauksia voida ennakoida. Tämän vuoksi automaattinen tunnistus ei ole yksistään riittävä mekanismi käytettävän sovellusversion päättämiseksi. Käyttäjälle on tarjottava mahdollisuus vaihtaa automaattisesti valittu sovellusversio toiseen.
<b>Surveyspal nykytila</b>	Sovelluksen käyttöliittymä on aina sama laitteesta riippumatta.
<b>Erilliset toteutukset</b>	Erillisten toteutusten tapauksessa molempiin toteutuksiin olisi lisättävä linkit, joista pääsee siirtymään toisen toteutuksen vastaavaan näkymään. Vaikka sovellusversioiden (käyttöliittymien) välillä pystyttäisiinkin liikkumaan, ei kaksi toteutusta (tietokoneille ja mobiililaitteille optimoitu) välttämättä riitä kattamaan kaikkia laitteita (esimerkiksi tablet-tietokone).
<b>Mukautuva toteutus</b>	Mukautuvassa toteutuksessa ei tehdä varsinaista laitetunnistusta ja käyttäjä saa aina saman version sovelluksesta. Sovellus sopeutuu aina päätelaitteen selainikkunaan, jonka koon määrittäminen on selainvalmistajan vastuulla. Mukautuvan toteutuksen käyttöliittymien välillä voidaan kuitenkin siirtyä koodiesimerkissä 3 sivulla 17 esitellyn <i>viewport meta</i> -elementin avulla. Käyttöliittymää voidaan muuttaa vaihtamalla elementissä määritettyä sivun sisällön leveyttä haluttuun pikselileveyteen (koodiesimerkki 7 sivulla 90) [7, s. 60].

---

Taulukko C.32: W3C-suositus: Tarjota käyttäjälle mahdollisuus valita käyttöliittymien välillä

## C.7 Muuta huomioitavaa

### Harkitse canvas-elementin tai SVG:n käyttöä dynaamiselle grafiikalle

<b>Suositus</b>	Canvas ja <i>SVG</i> tarjoavat vaihtoehtoisen tavan esittää grafiikkaa web-sovelluksessa.
<b>Surveyspal nykytila</b>	Sovelluksessa kyselyn tulostuloksen kuvaajat on toteutettu <i>SVG</i> -grafiikkana ( <i>Highcharts</i> [42] kirjaston avulla). Nämä ovat käytännössä ainoat sovelluksessa esiintyvät dynaamiset kuvat.
<b>Erilliset toteutukset</b>	Mobiilitoteutuksen tulisi myös noudattaa tätä käytäntöä.
<b>Mukautuva toteutus</b>	Ei erityistä huomioitavaa tai muutostarvetta.

Taulukko C.33: W3C-suositus: Harkitse canvas-elementin tai *SVG*:n käyttöä dynaamiselle grafiikalle

### Informoi käyttäjää automaattisesta verkon käytöstä

<b>Suositus</b>	Verkkoliikenne kuluttaa mobiililaitteen akkua ja voi johtaa datansiirtokustannuksiin. Tästä johtuen on tärkeää informoida käyttäjää sovelluksen käyttäessä verkkoyhteyttä. Informointi voidaan toteuttaa esimerkiksi näyttämällä yksinkertaista ikonia ilmoittamaan sovelluksen taustalla tapahtuvasta verkkoliikenteestä.
<b>Surveyspal nykytila</b>	Sovelluksessa ei näytetä mitään ilmoitusta verkon käytöstä.
<b>Erilliset toteutukset</b>	Verkon käytön ilmaiseminen voitaisiin toteuttaa mobiilitoteutukseen ilman tarvetta muuttaa nykyistä (tietokoneelle suunnattua) toteutusta.
<b>Mukautuva toteutus</b>	Mukautuvassa toteutuksessa toteutettu verkon käytön ilmaisin olisi saatavilla kaikilla laitteilla.

Taulukko C.34: W3C-suositus: Informoi käyttäjää automaattisesta verkon käytöstä

**Tarjota riittävät mahdollisuudet hallita automaattista verkon käyttöä**

---

<b>Suositus</b>	Mikäli sovellus tekee automaattisesti pyyntöjä verkon yli, tulisi käyttäjälle tarjota mahdollisuus hallita niitä. Käyttäjälle tulisi tarjota mahdollisuus poistaa automaattisen pyynnöt käytöstä, asettaa niiden aikataulutus tai laukaista ne manuaalisesti.
<b>Surveypal nykytila</b>	Sovellus tekee kohtuullisen paljon automaattisia pyyntöjä verkon yli. Käyttäjä ei voi vaikuttaa pyyntöjen aikataulutukseen tai estää niiden tekemistä automaattisesti.
<b>Erilliset toteutukset</b>	Automaattisten pyyntöjen hallinta voitaisiin toteuttaa pelkästään mobiilitoteutukseen.
<b>Mukautuva toteutus</b>	Mukautuvassa toteutuksessa pyyntöjen automaattinen hallinta olisi saatavilla kaikilla laitteilla.

---

Taulukko C.35: W3C-suositus: Tarjota riittävät mahdollisuudet hallita automaattista verkon käyttöä

## D. LIITE - KYSELY: SURVEYPAL MOBIILI

Tämä liite sisältää Surveypal-sovelluksen käyttäjille esitetyn kyselyn PDF-tulosteena. Kysely toteutettiin Surveypal-sovelluksella ja vastaajille lähetettiin henkilökohtaiset vastauslinnit sähköpostitse. Vastaajat vastasivat kyselyyn sovelluksen web-käyttöliittymän kautta, joka eroaa ulkoasultaan hieman tässä liitteessä olevasta PDF-tulosteesta.

Huom. Käytöllä tarkoitetaan kyselyiden luomista, hallintaa yms. käyttöä - ei kyselyihin vastaamista.

Oletko käyttänyt Surveypal -  
sovellusta jollain mobiililaitteella,  
kuten tablet-tietokoneella tai  
älypuhelimella?

\*

Olen

En ole

Huom. Käytöllä tarkoitetaan kyselyiden luomista, hallintaa yms. käyttöä - ei kyselyihin vastaamista.

Minkä tyyppisellä mobiililaitteella käytit Surveypal-sovellusta?

Tablet-tietokone

Älypuhelin

Jokin muu, mikä

Mikä käyttöjärjestelmä käyttämässäsi mobiililaitteessa oli?

\*

Apple iOS

Google Android

Microsoft Windows (phone)

Jokin muu, mikä

En tiedä

\*Aseta liukukytin mielestäsi sopivimpaan kohtaan

Kokemasi mobiilikäytön sujuvuus  (0 - 10)

Kuvaile lyhyesti Surveypal käyttökokemustasi käyttämälläsi mobiililaitteella.

\*

---

---

---

---

Kuva D.2: Kysely: Surveypal-mobiili, sivu 2 (vastaaja on vastannut myöntävästi ensimmäisen sivun kysymykseen (kuva D.1))



Huom. Käytöllä tarkoitetaan kyselyiden luomista, hallintaa yms. käyttöä - ei kyselyihin vastaamista.

Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal-sovellusta?  
\*

Tablet-tietokoneella

Älypuhelimella

Jokin muu, mikä

Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä.

	Ei lainkaan tärkeä			Todella tärkeä	
	1	2	3	4	5
Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen) *	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Kyselyn luominen ja muokkaaminen *	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Kyselyn julkaiseminen *	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Vastausten kerääminen (kutsujen lähettäminen ja kyselylinkin jakaminen) *	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tulosten katselu *	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Kuvaile lyhyesti, millaisessa tilanteessa käyttäisit Surveypal -sovellusta mobiililaitteella ja mitä tehtävää sillä todennäköisimmin suorittaisit.  
\*

---



---



---



---

Kuva D.3: Kysely: Surveypal-mobiili, sivu 3

Sukupuolesi?

Mies

Nainen

Ikäsi?

Alle 18v

18 - 30v

31 - 45v

46 - 60v

yli 60v

Aseta liukukytin mielestäsi sopivimpaan kohtaan

Arvioi, kuinka suuri osa yleisestä internetiselailustasi tapahtuu mobiililaitteilla  (0 - 100)

Kuva D.4: Kysely: Surveypal-mobiili, sivu 4

## E. LIITE - KYSELYN TULOKSET: SURVEYPAL MOBIILI

Tämä liite sisältää liitteessä D esiteltyyn kyselyyn annetut, loppuun asti täytetyt vastaukset. Avoimet vastaukset eivät sisälly liitteeseen.

### Oletko käyttänyt Surveypal -sovellusta jollain mobiililaitteella, kuten tablet-tietokoneella tai älypuhelimella?

Olen	17,2% (79)
En ole	82,8% (379)

Taulukko E.1: Tulokset: Oletko käyttänyt Surveypal -sovellusta jollain mobiililaitteella, kuten tablet-tietokoneella tai älypuhelimella?

### Minkä tyyppisellä mobiililaitteella käytit Surveypal-sovellusta?

Tablet-tietokone	63,6% (49)
Älypuhelin	61,0% (47)
Jokin muu, mikä	1,3% (1)

Taulukko E.2: Tulokset: Minkä tyyppisellä mobiililaitteella käytit Surveypal-sovellusta?

### Mikä käyttöjärjestelmä käyttämässäsi mobiililaitteessa oli?

Apple iOS	71,8% (56)
Google Android	15,4% (12)
Microsoft Windows (phone)	25,6% (20)
Jokin muu, mikä	2,6% (2)
En tiedä	0,0% (0)

Taulukko E.3: Tulokset: Mikä käyttöjärjestelmä käyttämässäsi mobiililaitteessa oli?

**Kokemasi mobiilikäytön sujuvuus**

0 (Heikko)	2,6% (2)
1	5,1% (4)
2	0,0% (0)
3	6,4% (5)
4	6,4% (5)
5	2,6% (2)
6	14,1% (11)
7	24,4% (19)
8	15,4% (12)
9	17,9% (14)
10 (Erinomainen)	5,1% (4)
(Keskiarvo: 6.54, Keskihajonta: 2.46)	

Taulukko E.4: Tulokset: Kokemasi mobiilikäytön sujuvuus

**Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal-sovellusta?**

Tablet-tietokoneella	78,8% (360)
Älypuhelimella	57,5% (263)
Jokin muu, mikä	4,2% (19)

Taulukko E.5: Tulokset: Millä mobiililaitteella voisit kuvitella käyttäväsi Surveypal-sovellusta?

**Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä**

Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen)

1 (Ei lainkaan tärkeä)	8,5% (39)
2	22,1% (101)
3	23,8% (109)
4	26,2% (120)
5 (Todella tärkeä)	19,4% (89)
(Keskiarvo: 3.26, Keskihajonta: 1.24)	

Taulukko E.6: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyiden hallinta (nimeäminen, jakaminen, poistaminen)

**Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä**

Kyselyn luominen ja muokkaaminen

1 (Ei lainkaan tärkeä)	12,7% (58)
2	26,2% (120)
3	23,6% (108)
4	19,2% (88)
5 (Todella tärkeä)	18,3% (84)
(Keskiarvo: 3.04, Keskihajonta: 1.3)	

Taulukko E.7: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyn luominen ja muokkaaminen

**Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä**

Kyselyn julkaiseminen

1 (Ei lainkaan tärkeä)	7,4% (34)
2	12,2% (56)
3	21,2% (97)
4	34,1% (156)
5 (Todella tärkeä)	25,1% (115)

(Keskiarvo: 3.57, Keskihajonta: 1.2)

Taulukko E.8: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Kyselyn julkaiseminen

**Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä**

Vastausten kerääminen (kutsujen lähettäminen ja kyselylinkin jakaminen)

1 (Ei lainkaan tärkeä)	5,9% (27)
2	10,9% (50)
3	20,1% (92)
4	31,7% (145)
5 (Todella tärkeä)	31,4% (144)

(Keskiarvo: 3.72, Keskihajonta: 1.19)

Taulukko E.9: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Vastausten kerääminen (kutsujen lähettäminen ja kyselylinkin jakaminen)

**Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä**

Tulosten katselu

1 (Ei lainkaan tärkeä)	1,7% (8)
2	4,1% (19)
3	10,0% (46)
4	26,6% (122)
5 (Todella tärkeä)	57,6% (263)

(Keskiarvo: 4.34, Keskihajonta: 0.94)

Taulukko E.10: Tulokset: Arvio, miten tärkeitä Surveypal-sovelluksen ominaisuudet ovat/olisivat mobiilikäytössä - Tulosten katselu

**Sukupuolesi?**

Mies	26,1% (117)
Nainen	73,9% (331)

Taulukko E.11: Tulokset: Sukupuolesi?

**Ikäsi?**

Alle 18v	0,0% (0)
18 - 30v	16,9% (76)
31 - 45v	47,0% (212)
46 - 60v	32,8% (148)
yli 60v	3,3% (15)

Taulukko E.12: Tulokset: Sukupuolesi?

**Arvioi, kuinka suuri osa yleisestä internetselailustasi tapahtuu  
mobiililaitteilla**

0%	2,4% (11)
10%	5,5% (25)
20%	9,5% (43)
30%	11,5% (52)
40%	10,0% (45)
50%	12,2% (55)
60%	10,4% (47)
70%	12,6% (57)
80%	16,0% (72)
90%	9,8% (44)
100%	0,0% (0)

(Keskiarvo: 52.66, Keskihajonta: 25.4)

Taulukko E.13: Tulokset: Arvioi, kuinka suuri osa yleisestä internetselailustasi tapahtuu mobiililaitteilla