Kemi Tornion ammattikorkeakoulu

Mobiiliohjelmointi ja käyttöliittymäsuunnittelu

Petri Säkkinen

Kaupan ja kulttuurin koulutusalan opinnäytetyö
Tietojenkäsittelyn koulutusohjelma
Tradenomi

TORNIO 2012

TIIVISTELMÄ

KEMI-TORNION AMMATTIKORKEAKOULU

Koulutusohjelma:	Tietojenkäsittelyn koulutusohjelma
Opinnäytetyön tekijä:	Säkkinen Petri
Opinnäytetyönnimi:	Mobiiliohjelmointi ja käyttöliittymäsuunnittelu
Sivuja:	36 (3):
Päiväys:	25.11.2012
Opinnäytetyön ohjaaja: Koskenniemi Yrjö	

Tässä opinnäytetyössä selitetään mobiililaitteiden- ja mobiilikäyttöjärjestelmien nykytilannetta ja tulevaisuutta, mobiiliohjelmoinnin tekniikoita, käyttöliittymäsuunnittelun- ja toteutuksen eri vaiheita sovelluskehittäjän näkökulmasta. Työn painopisteenä oli tutkia mobiililaitteiden käyttöliittymien merkitystä.

Tutkimuksessa käytettiin kvalitatiivista ja konstruktiivista tutkimusmenetelmää. Tutkimuksessa toteutettiin kysely, joka oli pohjana esimerkkikäyttöliittymän suunnittelussa. Suunnitteluvaiheessa dokumentointiin käytettiin kuvakaappauksia ja esimerkkikohtia sovelluksen lähdekoodista. Toteutusvaiheessa käytettiin avoimia lähdekoodin ohjelmistoja, kuten Eclipseä ja Gimp-ohjelmistoa. Työvälineinä PC-tietokoneessa olivat Microsoft Office-paketti ja testauslaitteistoina Samsung Galaxy S3 ja Galaxy 3 Android-puhelimet.

Tutkimuksen tuloksena syntyi esimerkkisovellus- ja käyttöliittymä sovelluskehitystyön eri vaiheiden kautta. Kyselytutkimuksen tuloksena saatiin tietoa käyttäjien mielipiteistä käyttöliittymiä kohtaan ja mitä eri asioita käyttäjät käyttöliittymissä arvostavat.

Tutkimukseni tulosten perusteella sovelluskehityksessä käyttöliittymällä on suuri merkitys ja suositeltavaa on tehdä käyttöliittymäprojekti asiakaslähtöisesti.

Asiasanat: mobiiliohjelmointi, käyttöliittymäsuunnittelu, käytettävyys, käyttökokemus, sovelluskehitysalustat,

ABSTRACT

KEMI-TORNIO UNIVERSITY OF APPLIED SCIENCES

Degree programme: Business information technology bachelor's thesis	
Author:	Säkkinen Petri
Thesis title:	Mobile programming and user interface desingning
Pages:	36 (3):
Date:	25.11.2012
Thesis instructor:	Koskenniemi Yrjö

The objective of the thesis is to explain mobile devices and operating systems current situation and the future, mobile programming techniques, user interface designing and implementation of the various stages from the application developer's view. The emphasis was to research the role of user interfaces for touch- screen mobile device users.

The research used both qualitative and constructive research methods. The research conducted a survey, which was the basis for an example of user interface design. The planning documents are made by using screenshots and example source code of the application points. During the implementation phase was used to open-source software such as Eclipse and Gimp. Tools on PC was Microsoft Office-package and test equipment Samsung Galaxy S3 and Galaxy 3-Android phones.

The result of the research was example application and the user interface development through different phases. Survey results provided insight into the opinions of the users and the different things what users will appreciate in the user interfaces.

The research results of application development, user interface is great importance, and it is recommended to make project a customer-oriented.

Keywords: Mobile programming, user interface designing, usability, user experience, application development platforms

SISÄLLYS

TIIVISTELMÄ 3
ABSTRACT 4
SISÄLLYS 5
1 JOHDANTO 7
1.1 Työn lähtökohta, tavoitteet ja rajaukset 7
1.2 Tutkimusmenetelmät 7
1.3 Teoreettinen viitekehys 8
1.4 Tutkimuksessa käytetyt työvälineet 8
1.5 Käsitteet 9
2 MOBIILIOHJELMOINTI 10
2.1 Tekniikat 10
2.2 Mobiilikäyttöjärjestelmät 10
2.2.1 iOS 11
2.2.2 Android 11
2.2.3 MeeGo 12
2.2.4 Sailfish 13
2.2.5 Tizen 13
2.2.6 Symbian 13
2.2.7 Windows Phone 14
2.2.8 Maemo 14
2.3 Kehitysympäristöt 15
2.3.1 Eclipse 15
2.3.2 Visual Studio For Windows Phone 16
2.3.3 Xcode 16
2.3.5 Qt 17
2.5 Ohjelmointikielet mobiiliohjelmoinnissa 18
2.5.1 C++ 18
2.5.2 Java Me 19
2.5.3 Java 19
2.5.4 jQuery 19
2.5.6 JavaScript 19
2.5.7 HTML5 20
2.6 Testausympäristö 20

2.6.1 SDK 20
2.6.2 Emulaattorit 21
2.7 Tietoturva 21
2.8 Sovelluskaupat 21
3 KÄYTTÖLIITTYMÄSUUNNITTELU 25
3.1 Yleistä 25
3.2 Erilaisia käyttöliittymiä 25
3.3 Käytettävyys 26
3.4 Käyttökokemus 27
3.5 Käyttöliittymän elementit 27
3.6 Esimerkkikäyttöliittymän suunnittelun työvaiheet 27
4 KYSELYTUTKIMUS 28
4.1 Tutkimuksen tavoitteet 28
4.2 Tutkimuksen toteutuksen vaiheet 28
4.3 Tutkimuksen tulokset 28
5 POHDINTA 30
LÄHTEET 31
LIITTEET 33
Liite 1. 34
Liite 2. 35
Liite 3. 37

1 JOHDANTO

1.1 Työn lähtökohta, tavoitteet ja rajaukset

Opinnäytetyöni lähtökohtana on tutkia mobiiliohjelmoinnin nykytilaa ja tulevaisuutta, käyttöliittymäsuunnittelua, käytettävyyttä, mobiilikäyttöjärjestelmiä ja ohjelmointikieliä. Työn tavoitteena on toteuttaa raportti mobiiliohjelmoinnin nykytilasta ja erilaisista mobiiliohjelmoinnin tekniikoista ja kehitysympäristöistä. Työn tarkoituksena on toteuttaa kyselytutkimus käyttöliittymistä ja yksinkertainen esimerkkisovellus, joka sisältää käyttöliittymän.

Opinnäytetyötutkimus keskittyy mobiiliohjelmointiin, käyttöliittymäsuunnitteluun ja käytettävyyteen sovelluskehittäjän näkökulmasta. Työssä kerrotaan yleisesti mobiiliohjelmoinnista ja käyttöliittymäsuunnitteluun käytettävistä apuvälineistä ja laitteistoista.

1.2 Tutkimusmenetelmät

Tutkimuksessani käytän konstruktiivista tutkimusmenetelmää, jonka avulla yritän selvittää, miten luoda yksinkertainen, toimiva ohjelma- ja käyttöliittymä, mitkä asiat tulee ottaa huomioon ja mitä eri vaiheita suunnittelu ja toteutus sisältää. Konstruktiivisen tutkimuksen avulla pystyy luomaan käytännöllisesti ja teoreettisesti ratkaisuja erilaisiin merkittäviin ongelmiin oman sovelluksen ja käyttöliittymän kohdalla.

Kvalitatiivisella tutkimusmenetelmällä pyrin saamaan kyselyn avulla käyttäjiltä tietoa siitä, minkälainen on hyvä, toimiva käyttöliittymä ja mitä käyttäjä käyttöliittymältä odottaa.

1.3 Teoreettinen viitekehys

Ohjelmointi ja myös mobiiliohjelmointi on yksinkertaisimmillaan toimintaohjeiden antamista ennalta määrätyn toimenpiteen suorittamista varten. Ohjelmoinnin kaltaista toimintaa esiintyy jokaisen ihmisen arkielämässä lähes päivittäin. Algoritmista esimerkkinä voisi olla se, että annamme jollekulle puhelimessa ajo-ohjeet, joiden avulla hänen tulee päästä perille ennestään vieraaseen paikkaan. Tällöin luomme sarjan ohjeita ja komentoja, jotka ohjaavat toimenpiteen suoritusta. Alkeellista ohjelmointia on tavallaan myös mikroaaltouunin käyttäminen, sillä tällöin uunille annetaan selkeät ohjeet siitä, kuinka kauan ja kuinka suurella teholla sen tulee toimia.
Ohjelmoinnissa on olemassa eri tasoja riippuen siitä, minkälaista työvälinettä tehtävän ratkaisuun käytetään. Pitkälle kehitetyt korkean tason työvälineet mahdollistavat työskentelyn käsitteillä ja ilmaisuilla, jotka parhaimmillaan muistuttavat luonnollisen kielen käyttämiä käsitteitä ja ilmaisuja, kun taas matalan tason työvälineillä työskennellään hyvin yksinkertaisilla ja alkeellisilla käsitteillä ja ilmaisuilla. (Hyvönen & Lappalainen 2010. Hakupäivä 07.11.2012)

Mobiiliohjelmointi terminä tarkoittaa mobiilikäyttöjärjestelmää varten rakennettavaa sovelluksen ohjelmointia. Mobiiliohjelmoinnissa on tärkeää huomioida myös laite, jota varten ohjelma tehdään. Esimerkiksi laitteen suoritin-nopeus, näytön koko ja resoluutio tulee ottaa huomioon suunnitteluvaiheessa. Myöskin natiiveja ohjelmia, jotka tarkoittavat prosessorin- ja sen ohjesarjalle käännettyä koodia ohjelmoitaessa, tulee ottaa rajapinnat huomioon. (What Is Native Code? 2006. Hakupäivä 23.11.2012)

1.4 Tutkimuksessa käytetyt työvälineet

Teknistä laitteistoa edustaa opinnäytetyön tekijän oma pöytätietokone Acer Aspire M3400, sekä testaukseen käytettävät Android-puhelimet Samsung Galaxy S3 ja Samsung Galaxy 3. Käyttöjärjestelmänä on Microsoft Windows 7, sekä Office-paketti, jota käytetään dokumentoinnin eri vaiheissa ja tutkielman luomisessa. Kuvakaappausten muokkaamiseen käytän Gimp-ohjelmaa, joka on vapaan lähdekoodin linsenssillä oleva ohjelmisto. Sovelluskehitykseen käytän Eclipsessä Android SDK:ta ja sen sisältämää emulaattoria.

1.5 Käsitteet

Mobiiliohjelmointi on ohjelmointia, jossa keskitytään luomaan ohjeita mobiilikäyttöjärjestelmälle suorittaa ennalta määritetty toimenpide.

Käyttöliittymäsuunnittelu on käyttöliittymän eli laitteen tai käyttöjärjestelmän, kommunikointiin ihmisen kanssa tarkoitetun osan suunnitteluvaihe. (Rouhiainen 1997. Hakupäivä 16.11.2012)

SDK eli Software development kit on tyypillisesti joukko ohjelmistokehityksen työkaluja, joiden avulla voidaan luoda sovelluksia tietylle ohjelmistopaketille, ohjelmiston puitteille, laitteistoille, tietokonejärjestelmille, video-pelikonsoleille, käyttöjärjestelmille tai vastaavalle alustalle. (Software Developers kit (SDK) 2005. Hakupäivä 22.11.2012)

J2ME eli Java 2 Micro Edition. Javan luokkakirjasto mobiiliohjelmointiin (Oracle 2012. Hakupäivä 22.11.2012)

API (Application Programmer Interface) on ohjelmointialustojen tai ohjelmistojen rajapinta, jonka kautta pääsee ohjelman sisäisiin toimintoihin (3Scale 2012. Hakupäivä 22.11.2012)

GUI (Graphical User Interface) tarkoittaa graafista käyttöliittymää.

OS (Operating System) tarkoittaa mitä tahansa käyttöjärjestelmää.

2 MOBIILIOHJELMOINTI

2.1 Tekniikat

Mobiiliohjelmoinnin tekniikoita on useita. Eri mobiilikäyttöjärjestelmät hyödyntävät suurimmaksi osaksi erilaisia ohjelmointikieliä. Esimerkiksi Androidille ohjelmointikielenä on Java-kieli, mutta on mahdollista myös ohjelmoida C-kielellä. Pääasiassa kuitenkin Android-sovellusten ohjelmointiin käytössä on Java-kieli. Web-tekniikat ovat yleistymässä, mobiililaitteiden määrän kasvaessa.

Matkapuhelimille voi nykyään tehdä ohjelmia käyttämällä web-ohjelmointikieliä, kuten HTML, JavaScript ja CSS. Ne eivät silti toimi selaimen kautta, vaan ne ovat oikeita ohjelmia. Ohjelmia voi myydä ohjelmakaupoissa ja ne voivat käyttää ainakin osaa laitteiden API:sta. Ohjelman tekeminen ja ohjelmoinnin opiskelu voi olla web-tekniikoilla paljon helpompaa ja nopeampaa kuin yleisellä C++-ohjelmointikielellä tai Objective-C:llä. Ohjelmointityökalut ovat vaihtelevassa määrin Cross-Platform ympäristöjä, joilla pystytytään luomaan ohjelma usealle eri rajapinnalle. (Pihlajamäki 2010. Hakupäivä 29.10.2012)

Tehokkaita apuvälineitä ohjelmoinnissa ovat myös Javascript-kirjastot, kuten jQuery Mobile ja SenchaTouch. Palvelinohjelmoinnin puolella kieliä ovat esimerkiksi PHP, NET, Perl, Python ja Ruby. Kirjastot, kuten PhoneGap tai Appcelerator on syytä mainita tehokkaina apuvälineinä. Molemmat kehitysalustat mahdollistavat ohjelmien tuottamisen eri käyttöjärjestelmille. Näissä kehitysalustoissa voidaan hyödyntää helposti useita ohjelmointikieliä, kuten Javascript, HTML5 ja CSS3. (Carmen 2012. Hakupäivä 29.10.2012)

2.2 Mobiilikäyttöjärjestelmät

Varsinaisia älypuhelimia ja tablet-tietokoneita on ollut markkinoilla vasta muutamia vuosia. Tätä ennen kämmentietokoneet ja kommunikaattorit olivat markkinoilla, mutta niiden yleisyys ei ollut sitä luokkaa kuin nykyisten mobiililaitteiden.
Tavalliset käyttöjärjestelmät ovat liian raskaita mobiililaitteiden pienille resursseille. Mobiililaitteissa korostuvat syöttölaitteiden, muistin ja tallennuskapasiteetin rajallisuus.

Pitkä käynnissäoloaika ja muistivuodot eivät saa vaikuttaa laitteen jatkuvan käytön toimintaan. Virrankulutuksen täytyy olla pieni. Moniajossa mobiililaitteissa käynnissä olevat sovellukset eivät saa mitenkään häiritä itse puhelimen toimintaa. (Luoma, Kaunisto 2007. Hakupäivä 29.10.2012)

2.2.1 iOS

iOS, aikaisemmalta nimeltään myös iPhone OS perustuu Applen kehittämään Darwinin ytimeen, johon samaan perustuu myös Mac OS. iOS on kuitenkin erilainen siinä mielessä, että se on suunnitteltu ARM-prosessoreille. Mac OS-ohjelmat eivät kuitenkaan toimi suoraan iPhone laitteella. (Apple 2012. Hakupäivä 5.11.2012).

Kuva 1. iOS käyttöjärjestelmän käyttöliittymä iPhone-laitteella. (Computerworld 2012. Hakupäivä 10.11.2012)

2.2.2 Android

Android perustuu muokattuun Linux-ytimeen. Android käyttöjärjestelmä on rakennettu käyttäen vapaata lähdekoodia. Kehitystyökalut eivät ole kuitenkaan kokonaan lisensoituja Open Source:ksi. Android hyödyntää pääasiassa Java-ohjelmointikieltä. Androidia kehittää Google-hakukoneyhtiö. Androidiin on integroituna useita Google:n sovelluksia, esimerkiksi: Google Play, Google Maps ja Youtube. Tällähetkellä Android on maailman suurin mobiilikäyttöjärjestelmä. Android-käyttöjärjestelmää on käytössä muuallakin, kuin mobiililaitteissa. Esimerkiksi digitaalikameroissa. (Android 2012. Hakupäivä 10.11.2012)

Kuva 2. Android 4.2 versio.(Tumblr 2012. Hakupäivä 10.11.2012).

2.2.3 MeeGo

MeeGo on Linux-pohjainen, Nokian ja Intelin kehittämä käyttöjärjestelmä. Meegon yhdistää Nokian Maemo-alusta ja Intelin kehittämä Moblin. Meegossa on ARM- ja x86-arkkitehtuurituki. Meego hanketta isännöi Linux Foundation (Meego.com. 2012. Hakupäivä 10.11.2012)

Nokian pääjohtaja Stephen Elop ilmoitti vuonna 2011, että Nokian tulevat älypuhelimet pohjautuvat Windows Phone käyttöjärjestelmään, joten Nokian N9 jäi ainoaksi Nokian MeeGo laitteeksi. Nykyisin MeeGosta on kehitteillä Mer-pohjaan perustuva Sailfish, Suomalaisen Jolla-yhtiön projektina. Samsung kehittää myös Intelin kanssa MeeGon pohjalle perustuvan Tizenin.

Kuva 3. Meego harmattan käyttöjärjestelmän käyttöliittymä. (Ess 2012. Hakupäivä 10.11.2012)

2.2.4 Sailfish

Sailfish on Jolla Mobilen kehittämä käyttöjärjestelmä. Sailfish perustuu MeeGon Mer-versiopohjaan, jota Jolla on kehittänyt eteenpäin luoden oman käyttöjärjestelmänsä. Sailfish-käyttöjärjestelmässä Jolla on panostanut moniajoon ja Android-sovellusten tukemiseen. Sailfish julkaistaan Marraskuussa 2012. (Jolla 2012. Hakupäivä 22.11.2012)

2.2.5 Tizen

Tizen on Meegon käyttöjärjestelmäpohjaan perustuva käyttöjärjestelmä. Tizen on Samsungin ja Intelin kehittämä, avoimeen lähdekoodiin ja Linux-kerneliin eli ytimeen perustuva käyttöjärjestelmä. Intelin tarkoituksena on tulevaisuudessa korvata MeeGo Tizenillä. Tizen-projektissa on mainostettu sen ulkopuolisille kehittäjille tarkoitetuista ohjelmistokehyksistä, jotka pohjautuvat HTML5-kieleen. (Tizen.org 2012. Hakupäivä 10.11.2012)

2.2.6 Symbian

Symbian OS perustuu mikrokerneliin, jonka tarkoituksena on ajaa suorittimen ajan jakamista, tiedonvälitystä eri prosessien välillä ja muistin varausta. Symbian käyttöjärjestelmä on kirjoitettu C++-kielellä. Tänäpäivänä kehityksestä vastaa Symbian Foundation. Symbian tukee moniajoa. Symbianille ominaista on sen vähäinen virran käyttö. (Symbian Foundation 2012. Hakupäivä 10.11.2012)

Kuva 4. Symbian Belle käyttöliittymä. (Nokia developer 2012. Hakupäivä 10.11.2012)

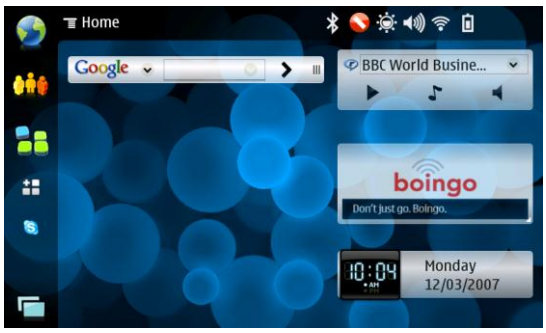
2.2.7 Windows Phone

Windows Phone on Microsoftin kehittämä Windows CE-käyttöjärjestelmään pohjautuva mobiilikäyttöjärjestelmä, joka aiemmin tunnettiin nimellä Windows Mobile. Windows Phonelle on ominaista, että Microsoft vaatii laitevalmistajilta laitteistovaatimuksia. Windows Phonen käyttöliittymä eroaa suurimpien kilpailijoiden käyttöliittymistä laatikkomaisilla LiveTile-tiileillä, jotka päivittyvät aktiivisesti. Käyttöjärjestelmään on integroitu monia Microsoftin palveluita, esimerkiksi Skydrive, Office-paketti ja Xbox-live. Microsoftin Windows Phone ohjelmistokehitysalustana käytetään yleisesti Silverlightia ja Visual studiota. Nokia on tällähetkellä suurin Windows Phone laitevalmistaja. (Windows Phone Suomi 2012. Hakupäivä 10.11.2012)

Kuva 5. Windows Phone 8 käyttöliittymä. (Mshcdn.com 2012. Hakupäivä 10.11.2012)

2.2.8 Maemo

Maemo on 2005 julkaistu, Linux Debianiin perustuva mobiilikäyttöjärjestelmä. Maemo on pääasiassa avoimeen lähdekoodiin perustuva, mutta se sisältää joitain suljettuja koodirivistöjä tai kansioita. Maemolle on ominaista, että siihen pystyy asentamaan Debianiin liittyviä ohjelmistoja ongelmitta. Hyvänä esimerkkinä voidaan pitää Firefox-selainta. (Maemo.org 2012. Hakupäivä 10.11.2012)



Kuva 6. Maemo mobiilikäyttöjärjestelmän käyttöliittymä. (Smabo.de 2012. Hakupäivä 10.11.2012)

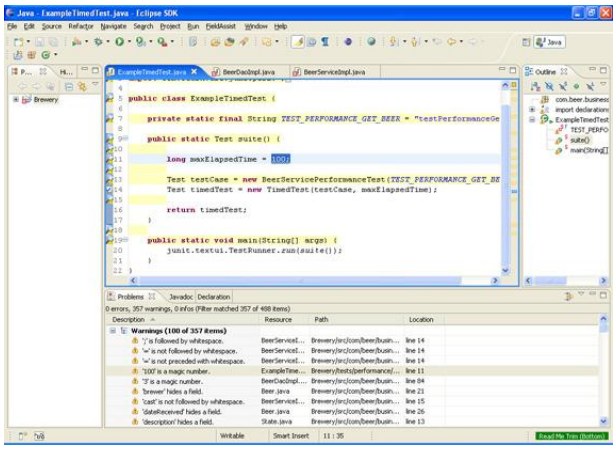
2.3 Kehitysympäristöt

Integroitu kehitysympäristö (IDE) on ohjelmointiympäristö, joka on pakattu sovellusohjelma, jossa on tyypillisesti koodieditorissa kääntäjä, debuggeri ja graafinen käyttöliittymä (GUI). IDE voi olla itsenäinen sovellus tai se voidaan sisällyttää osaksi yhteen tai useampaan yhteensopivaan sovellukseen. Kehitysympäristöt tarjoavat käyttäjäystävälliset puitteet monille nykyaikaisille ohjelmointikielille, kuten Visual Basicille, Javalle ja Power Builderille. Kehitysympäristöissä HTML-kieli on yleisimmin käytetty ohjelmointikieli. Esimerkiksi verkkosivujen ohjelmointiin käytetään kehitysympäristöjä, kuten Dreamweaver:ia tai FrontPage:a. Nykyisin myös HTML5-kieli on yleistymässä, joka tukee laajasti erilaisia mediaelementtejä. (Integrated development environment (IDE) 2007. Hakupäivä 29.10.2012)

2.3.1 Eclipse

Eclipse on sovelluskehitysympäristö (IDE). Eclipse sisältää seuraavia osia: editorin, kääntäjän ja joukon erilaisia käännöksen ja testauksen työkaluja. Eclipseen voi lisätä erilaisia lisäominaisuuksia, kuten versionhallintatyökalun, C++-kehitysympäristön, Cobol-kehitysympäristön, graafiset-työkalut, visuaalisen editorin ja Eclipsen mallinnustyökalut (Benson & Bergius. Hakupäivä 28.10.2012)

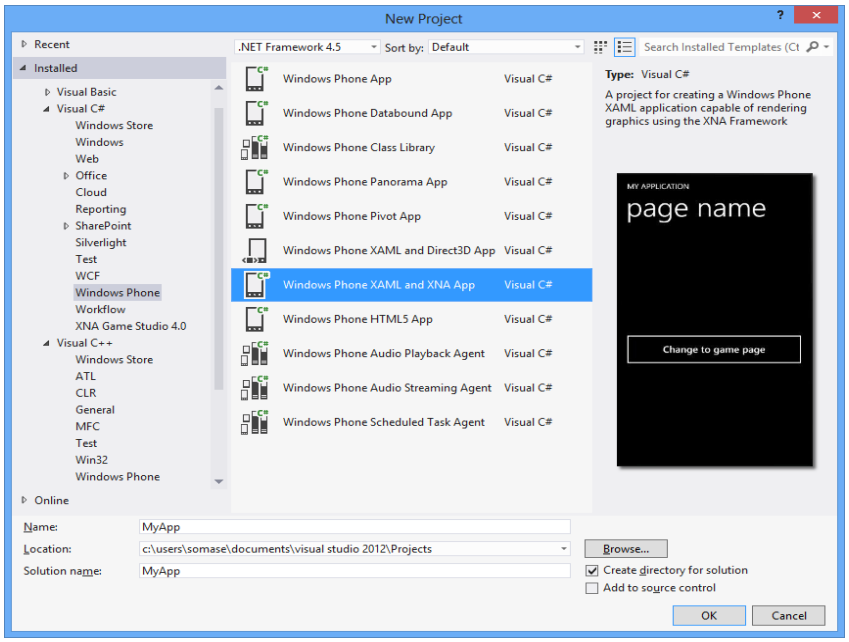
Mobiiliohjelmoinnissa Eclipseä käytetään erityisesti Android-sovellusten toteutukseen. Android SDK:ta voi käyttää Eclipsen lisäksi myöskin Appcelerator-kehitysympäristössä. Eclipsellä voi ohjelmoida myös tavallisia PC-tietokoneille tarkoitettuja ohjelmia.



Kuva 7. Java-ohjelmointia Eclipse ohjelmalla. (IBM 2012. Hakupäivä 13.11.2012)

2.3.2 Visual Studio For Windows Phone

Visual Studio on Microsoftin Windows Phonen kehitysympäristö. Kehitysympäristössä on mahdollista ohjelmoida eri ohjelmointikielillä. Näitä ovat esimerkiksi Visual Basic, C++- ja C#-kielet. Ympäristöllä mobiilisovellusten lisäksi voi luoda myös Web-sovelluksia ja Windows-sovelluksia. Visual Studioon voi integroida myös eri valmistajien täydennyspaketteja. (Visual Studio Express For Windows Phone 2012. Hakupäivä 13.11.2012)



Kuva 8. XAML- ja XNA-ohjelmointia Visual Studio for Windows Phone ohjelmalla. (Livefilestore 2012. Hakupäivä 13.11.2012)

2.3.3 Xcode

Xcode on Apple:n ilmainen ohjelmistokehitysympäristö, jonka avulla voidaan kehittää ohjelmistoja työpöytä-mäkkien lisäksi myös esimerkiksi Applen iOS-laitteille.

Xcode tukee useita erilaisia ohjelmointikieliä. Natiivien OSX- ja iOS-sovelluksien lisäksi työkaluilla on mahdollista kehittää komentoriviohjelmia sekä esimerkiksi Java-sovelmia.

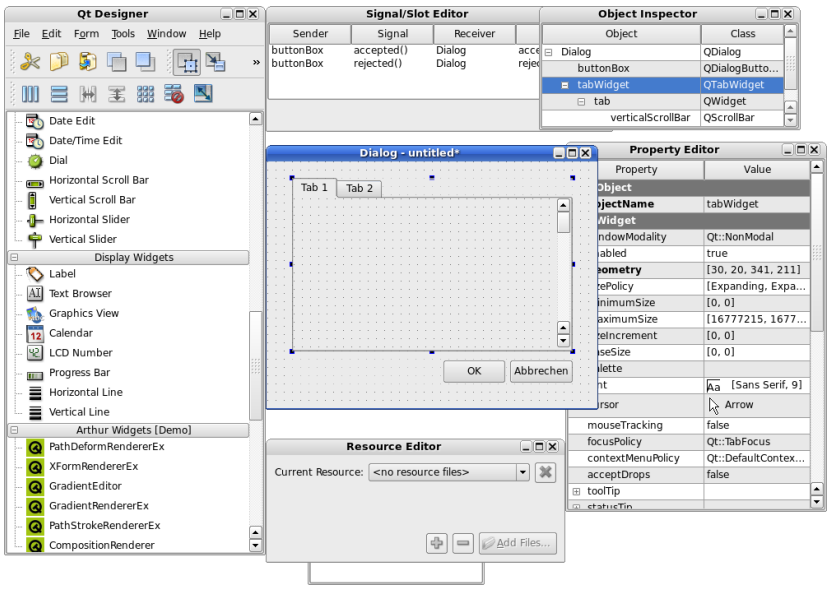
Ohjelman käyttöliittymä on applemaiseen tapaan pc-käyttäjälle vieras ja monipuolinen ohjelmisto vaatii totuttelua kokeneemmaltakin omenafanilta. Ohjelmistokehitykseen Applen alustoilla se on pikkupuutteistaan huolimatta parhaita ratkaisuja. (Pitkänen 2012. Hakupäivä 13.11.2012)

Kuva 9. Xcode kehitysympäristö. (Apple 2012. Hakupäivä 22.11.2012)

2.3.5 Qt

Qt on C++-ohjelmointikielellä toteutettu sovelluskehys, jonka avulla sovelluksia voidaan siirtää helposti useille eri alustoille. Tunnettuja ohjelmaa käyttäviä työpöytäsovelluksia ovat mm. Skype ja Google Earth. Qt on käytössä myös monissa sulautetuissa järjestelmissä ja se toimii MeeGon pääasiallisena rajapintana sovellusohjelmoijlle. Qt:n alkuperäinen kehittäjä ja omistaja oli norjalainen Trolltech. Nokia osti Trolltechin vuonna 2008 ja on jatkanut Qt:n kehittämistä siitä asti. Nokia on muuttanut Qt:n lisensointia sallivammaksi ja kehitysprosessia avoimemmaksi. Qt on saatavissa avoimella LGPL- tai kaupallisella lisenssillä. Toisin kuin monissa muissa moderneissa ohjelmointialustoissa, Qt-sovellusten pääasiallinen ajoympäristö ei perustu virtuaalikoneeseen. Qt-ohjelmat kirjoitetaan C++-ohjelmointikielellä ja käännetään suoraan laitteistolla ajettavaksi binäärikoodiksi.

Qt:n ohjelmointirajapinta eli (API) on hyvin dokumentoitu, selkeä ja yhtenäinen - etenkin ottaen huomioon, että kirjastoa on kehitetty lähes 20 vuotta. Qt:n kehittäjät ovat alusta asti panostaneet kirjaston rajapinnan laatuun ja dokumentoineet rajapintasuunnitteluperiaatteensa. Qt:n pääasiallinen tapa käyttöliittymän tekoon on uusimmissa versioissa Qt Quick. Sen tärkein osa on JavaScriptiin perustuva QML-ohjelmointikieli. Se on dynaamisesti tyypitetty tulkattava kieli, jota ajetaan virtuaalikoneessa natiivin ohjelmakoodin rinnalla. (Saarinen, Järvensivu, Rosendahl, 2011. Hakupäivä 13.11.2012)



Kuva 10. QT sovelluskehitysympäristö.

2.5 Ohjelmointikielet mobiiliohjelmoinnissa

Mobiiliohjelmoinnissa keskitytään eri käyttöjärjestelmille soveltuvien ohjelmointikielien ja API:n käyttöön. Mobiiliohjelmointikieliä on useita. Näistä yleisimpiä ohjelmointikieliä ovat: C++, Java Me, Java, jQuery, jQuery Mobile, JavaScript ja HTML5. Seuraavassa ohjelmointikielistä tarkemmin.

2.5.1 C++

C++ on perinteikäs ohjelmointikieli, jota voi käyttää monilla alustoilla ja lukuisissa käyttöjärjestelmissä. Se on monipuolinen kieli, joka sopii toisaalta matalan tason ohjelmointiin, kuten käyttöjärjestelmien ja laiteajurien kirjoittamiseen ja toisaalta käytännöllisten sovellusten kuten toimisto-ohjelmien ja pelien tekemiseen

C++:lla on maine tehokkaana kielenä. On totta, että C++:lla voi päästä erittäin hyviin tuloksiin, ja suuren käyttäjäkunnan ansiosta sille on myös monta tehokasta lisäkirjastoa. Monet muut kielet ovat erikoistuneet tietynlaisiin sovelluksiin ja ovat siksi omilla aloillaan paljon helppokäyttöisempiä. Esimerkiksi nettisivuja ohjelmoidaan usein PHP:llä ja selainpelejä Javalla ja Flashilla, ja pienten pelien tekoon on erilaisia pelinteko-ohjelmia ja kieliä kuten CoolBasic. Kuitenkin C++ on useiden suurten peli- ja ohjelmistoalan yritysten käytössä, koska se antaa enemmän vapauksia kuin valmiit ratkaisut. (Ohjelmointiputka, 2009. Hakupäivä 16.11.2012)

2.5.2 Java Me

Java Platform Micro Edition (Java ME) tarjoaa vankan, joustavan ympäristön sovellusten ajamiseen mobiililaitteissa, kuten matkapuhelimissa. Java Me on luotu pieniä resursseja omaavia laitteita varten. Java ME:ssä on Javaan verrattuna rajoittuneemmat luokkakirjastot. (Java.com 2012. Hakupäivä 16.11.2012)

2.5.3 Java

Java on Sun Microsystem:in kehittämä järjestelmariippumaton oliopohjainen ohjelmointikieli. Java muistuttaa syntaksiltaan erittäin paljon ohjelmistoteollisuudessa suosittua C++-kieltä, mutta poikkeaa siitä monessa kohtaa hyvin radikaalisti. Järjestelmäriippumattomuus tarkoittaa, että Java-ohjelmia voi kehittää ja testata esimerkiksi Linuxissa, mutta sama koodi toimii myos Windowsissa, Macintoshissa tai Java-verkkotietokoneessa (Network Computer). Java on täysin oliopohjainen ohjelmointikieli. Yksinkertaisinkin tietotyyppi on periaatteessa olio, vaikka rajoituksesta poiketaan tehokkuussyistä lukujen ja merkkien kohdalla. Myös yksinkertaisia tietotyyppejä voidaan käsitellä oliomaisesti ns. Wrapper-luokilla.
Javassa on sisäänrakennettu tuki säikeistä moniajoa varten. Se poikkeaa muista yleisimmistä ohjelmointikielistä, sillä niiden moniajo on yleensä toteutettu käyttöjärjestelmätasolla. Javassa moniajo on toteutettu virtuaalikoneeseen säikeillä, joten moniajo toimii järjestelmäriippumattomasti. (Heinonen 2012. Hakupäivä 16.11.2012)

2.5.4 jQuery

Jquery on ilmainen, avoimen lähdekoodin JavaScript-kirjasto. Jquery on tarkoitettu kaikille selaimille. Jqueryn syntaksi on tehty helpoksi tai helposti ymmärrettäväksi ja tämän takia sen suosio nousee jatkuvasti. Jquery myös toimii hienosti vuorovaikutuksessa AJAX:in kanssa. (jQuery.com 2012, Hakupäivä 16.11.2012)

2.5.6 JavaScript

JavaScript on yksinkertainen, suoraan HTML-koodin sekaan kirjoitettava skriptikieli. Se mahdollistaa interaktiivisten toimintojen lisäämisen WWW-sivuille. Esimerkkeinä mainittakoon vaikkapa lomakkeen tietojen tarkistaminen ja erilaiset valikko-ohjatut WWW-sivuratkaisut. Vaikka toiminnallisesti se onkin ratkaisevasti erilainen kuin Java,

on syntaksi kuitenkin hyvin pitkälle samanlaista. Toisin kuin Javaa, JavaScript:iä ei kuitenkaan tarvitse kääntää minkäänlaiseksi tavukoodiksi ennen ajamista, vaan ohjelma kirjoitetaan todellakin suoraan WWW-sivulle. Näinollen siis jokaisella NetScape- tai IE-selaimen omistajalla on mahdollisuus tehdä JavaScript-ohjelmia sivulleen, ilman erillisiä kehitysohjelmistoja. (Särkkä 1998. Hakupäivä 16.11.2012)

2.5.7 HTML5

HTML5 on uusi standardi HTML-ohjelmointikielelle. Edellisen version HTML, HTML 4.01, tuli vuonna 1999. Internet on muuttunut paljon sen jälkeen. HTML5 on vielä keskeneräinen. Kuitenkin suuret selaimet tukevat monia uusia HTML5 elementtejä ja API:a eli ohjelmointikielen rajapintaa. HTML5:n standardi tukee hyvin mediaelementtejä, esimerkiksi kuvaa ja ääntä. HTML5 on yhteensopiva monien alustojen kanssa, mikä tekee siitä monikäyttöisen ja tehokkaan ohjelmointikielen. (W3schools.com 2012. Hakupäivä 22.11.2012)

2.6 Testausympäristö

Mobiilisovelluksen testaaminen tapahtuu käytännössä Alfa-vaiheessa emulaattorin avulla kehitysympäristössä. Sovelluksen myöhemmässä vaiheessa voidaan Beta-testauksessa testata sovellusta mobiililaitteella, kuten älypuhelimella tai tablet-tietokoneella. Beta-testaukseen minulla oli käytössä Samsung Galaxy S3-älypuhelin, jonka siirsin eli exporttasin apk.-tiedostona Eclipsestä. Suurimmaksi osaksi ohjelman käyttöliittymän testaus tapahtui emulaattorilla. Testausmenetelmänä oli lasilaatikkotestaus.

2.6.1 SDK

Software Development Kit on tyypillisesti joukko ohjelmistokehityksen työkaluja, joiden avulla voidaan luoda sovelluksia tietylle ohjelmistopaketille, ohjelmiston puitteille, laitteistoille, tietokonejärjestelmille, video-pelikonsoleille, käyttöjärjestelmille tai muille vastaaville alustoille. (Rouse, Software developers kit 2005. Hakupäivä 22.11.2012)

2.6.2 Emulaattorit

Emulaattorit ovat ohjelmointikehitysympäristöön integroituja ohjelmia tai laitteistolaajennoksia, joilla on mahdollista käyttää laitteita virtuaalisesti muunlaisilla tietokoneilla ja käyttöjärjestelmillä kuin mihin ne on alun perin tarkoitettu. Esimerkiksi Android-sovellusten rakentamisessa voi käyttää Eclipse-kehitysmpäristöä, johon asennettava Android-SDK sisältää Android-käyttöjärjestelmän toiminnallisuuksia hahmottelevan virtuaalisen laitteen eli emulaattorin ohjelman testausta varten.

2.7 Tietoturva

Useat mobiililaitteet ovat tietoturvan kannalta ongelmallisia. Mobiililaitteet käyttävät erilaisia tekniikoita, joilla tiedonsiirtoa voidaan tehdä. Älypuhelimet ja tabletit ovat jo lähellä tietokoneiden tasoa, joten laitteiston suojauksesta on pidettävä huolta. Toinen ongelma voi olla laitteen katoaminen tai muistikortin tuhoutuminen. Myös virukset ovat lisääntyneet hurjasti eri mobiilikäyttöjärjestelmillä. Esimerkiksi sovelluskaupoista voi löytyä paljon ohjelmia, jotka sisältää haittaohjelmia tai viruksia. Mobiililaitteen suojaukseen on muutamia keinoja. Puhelin kannattaa lukita tunnuskoodilla, salasanalla tai lukituskuviolla. Puhelimeen, viesteihin tai sähköpostiin ei tulisi tallentaa henkilötietoja. Pankkikortin tunnuslukua, tilien salasanoja tai suojauskoodeja yhteystietoihin ilmeisillä nimillä ei tule käyttää. Salanimeä tai nimimerkkiä voi käyttää tarvittaessa. On hyvä tarkistaa sovelluksen käyttöluvat ennen sen lataamista. Tiedostojen ja sovellusten lataaminen kannattaa tehdä vasta, kun on tarkistanut, että niiden lähde on turvallinen. Vanhan puhelimen vaihtamistilanteessa kannattaa palauttaa tehdasasetukset. Suositeltavaa on sallia vain luotettavien sovellusten automaattiset päivitykset. (Mobile Security 2012. Hakupäivä 22.11.2012)

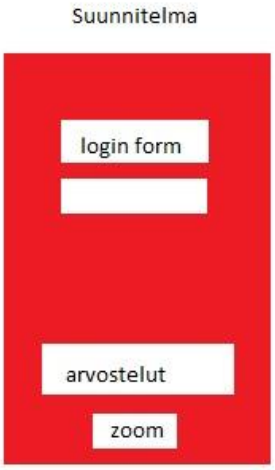
2.8 Sovelluskaupat

Tänäpäivänä sovelluskaupat ovat suuria, sovellusten lataamiseen tarkoitettuja palveluita. Sovelluskaupoista on mahdollista ladata erilaisia sovelluksia, erilaisiin käyttötarkoituksiin. Osa sovelluksista ovat vapaasti ladattavissa ja osa sovelluksista on

maksullisia. Tunnetuimpia sovelluskauppoja ovat: Google Play, App Store ja Windows Phone Store tai Market Place.

2.9 Esimerkkisovellus

Tavoitteenani oli esimerkkikäyttöliittymän lisäksi luoda yksinkertainen natiivi Android Java-sovellus käyttäen Android SDK-kehittäjätyökalua. Lähtökohtana oli luoda hyvin yksinkertainen sovellus, käyttäen hyödyksi Android SDK:n omia graafisia elementtejä. Eclipse Indico-ohjelmaa käyttämällä loin uuden Android-projektin, jolle annoin nimeksi esimerkkisovellus1.



Kuva 11. Esimerkkikäyttöliittymän suunnitelma.

Sovelluksen ja käyttöliittymän suunnitteluvaiheeseen käytin Microsoft Office-pakettia, Gimp-ohjelmaa ja MS-Paintia. Tarkoituksena oli luoda hahmotelma itselle käyttöliittymästä. Väriteemaksi valitsin punaisen ja valkoisen.

```
<color name="red">#FF0000</color>
<color name="white">#FFFFFF</color>
```

Esimerkkisovelluksessa loin yksinkertaisen Hello World-oletusohjelman, johon muokkasin parametreillä erilaisia graafisia elementtejä. Käyttöliittymän Zoomaus-parametrit ActivityMain.xml tiedostosta:

```
<ZoomControls
    android:id="@+id/zoomControls1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="16dp" />
```

Lisäksi ohessa myös arvostelu-parametrit, nimi ja salasanakentät ActivityMain.xml tiedostosta.

```
<EditText
    android:layout_centerHorizontal="true"
    android:ems="10"
    android:inputType="textPassword" />

  <EditText
    android:id="@+id/editText1"
    android:layout_alignLeft="@+id/editText2"
    android:layout_below="@+id/editText2"
    android:layout_marginTop="16dp"
    android:ems="10" />

  <TextView
    android:id="@+id/textView1"
    android:text="@string/hello_world"
    android:textColor="@color/white"
    tools:context=".MainActivity" />



    android:id="@+id/radioGroup1"
    android:layout_alignBottom="@+id/zoomControls1"

  <RatingBar
    android:id="@+id/ratingBar1"
```

```
        android:layout_above="@+id/zoomControls1"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="21dp" />

    <ImageView
        android:id="@+id/imageView1"



    <Button
        android:id="@+id/button1"
        android:layout_alignTop="@+id/imageView1"
        android:layout_centerHorizontal="true"
        android:text="arvostele" />
```

Sivu2.xml-tiedosto:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:background="#ff0000" >
    android:text="@string/teksti_kaksi"
```

String.xml tiedoston tekstiasetus.

```
<string name="teksti_kaksi">Kiitos arvostelustasi!</string>
```

ActivityMain.java on tämän raportin liitteenä (Liite3)

Kuva 12. Esimerkkikäyttöliittymä

3 KÄYTTÖLIITTYMÄSUUNNITTELU

3.1 Yleistä

Käyttöliittymä terminä tarkoittaa laitteen osaa, jota ihminen käyttää kommunikointiin laitteen kanssa. Käyttöliittymän tarkoitus on ohjata ohjelmistoja ja laitteistoja. Nykyiset laitteet, joita päivittäin käytämme, lähes kaikissa on jonkinlainen käyttöliittymä. Mobiililaitteissa, älypuhelimissa kuin tablet-tietokoneissakin laitteet käyttävät pääsääntöisesti graafista käyttöliittymää (eng. graphical user interface).

Kosketusnäytöllisissä laitteissa käyttöliittymän merkitys korostuu ilman fyysistä näppäimistöä. Erilaiset mobiililaitteet sisältävät monia toiminnallisuuksia, kuten kameran, selaimen ja soittimen. Käyttöliittymältä vaaditaan entistä enemmän, jotta käytön tehokkuus säilyisi. (Rouhiainen 1997. Käyttöliittymän visuaalinen suunnittelu. Hakupäivä 16.11.2012)

3.2 Erilaisia käyttöliittymiä

Käyttöliittymiä on erilaisia, erilaisiin käyttötarkoituksiin. Esimerkiksi tablet-tietokoneissa käyttöliittymä pohjautuu kosketusnäyttöön, jossa hyödynnetään paljon widgettejä, buttoneita ja ikoneita. PC-tietokoneissa käyttöliittymät ovat graafisia, jossa navigointi tapahtuu pääasiassa fyysisen hiiren ja näppäimistön avustuksella.

Kuva 13. Mac OSX käyttöjärjestelmän käyttöliittymä.

Kuva 14. Microsoft Windows 8 käyttöjärjestelmän käyttöliittymä

3.3 Käytettävyys

Sovelluksen käytettävyydellä (eng. usability) tarkoitetaan, kuinka hyvin sovellus sopii käyttäjille tehtäväkokonaisuuden suorittamiseen tietyssä ympäristössä ja millaisia henkisiä ja fyysisiä ponnisteluja sen käyttäminen vaatii. Tuotteen käytettävyys määrittelee ratkaiseeko tuote käyttäjän näkökulmasta oikeat ongelmat eli onko tuotteessa käyttäjän tarvitsemat ominaisuudet ja ratkaiseeko tuote ongelmat oikealla tavalla eli lyhyesti onko tuotetta helppo käyttää. Käytettävyyden suunnittelussa opetellaan tuntemaan käyttäjät ja heidän toiveensa ja tarpeensa, jotta käyttäjänäkökulma pystytään huomioimaan suunniteltavassa tuotteessa. (Immonen, Käytettävyys 2003. Hakupäivä 16.11.2012)

Käytettävyys koostuu useasta osatekijästä. Esimerkiksi ISO-standardi (International Organization for Standardization) määrittelee käytettävyyden näin: tarkkuus, tehokkuus ja tyytyväisyys, jolla määritellyt käyttäjät saavuttavat määritellyt tavoitteet tietyssä ympäristössä. ISO-9241-11. Tarkkuus määrittelee onko järjestelmässä käyttäjän kannalta oikeat ominaisuudet. Tehokkuus määrittelee kuinka helppoa ja nopeaa järjestelmän käyttäminen käyttäjälle on. Tyytyväisyys määrittelee onko järjestelmän käyttäminen käyttäjän mielestä miellyttävää ja onko käyttäjä tyytyväinen käyttötilanteeseen eli pitääkö käyttäjä tuotteesta. (VTT 2012. Hakupäivä 16.11.2012)

3.4 Käyttökokemus

Käyttökokemus (user experience) on kokonaisvaltainen kokemus, joka käyttäjällä syntyy vuorovaikutuksesta sähköiseen palveluportaaliin tai mobiilikäyttöliittymään. Käyttökokemukseen vaikuttavat palvelun kokonaiskonsepti, toiminnot, sisältö, käytettävyys, visuaalisuus, mielikuvat ja käyttötilanne. Positiivinen käyttökokemus parantaa käyttäjäuskollisuutta ja brandimielikuvaa. (Logica 2012. Hakupäivä 17.11.2012)

3.5 Käyttöliittymän elementit

Graafinen käyttöliittymä koostuu erilaisista käyttöliittymäelementeistä, jotka ovat käyttäjälle näkyviä asioita tai toimintoja esittäviä kuvioita. Käyttäjä toimii ja tekee valintoja osoittamalla esimerkiksi hiirellä kuvioita tai niiden osia.

Mobiilikäyttöjärjestelmissä käyttöliittymän peruselementit ovat pääasiassa kuvakkeita, ikoneita, listausta ja painikkeita. Näitä käyttämällä saa avatuksi uusia elementtejä ja niiden avulla lisää toiminnallisuuksia. Yleensä graafinen käyttöliittymä koostuu useista eri elementeistä ja ne näkyvät näytöllä yhtä aikaa ja ovat usein yhdistettynä toisiinsa. (Graafisen käyttöliittymän elementit 2003. Hakupäivä 17.11.2012)

3.6 Esimerkkikäyttöliittymän suunnittelun työvaiheet

Käyttöliittymäsuunnittelun ensimmäinen vaihe on yleensä käyttäjän tarpeiden kartoitus. Koska kyseessä on oma sovellukseni, sovelluksen tavoitteena on vain luoda muutama elementti malliksi. Käyttöliittymäsuunnitteluprosessi tulisi olla mielestäni tarkasti dokumentoitua ja käyttäjälähtöistä. Suunnittelutyössä laadunvarmistukseen käytin prosessimenettelyä, jossa suunnittelen ensimmäisenä näyttöjen tilankäytön asettelun ja navigoinnin logiikan. Tätä kutsutaankin joskus ns. rautalankamalliksi, jossa ei vielä ole graafisia elementtejä. Tämän vaiheen jälkeen suunnittelin esimerkkisovelluksen elementit, värit ja fontit. Sovellus koostuu yksinkertaisista elementeistä, kuten nimi- ja salasana-tekstikentistä, arviointiasteikosta ja zoomaus-asetuksesta, buttoneista ja taustaväristä.

4 KYSELYTUTKIMUS

4.1 Tutkimuksen tavoitteet

Kyselytutkimuksen tavoitteena oli saada käsitys käyttäjien tarpeista käyttöliittymiä kohtaan käyttöliittymäsuunnitteluvaihetta varten, mitä ihmiset haluavat mobiililaitteen käyttöliittymältä, mitä asioita käyttäjät käyttöliittymässä arvostavat, missä on kehitettävää ja vaikuttaako käyttöliittymä laitehankinnansuunnitteluun. Tutkimuksen tavoitteena oli luoda yleiskäsitys graafisten käyttöliittymien merkityksestä.

4.2 Tutkimuksen toteutuksen vaiheet

Kyselytutkimuksen suunnitteluun käytin Microsoft Word-asiakirjaa, johon loin alustavat kysymykset. Keräsin ison kuvan käyttöliittymien eri elementteihin liittyviä kysymyksiä. Kyselyn toteutusvaiheessa käytin ilmaista SurveyMonkey-internetsivustoa kyselyn luomista varten. Kyselyyn muodostui lopulta 10 eri kysymystä koskien mobiililaitteiden käyttöliittymiä ja käytettävyyttä. Halusin pitää kyselyn lyhyenä ja ytimekkäänä, jotta mahdollisimman moni vastaisi siihen. Kysely on tämän raportin liitteenä. (Liite1).

4.3 Tutkimuksen tulokset

Kyselytutkimuksessa halusin ottaa selville mobiililaitteiden käyttäjien erilaisia käyttökokemuksia. Tarkoituksena oli saada jonkinlaista yleiskuvaa siitä, kuinka tärkeänä käyttäjät pitävät käyttöliittymien toimivuutta ja visuaalisuutta. Kyselytutkimuksesta käy ilmi, että käyttöliittymällä on käyttäjille hyvinkin suuri merkitys. 71,3% vastaajista arvioi uutta mobiililaitetta ostaessaan käyttöliittymällä olevan merkitystä. Kyselyyn vastanneiden mobiililaitteista suurinosa käyttää Android käyttöjärjestelmää 43,8%. iOS ja Windows Phone saivat 20% osuuden. Muut olivat selvästi pienempiä käyttöjärjestelmiä. Historiaan jäävää Symbiania oli vielä 11,3% mobiililaitteiden käyttöjärjestelmistä.

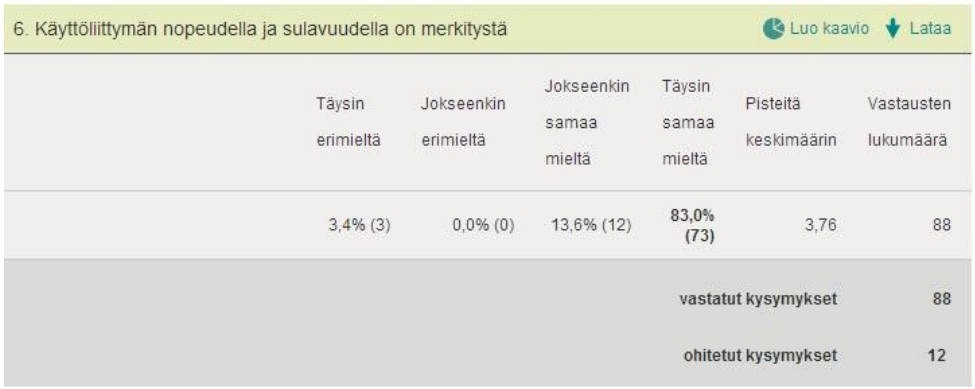
4. Mitä käyttöjärjestelmää mobiililaitteesi käyttää?

Luo kaavio | Lataa

	Vastausten prosenttiosuus	Vastausten lukumäärä
Android	43,8%	35
iOS	20,0%	16
Windows Phone	20,0%	16
Symbian	11,3%	9
MeeGo	1,3%	1
Muu	3,8%	3
	vastatut kysymykset	80
	ohitetut kysymykset	20

Kuva 15. Kyselytutkimuksen tulokset käyttöjärjestelmistä.

Eniten merkitystä käyttäjille oli selvästi käyttöliittymän nopeudella ja sulavuudella. Peräti 83% vastaajista oli sitä mieltä, että käyttöliittymän nopeudella ja sulavuudella oli erittäin tärkeä merkitys.



6. Käyttöliittymän nopeudella ja sulavuudella on merkitystä

Luo kaavio | Lataa

	Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
	3,4% (3)	0,0% (0)	13,6% (12)	**83,0% (73)**	3,76	88
					vastatut kysymykset	88
					ohitetut kysymykset	12

Kuva 16. Kyselytutkimuksen tulokset käyttöliittymän nopeuden merkityksestä.

Käyttöliittymän visuaalisella ilmeellä ja kuvakkeiden ja ikonien värimaailman merkityksellä vastausten jakauma oli suurempi (liite 2). Tästä voidaan päätellä, että käyttöliittymän tärkeimmät ominaisuudet ovat sulavuus, nopeus ja muokattavuus. Visuaalisuuden merkitys on osalle käyttäjille kuitenkin tärkeää, joten sekin puoli on hyvä huomioida käyttöliittymäsuunnittelun eri vaiheissa. Käyttöliittymässä suunnitteluvaihe on erityisen tärkeää ja se olisi hyvä tehdä asiakaslähtöisesti.

5 POHDINTA

Android-sovelluksen ja esimerkkikäyttöliittymän luominen soveltuu mielestäni erinomaisesti ohjelmoinnin perustaitojen harjoitteluun, koska ohjelmointia voi parhaiten harjoitella vain ohjelmoimalla. Sovelluksen suunnittelu-, toteutus- ja testausvaiheessa huomasin, että dokumentointi on erityisen tärkeää mitä pidemmälle projekti etenee. Tiedostojen latauspaikat, tallennukset, versionumerot, kuvat, ikonit, ja muut elementit on hyvä pitää varmuuskopioituna mielellään useassa paikassa. Alun perin suunnitellun aikataulutuksen pitäminen oli äärettömän vaikeaa, sillä projekti vei huomattavasti enemmän aikaa, mitä suunnitelmaan olin ajatellut. Varsinkin esimerkkisovelluksen ohjelmointi vei aikaa. Tutkielmaan sisältyvän kyselytutkimuksen tulosten perusteella käy hyvin ilmi, että käyttöliittymän tärkeimmät ominaisuudet käyttäjien mielestä olivat sulavuus, nopeus ja muokattavuus. Suurinosa käyttäjistä arvostaa erityisesti käyttöliittymän nopeutta käyttöjärjestelmässä. Otin huomioon käyttöliittymän suunnittelussa nopeuden ja sulavuuden tekemällä käyttöliittymästä suhteellisen yksinkertaisen ja vähän dataa käyttävän. Kaikenkaikkiaan käyttöliittymäsuunnittelu ja mobiiliohjelmointi ovat hyvin lähellä toisiaan ja nykyisin laitevalmistajat valmistavat suuria määriä kosketusnäytöllisiä laitteita, joten käyttöliittymien merkitys korostuu entisestään fyysisten osoitin- ja oheislaitteiden vähentyessä. Web-tekniikat yleistyvät kovalla vauhdilla mobiiliohjelmoinnissa, joiden etuna on rakentaa sovelluksia useille rajapinnoille ja eri mobiilikäyttöjärjestelmille. Sovelluskehittäjän kannattaa siis tutustua ns. Multi Platform SDK-kehittäjätyökaluihin. Esimerkkisovellukseni on Natiivi Android-sovellus, mutta jatkossa aion panostaa HTML5- ja Jquery-ohjelmointikieliin.

LÄHTEET

3Scale 2012. What is an API? Your guide to the Internet Business Revolution Hakupäivä 22.11.2012. <http://www.3scale.net/wp-content/uploads/2012/06/What-is-an-API-1.0.pdf>

Apple 2012. iOS. Maailman kehittynein mobiilikäyttöjärjestelmä. Hakupäivä 05.11.2012. <http://www.apple.com/fi/ios/what-is/>

Benson, Anne & Bergius, Tanja 2006. Eclipse 3.1 Pikaopas. Hakupäivä 12.11.2012. <http://myy.haagahelia.fi/~ict1td002/OSIO_1_IltaTiko/Esimerkit/eclipse_ohje_koste1.pdf>

Carmen, Mary 2012. Most commonly used programming languages for mobile applications. Hakupäivä 29.10.2012. <http://www.onbile.com/info/most-commonly-used-programming-languages-for-mobile-applications/>

Google 2012. Android. Hakupäivä 22.11.2012. <http://www.android.com/about/>

Google 2012. Mobile Security. Hakupäivä 08.11.2012. <http://www.google.fi/intl/fi/goodtoknow/online-safety/mobile-security/>

Heinonen, Mauri 2010. Javan perusteita. Hakupäivä 22.11.2012. <http://ohjelmointi.medianurkka.com/wp-content/uploads/2010/04/java.pdf>

Hyvönen, Martti & Lappalainen, Vesa 2010. Ohjelmointi1. Hakupäivä 07.11.2012. <http://kurssit.it.jyu.fi/ITKP102/moniste/html/moniste.html#o1Mita_ohjelmointi_on_>

Immonen, Jarkko 2003. Graphical User Interfaces. Hakupäivä 22.11.2012. <http://cs.joensuu.fi/~jimmonen/gkl_moniste/gkl_v202.html>

Interacta 2012. GUIDe- Käyttöliittymäsuunnittelu. Hakupäivä 22.11.2012. <http://www.interacta.fi/suunnitteluvaihe.html>

Jolla 2012. Sailfish. Hakupäivä 22.11.2012. <http://jolla.com/>

Jquery Foundation 2012. jQuery About. Hakupäivä 22.11.2012. <http://jquery.org/about/>

Logica 2012. Käyttökokemus ja käytettävyys. Hakupäivä 22.11.2012. <http://www.logica.fi/we-do/enterprise-content-management/usability-and-user-experience/>

Luoma.H & Kaunisto.R 2007. Mobiilit käyttöjärjestelmät. Hakupäivä 22.11.2012. <http://robo.cop.fi/eo_linux/Mobiilikayttojarjestelmat/Mobiilikayttojarjestelmat.pdf>

Maemo.org 2012. Intro. Hakupäivä 22.11.2012. <http://maemo.org/intro/>

Meego.com 2012. Meego About. Hakupäivä 22.11.2012. <https://meego.com/about>

Metabolix, Ohjelmointiputka 2009. C++-ohjelmointi Osa1. Hakupäivä 14.11.2012. <http://www.ohjelmointiputka.net/oppaat/opas.php?tunnus=cpp_ohj_01>

Microsoft 2012. Visual Studio Express For Windows Phone. Hakupäivä 22.11.2012. <http://www.microsoft.com/visualstudio/eng/products/visual-studio-express-for-windows-phone#product-express-phone-details>

Microsoft 2012. Windows Phone Suomi. Hakupäivä 22.11.2012. <http://www.windowsphone.com/fi-fi>

Nokia 2012. Symbian Platform. Hakupäivä 22.11.2012. <http://www.developer.nokia.com/Devices/Symbian/>

Oracle 2012. What is J2ME or Java Me? Hakupäivä 22.11.2012.

<http://java.com/en/download/faq/whatis_j2me.xml>

Pihlajamäki, Teppo 2010. Mobiililaitteiden cross-platform ohjelmointi web-tekniikoilla. Hakupäivä 28.10.2012.
<http://teppo.tv/kirjoituksia/mobile.html>

Pitkänen, Jarmo 2012. Softakehitystä Applen alustoille. Hakupäivä 22.11.2012.
<http://www.tietokone.fi/softa/macintosh/xcode_4_5>

Rouhiainen, Eeva-Kaisa 1997. Käyttöliittymän visuaalinen suunnittelu. Hakupäivä 16.11.2012.
<http://www.mit.jyu.fi/opiskelu/seminaarit/bak/kayttoliittyma/index.html#luku2>

Rouse, Margaret 2007. Integrated development environment (IDE). Hakupäivä 07.11.2012.
<http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>

Rouse, Margaret 2005. Software Developers kit (SDK). Hakupäivä 07.11.2012.
<http://searchcio-midmarket.techtarget.com/definition/software-developers-kit>

Rouse, Margaret 2012. What Is Native Code?. Hakupäivä 23.11.2012.
<http://searchsoa.techtarget.com/definition/native-code>

Saarinen, Aki & Järvensivu, Tuomas & Rosendahl, Sami 2011. Qt-sovelluskehitys. Hakupäivä 22.11.2012.
<http://reaktor.fi/osaaminen/qt-sovelluskehys/>

Särkkä, Simo 1998. JavaScript opas. Hakupäivä 22.11.2012.
<http://users.tkk.fi/u/ssarkka/javascript/index.html>

Tizen.org 2012. Tizen About. Hakupäivä 22.11.2012.
<https://www.tizen.org/about>

VTT 2012. Mitä käytettävyys tarkoittaa?. Hakupäivä 22.11.2012.
<http://www.vtt.fi/research/technology/contextawareservices/hti_what_usability.jsp?lang=fi>

W3Schools 2012. HTML5 Introduction. Hakupäivä 22.11.2012.
<http://www.w3schools.com/html/html5_intro.asp>

LIITTEET

Liite 1.

Kyselytutkimus

1. Sukupuolesi

Sukupuolesi Nainen
Mies

2. Ikäsi

Ikäsi 18-25
26-35
36-45
46-60
yli 60

3. Omistan mobiililaitteen, esimerkiksi älypuhelimen tai tablet-tietokoneen

Omistan mobiililaitteen, esimerkiksi älypuhelimen tai tablet-tietokoneen Kyllä
Ei

4. Mitä käyttöjärjestelmää mobiililaitteesi käyttää?

Mitä käyttöjärjestelmää mobiililaitteesi käyttää? Android
iOS
Windows Phone
Symbian
MeeGo
Muu

5. Käyttöliittymän visuaalinen ilme on tärkeää

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä

6. Käyttöliittymän nopeudella ja sulavuudella on merkitystä

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä

7. Kiinnitän huomiota käyttöliittymän kuvakkeiden ja ikonien ulkonäköön ja värimaailmaan

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä

8. Aloitusnäytön muokattavuus on tärkeä ominaisuus

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä

9. Olen tyytyväinen mobiililaitteeni käyttöliittymään

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä

10. Uutta mobiililaitetta ostaessani, käyttöliittymällä on merkitystä

Uutta mobiililaitetta ostaessani, käyttöliittymällä on merkitystä Kyllä
Ei
En osaa sanoa

Liite 2.

Kyselytutkimuksen tulokset

Yhteenveto vastauksista

Sivu: 1

1. Sukupuolesi

	Vastausten prosenttiosuus	Vastausten lukumäärä
Nainen	59,0%	59
Mies	41,0%	41

2. Ikäsi

	Vastausten prosenttiosuus	Vastausten lukumäärä
18-25	30,0%	30
26-35	32,0%	32
36-45	25,0%	25
46-60	12,0%	12
yli 60	1,0%	1

3. Omistan mobiililaitteen, esimerkiksi älypuhelimen tai tablet-tietokoneen

	Vastausten prosenttiosuus	Vastausten lukumäärä
Kyllä	79,0%	79
Ei	21,0%	21

4. Mitä käyttöjärjestelmää mobiililaitteesi käyttää?

	Vastausten prosenttiosuus	Vastausten lukumäärä
Android	43,8%	35
iOS	20,0%	16
Windows Phone	20,0%	16
Symbian	11,3%	9
MeeGo	1,3%	1
Muu	3,8%	3

5. Käyttöliittymän visuaalinen ilme on tärkeää

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
5,7% (5)	3,4% (3)	53,4% (47)	37,5% (33)	3,23	88

6. Käyttöliittymän nopeudella ja sulavuudella on merkitystä

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
3,4% (3)	0,0% (0)	13,6% (12)	83,0% (73)	3,76	88

7. Kiinnitän huomiota käyttöliittymän kuvakkeiden ja ikonien ulkonäköön ja värimaailmaan

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
0,0% (0)	19,3% (17)	52,3% (46)	28,4% (25)	3,09	88

8. Aloitusnäytön muokattavuus on tärkeä ominaisuus

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
2,3% (2)	18,2% (16)	37,5% (33)	42,0% (37)	3,19	88

9. Olen tyytyväinen mobiililaitteeni käyttöliittymään

Täysin erimieltä	Jokseenkin erimieltä	Jokseenkin samaa mieltä	Täysin samaa mieltä	Pisteitä keskimäärin	Vastausten lukumäärä
1,3% (1)	11,4% (9)	53,2% (42)	34,2% (27)	3,2	79

10. Uutta mobiililaitetta ostaessani, käyttöliittymällä on merkitystä

	Vastausten prosenttiosuus	Vastausten lukumäärä
Kyllä	71,3%	67
Ei	6,4%	6
En osaa sanoa	22,3%	21

Liite 3.

MainActivity.java:n lähdekoodia:

```
package com.example.esimsovellus1;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.MenuItem;
import android.support.v4.app.NavUtils;
public class MainActivity extends Activity {
@Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }
@Override
   public boolean onCreateOptionsMenu(Menu menu) {
      getMenuInflater().inflate(R.menu.activity_main, menu);
      return true;
   }
```