

Android-sovelluksen kehitys PhoneGap- sovelluskehityksen avulla

Janne Varis



Tekijä(t) Janne Varis	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Opinnäytetyön otsikko Android-sovelluksen kehitys PhoneGap-sovelluskehityksen avulla	Sivu- ja liitesivumäärä 31 + 0
Opinnäytetyön otsikko englanniksi Developing an Android application using the PhoneGap mobile development framework	
<p>Tämän projektin tavoitteena oli kehittää avoimen lähdekoodin sovellus Android-käyttöjärjestelmää käyttäville puhelimille ja tablet-laitteille käyttäen PhoneGap-sovelluskehystä, joka mahdollistaa sovellusten kehittämisen web-teknologioilla kuten HTML, CSS ja JavaScript, jolloin sovelluksen kehittämisessä ei tarvita Androidin natiiveja ohjelmointikieliä.</p> <p>Oppimistavoitteina tämän projektin aikana olivat PhoneGapin etujen selvittäminen ja sovelluskehityksen tarkka, mutta selkeä dokumentointi. PhoneGapin etujen tutkiminen ja dokumentointi tukevat kirjoittajan tulevaa sovelluskehitystä, ja sovelluskehityksen dokumentointi taas parantaa osaamista selkeässä tiedon tuomisessa lukijoille.</p> <p>Nimellä SetNote kehitetty sovellus on digitaalinen muistio, johon voi tallentaa tietoja muistilapuille ja lähettää niitä edelleen muille sähköpostitse. Valmis sovellus julkaistiin Google Play-sovelluskaupassa GNU GPL3-lisenssillä.</p> <p>Opinnäytetyön tavoitteena oli myös julkaista sovellus ja sen lähdekoodi, dokumentoida sen kehitysprosessi ja selvittää mitä etuja PhoneGap-sovelluskehiksestä ja sen mahdollistamista tekniikoista on sovellusten kehityksessä Android-käyttöjärjestelmälle.</p> <p>Perustuen tämän projektin aikana tehtyyn hybridisovelluksen kehitystyöhön ja moniin luotettaviin Android-kehittäjäyhteisön lähteisiin, tämä tutkimus on selvästi todistanut, että PhoneGap-sovelluskehityksen käytöllä on selviä etuja ja sen avulla voi saada etulyöntiaseman hybridisovellusten kehittämisessä Android-käyttöjärjestelmälle.</p>	
Asiasanat PhoneGap, Android, mobiilisovelluskehitys	

Author(s) Janne Varis	
Degree programme Degree programme in Business Information Technology	
Report/thesis title Android application development by using the PhoneGap framework	Number of pages and appendix pages 31+0
<p>The goal of this project was to develop an open source mobile application for Android smartphones and tablet PCs by using the PhoneGap framework, which allows for the development of applications by using web technologies like HTML, CSS and JavaScript, therefore rendering the native programming languages for the Android operating system unnecessary in the development process.</p> <p>The learning goals during this project were examining the advantages of PhoneGap, and the accurate, but clear, documentation of the application development process. Finding out the benefits of PhoneGap supports future application development, and the documentation of the application development process improves the skillset for providing clear information to the reader.</p> <p>Developed under the name SetNote, the application consists of a digital notebook, in which the user can save information as notes and send them to people by email. The finished application was published in the Google Play Store under the GNU GPL3 license.</p> <p>It was also the goal of this thesis to publish the mobile application and its source code, document its development process, and to find out what kind of benefits the PhoneGap framework and the techniques enabled by it offer in terms of developing applications for the Android operating system.</p> <p>Based on the actual hybrid application development process undertaken during this project and numerous credible sources in the Android developer community, the study has conclusively demonstrated that the PhoneGap framework has clear advantages and benefits in hybrid Android application development.</p>	
Keywords PhoneGap, Android, mobile, application, development	

Sisällys

1	Johdanto	1
2	Kehitysympäristön kuvaus	3
2.1	Laitteisto	3
2.2	Käyttöjärjestelmä	3
2.3	PhoneGap-sovelluskehys	4
2.4	Android SDK	4
2.5	Intel XDK.....	4
2.6	Node.js	5
2.7	Apache Ant	5
2.8	Oracle Java JDK	5
2.9	jQuery ja jQuery Mobile.....	5
3	Kehitysympäristön asennus.....	7
3.1	Oracle Java JDK	7
3.2	Apache Ant ja Node.js.....	8
3.3	Android SDK	8
3.4	PhoneGap-sovelluskehys	10
3.5	Intel XDK.....	10
4	Sovelluksen kehitys.....	12
4.1	PhoneGap-projektin luominen.....	12
4.2	Projektin lisääminen Intel XDK-ohjelmaan.....	12
4.3	Testilaitteen liittäminen.....	14
4.4	Sovelluksen rakenne	15
4.5	jQuery- ja jQuery Mobilen lisäys.....	16
4.6	ThemeRoller-teeman lisäys.....	17
4.7	Sovelluksen käyttöliittymä	18
4.8	Sovelluksen ikoni	22
4.9	Sovelluksen lähdekoodin kommentointi	23
4.10	Julkaisu ja GNU GPL3-lisenssiehdot.....	26
5	PhoneGap-sovelluskehysten hyödyt kehityksessä	27
5.1	Ohjelmointi.....	27
6	Pohdinta.....	29
	Lähteet	30

1 Johdanto

Tämän opinnäytetyön tavoitteena oli kehittää avoimen lähdekoodin sovellus Android-käyttöjärjestelmää käyttäville puhelimille ja tablet-laitteille. Sovelluskehityksen pohjana oli PhoneGap-sovelluskehys, joka mahdollistaa sovellusten kehittämisen web-teknologioilla (mm. HTML, CSS ja JavaScript), jolloin kehityksessä ei tarvitse käyttää Androidin natiiveja ohjelmointikieliä, vaan se voidaan toteuttaa niin kutsutun hybridi-kehityksen kautta.

Opinnäytetyön tavoitteena oli myös julkaista sovellus ja sen lähdekoodi, dokumentoida sen kehitysprosessi ja selvittää mitä etuja PhoneGap-sovelluskehiksestä ja sen mahdollistamista tekniikoista on sovellusten kehityksessä Android-käyttöjärjestelmälle.

Nimellä SetNote kehitetty sovellus on digitaalinen muistio, johon voi tallentaa helposti tietoja muistilapuille ja lähettää niitä edelleen muille sähköpostitse helppokäyttöisen ja visuaalisesti miellyttävän käyttöliittymän kautta.

Kehitys tapahtui Xubuntu Linux-käyttöjärjestelmässä, johon asennettiin tarvittava kehitysympäristö. Kehitysympäristöön kuuluvia osia olivat PhoneGap-sovelluskehiksen lisäksi Androidin kehitystyökalut (engl. *Android software development kit*), Intel XDK-ohjelma, jQuery-JavaScript-kirjasto ja jQuery Mobile-käyttöliittymäkehys. Sovelluskehitys tapahtui pääasiallisesti Intel XDK-ohjelman kautta, koska siinä on koodieditorin lisäksi myös laiteemulaattori Androidille, joten sovelluksen testaaminen onnistuu myös virtuaalisesti, eikä sitä tarvitse asentaa jatkuvasti testipuhelimeen tai -laitteeseen.

Opinnäytetyön konkreettisiin, mitattaviin ja todennettavissa oleviin tuloksiin kuuluvat itse Android-sovellus, sen julkaistu lähdekoodi, dokumentaatio sovelluksen kehityksestä ja kaikkeen edellä mainittuun perustuva tutkimus PhoneGap-sovelluskehiksen hyödyistä.

Työmenetelmät koostuivat sekä kehitysvaiheiden dokumentoinnista että käytännön sovelluskehityksestä. Nämä työmenetelmät toimivat käytännössä rinnakkain, koska sovelluskehityksen dokumentointi suoritettiin samanaikaisesti sen kehityksen kanssa virheettömyyden takaamiseksi.

Rajauksien osalta projekti keskittyi Android-näkökulmaan tutkimuksessa ja kehityksessä, jotta, sen laajuus saatiin pysymään järkevissä mitoissa. Projekti ei toteuttanut sovellusta millekään toimijalle, vaan se oli täysin itsenäinen kehitysprojekti, jonka lopputuloksena syntynyt Android-sovellus julkaistiin GNU GPL3-lisenssillä.

Tämän pääluvun 1. johdannon jälkeen opinnäytetyö etenee johdannon jälkeen sovelluskehityksen logiikan mukaisessa järjestyksessä. Pääluvussa 2 käsitellään kehityksessä käytettävä laitteisto ja käyttöympäristö sekä kehitysympäristö ja sen osat yksittäin. Tätä osiota seuraa kehitysympäristön asennuksen dokumentointi pääluvussa 3.

Pääluvussa 4 käydään läpi mobiilisovellusprojektin luonti PhoneGap-sovelluskehityksen avulla ja sen vieminen Intel XDK-sovellukseen. Osiossa käsitellään myös testilaitteen liittäminen, sovelluksen tiedostorakenne, jQuery ja jQuery Mobile-kirjastojen lisäys ja käydään läpi sovelluksen valmis koodi. Lopuksi käsitellään sovelluksen ja sen lähdekoodin julkaisu sekä GNU GPL3-lisenssin ehdot.

Pääluvussa 5 osiossa tutkitaan mitä etuja PhoneGap-sovelluskehityksestä ja sen mahdollistamista tekniikoista oli sovelluksen kehittämisessä Androidille. Pääluvussa 6 eli pohdinta-osiossa käydään läpi opinnäytetyön tulokset ja tehdään yhteenveto.

2 Kehitysympäristön kuvaus

2.1 Laitteisto

Kehityksessä käytetty kannettava tietokone oli malliltaan HP EliteBook 2570p. Seuraavassa listassa ovat tärkeimmät tiedot sen komponenteista.

- Prosessori: Intel Core i5-3210M @ 2.50 GHz
- Näytönohjain: Integroitu, Intel HD Graphics 4000
- RAM-muisti: 8 gigatavua, DDR3
- Kiintolevy: Hitachi Travelstar HTS72505, 500 gigatavua
- Näytön koko: 12.5"
- Näytön resoluutio: 1366x768 pikseliä

Kehityksessä käytettävä testilaite oli OnePlus-yrityksen OnePlus One Android-puhelin. Seuraavassa listassa ovat tärkeimmät tiedot sen komponenteista.

- Android-versio: 4.4.4 KitKat (tämän opinnäytetyön aikana käytetty versio)
- Piirisarja: Qualcomm MSM8974AC Snapdragon 801
- Prosessori: Quad-core 2.5 GHz Krait 400
- Näytönohjain: Adreno 330
- RAM-muisti: 3 gigatavua
- Tallennustila: 64 gigatavua
- Näytön koko: 5.5"
- Näytön resoluutio: 1080 x 1920 pikseliä

(GSMArena 2014.)

2.2 Käyttöjärjestelmä

Kehitystyön käyttöjärjestelmäksi valikoitui Xubuntu Linux. Sovelluskehitys tapahtui kokonaan Xubuntu Linuxissa, koska kehitysympäristön asentaminen ja käyttäminen oli sillä kirjoittajalle luontevinta ja nopeinta. Tähän vaikutti suuresti esimerkiksi Xubuntun paketinhallinta, joka mahdollistaa nopeat päivitykset ja asennukset suoraan käyttöjärjestelmän komentoriviltä (johon tästä eteenpäin viitataan termillä terminaali). Xubuntun käyttäminen kehitysympäristönä mobiilisovelluskehityksessä oli tämän opinnäytetyön kirjoittajalle tuttua myös aiemmalta HAAGA-HELIA ammattikorkeakoulun kurssilta, joka käsitteli PhoneGap-sovellusten kehittämistä Linuxissa. Xubuntusta asennettiin kirjoitushetkellä uusin saatavilla oleva versio, 14.10 Utopic Unicorn (Xubuntu, 2015).

2.3 PhoneGap-sovelluskehys

PhoneGap on Adoben avoimen lähdekoodin sovelluskehys (engl. *framework*), jonka avulla voidaan kehittää sovelluksia eri alustoille käyttäen web-tekniologioita. Näitä ovat esimerkiksi HTML, CSS ja JavaScript. PhoneGap-sovelluskehys mahdollistaa siis esimerkiksi HTML-, CSS- ja JavaScript-tiedostoista koostuvan web-sovelluksen tai sivuston paketoimisen sellaiseen muotoon, että se voidaan asentaa sovelluksena eri käyttöjärjestelmille, kuten tämän opinnäytetyön tapauksessa Androidille. Tällaiseen kehitykseen viitataan myös termillä hybridikehitys. (PhoneGap 2015.)

PhoneGap-sovelluskehys tarjoaa käyttöön myös omat rajapintansa, esimerkiksi laiterajapinnan mobiililaitteiden kameran käyttöön. Näitä rajapintoja hyödyntäen on mahdollista kehittää web-tekniologioita käyttäen sovelluksia, joiden ominaisuudet vastaavat pitkälti natiiveilla ohjelmointikielillä kehitettyjen sovellusten toiminnallisuutta, mutta ilman tarvetta käyttää eri käyttöjärjestelmien natiiveja ohjelmointikieliä. Usealle alustalle kehitettäessä tämä säästää huomattavasti aikaa, koska on mahdollista hyödyntää sama lähdekoodi eri käyttöjärjestelmille käännettävässä sovelluksessa. Käytännössä PhoneGap-sovelluskehysten avulla uudelle käyttöjärjestelmälle lisätään vain alusta (engl. *platform*) projektiin, jotta sovellus voidaan kääntää uudelle käyttöjärjestelmälle sopivaan muotoon. Tämän opinnäytetyön kirjoitushetkellä PhoneGap-sovelluskehys tukii seuraavia käyttöjärjestelmiä: Amazon Fire OS, Android, Blackberry 10, Firefox OS, iOS, Ubuntu, Windows Phone 8, Windows ja Tizen. (PhoneGap 2015.)

2.4 Android SDK

Android SDK (engl. *software development kit*) tai Androidin kehitystyökalut, on Googlen tarjoama paketti, joka tarvitaan Android-sovellusten kehitystä varten. Kehitystyökaluihin kuuluvat työkalujen itsensä ohella myös eri Android-versioiden rajapinnat, jotka mahdollistavat kehityksen käyttöjärjestelmän eri versioille.

2.5 Intel XDK

Intel XDK on HTML5 useille eri alustoille suunniteltu kehitystyökalu (engl. *cross-platform development tool*), joka tarjoaa yksinkertaisen kehitysnäkymän HTML5- ja hybridisovellusten kehittämiseen eri alustoille. (Intel 2015a.) Intel XDK sisältää koodieditorinsa lisäksi myös virtuaalisen laite-emulaattorin eri käyttöjärjestelmille, joihin kuuluu myös Android, jolle tämän opinnäytetyön SetNote-sovellus kehitettiin. Edellä mainittu emulaattori on kehityksessä erittäin hyödyllinen, koska se on erittäin nopea ja näyttää muutokset sovelluk-

sessä suoraan tietokoneella ilman, että sovellusta tarvitsee jatkuvasti asentaa uudestaan testilaitteeseen.

2.6 Node.js

Node.js on avoimen lähdekoodin useille eri alustoille suunniteltu runtime environment palvelinpuolen- ja verkkosovelluksille. Node.js mahdollistaa web-palvelimien ja verkkotyökalujen luomisen JavaScriptiä ja ydintoiminnallisuuksia hoitavia moduuleja käyttäen. (Node.js 2015.) Tämän opinnäytetyön kehitysympäristössä Node.js oli tarpeellinen, koska sitä käytettiin PhoneGap-sovelluskehityksen asentamiseen.

2.7 Apache Ant

Apache Ant on Apache Software Foundationin Java-kirjasto ja komentorivityökalu, jonka tehtävä on ajaa prosesseja, jotka on build-tiedostoissa määritetty kohteiksi ja jatkopisteiksi, jotka ovat toisistaan riippuvaisia. Apache Antin tunnetuin käyttötarkoitus on Java-sovellusten rakentaminen. Sitä voi käyttää myös muiden kuin Java-sovellusten rakentamiseen. Yleisemmin sen käyttötarkoitus voi olla minkä tahansa prosessin ohjaaminen, joka voidaan määritellä kohteina ja tehtävinä. (Apache Ant 2015.) Apache Ant tarvittiin tässä kehitysympäristössä, jotta PhoneGap pystyi rakentamaan apk-asennustiedoston Android-laitteita varten (Binstein 21.7.2014).

2.8 Oracle Java JDK

Java on joukko muutamia Sun Microsystemsin kehittämiä tietokoneohjelmia ja spesifikaatioita, joka tarjoaa järjestelmän sovellusohjelmien kehittämiseen ja käyttöönottoon cross-platform-ympäristössä. Nykyisin Javan omistaa Oracle Corporation. (Oracle 2015.) Opinnäytetyön kehitysympäristössä Java JDK (Java Development Kit) paketti tarvittiin Javan koontiin (Binstein 21.7.2014).

2.9 jQuery ja jQuery Mobile

jQuery ja jQuery Mobilea ei varsinaisesti lisätty kehitysympäristöön vaan itse sovellusprojektiin sen jälkeen, kun se oli luotu. Niiden tarve oli kuitenkin selvä jo ennen projektin luomista, joten ne on syytä käsitellä erillisinä kehitysympäristön osina tässä luvussa.

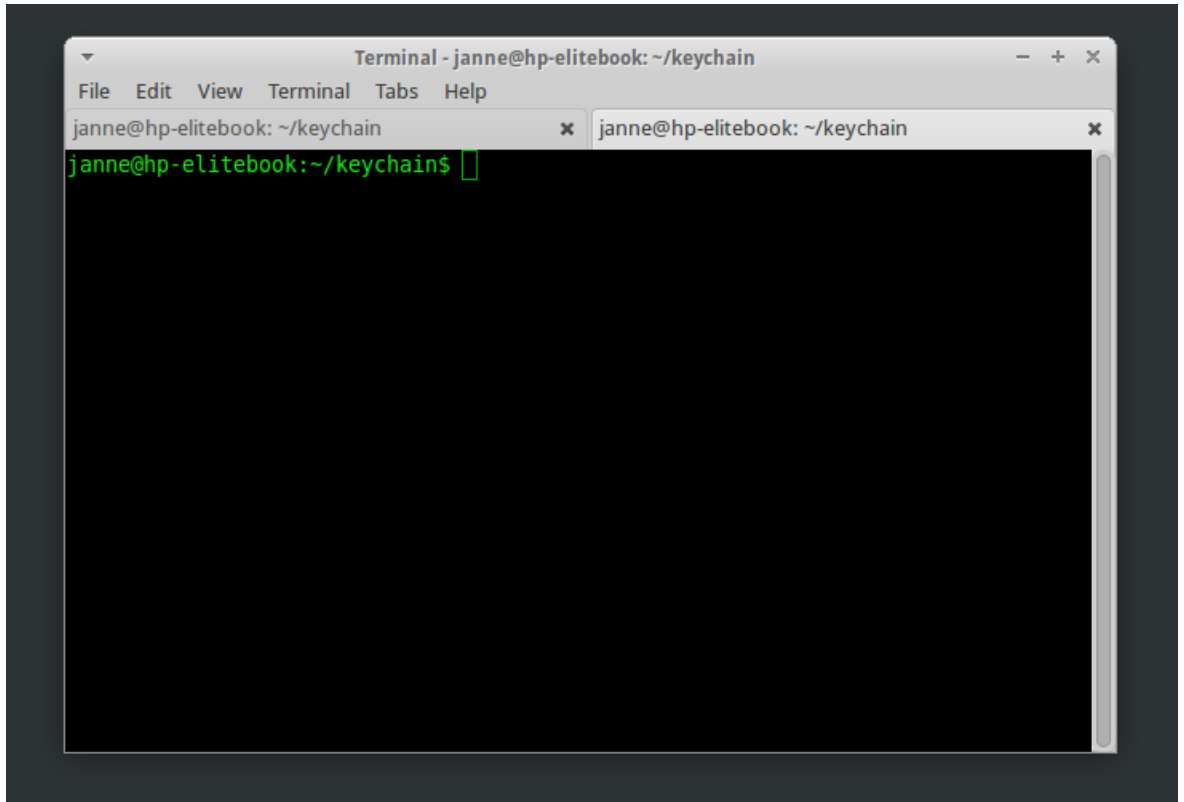
jQuery-JavaScript-kirjasto tekee asioista kuten HTML-dokumenttien läpikäynnistä ja manipuloinnista, tapahtumien käsittelystä, animaatioista ja Ajaxista paljon helpompaa helpokäyttöisellä rajapinnalla, joka toimii monissa eri selaimissa (jQuery 2015).

jQuery Mobile-käyttöliittymäkehys on suunniteltu skaalautuvien sivustojen ja sovellusten käyttöliittymien kehittämiseen kaikille mobiililaitteille. Siihen kuuluu myös web-pohjainen Theme Roller-palvelu, jolla voi helposti luoda uusia väriteemoja jQuery Mobilen käyttöliittymäelementeille. (jQuery Mobile 2015.)

3 Kehitysympäristön asennus

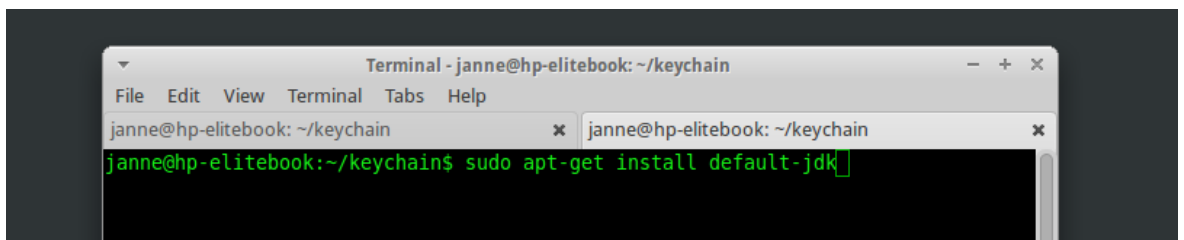
3.1 Oracle Java JDK

Java JDK-paketin asennus Xubuntu Linuxissa suoritettiin komentorivin eli terminaalin kautta, kuten useimmat muistakin tarvittavista asennuksista. Xubuntussa terminaali avataan näppäinyhdistelmällä Windows key + t, jonka jälkeen terminaaliin voidaan syöttää tarvittavat asennuskomennot. Alla kuva terminaalista (Kuva 1).



Kuva 1. Terminaali

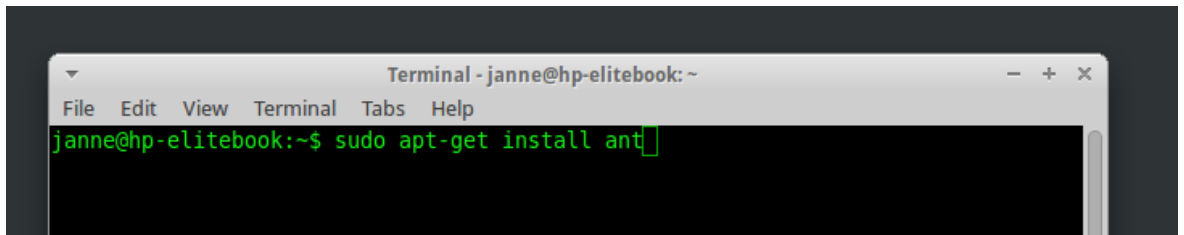
Java JDK-paketin kohdalla komento oli `sudo apt-get install default-jdk` (Higman 17.10.2013). Komennossa `sudo` tarkoittaa sitä, että komento ajetaan pääkäyttäjän oikeuksilla, jolloin voidaan tehdä muutoksia järjestelmään tarvittavalla tasolla. Toinen osa `apt-get install` on paketinhallinnan asennuskomento, joka asentaa määritetyn paketin, tässä tapauksessa paketin nimi on `default-jdk`. Alla kuva komennosta (Kuva 2).



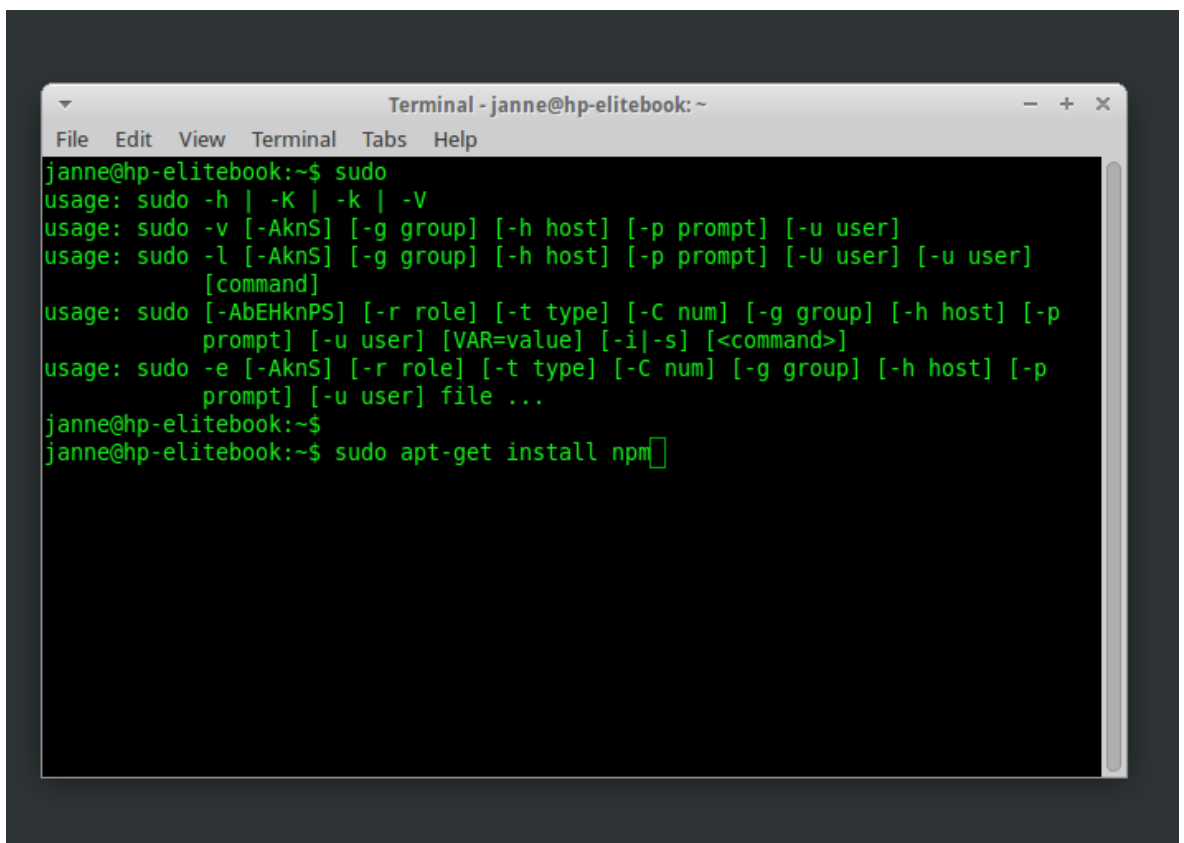
Kuva 2. Java JDK-paketin asennuskomento terminaalissa

3.2 Apache Ant ja Node.js

Apache Ant asennettiin samalla tavalla kuin 3. pääluvun 1. alaluvussa asennettu Java JDK. Terminaaliin syötettävä asennuskomento oli tässä tapauksessa *sudo apt-get install ant*, jossa paketin nimi on siis *ant* (Higman 17.10.2013). Node.js asennettiin samalla tavalla kuin 3. pääluvun 1. alaluvussa asennettu Java JDK. Terminaaliin syötettävä asennuskomento oli tässä tapauksessa *sudo apt-get install npm*, jossa paketin nimi on *npm* (Binstein 21.7.2014). Alla kuvat asennuskomennoista terminaalisissa (Kuva 3, Kuva 4).



Kuva 3. Apache Ant-asennuskomento terminaalisissa



Kuva 4. Node.js-asennuskomento terminaalisissa

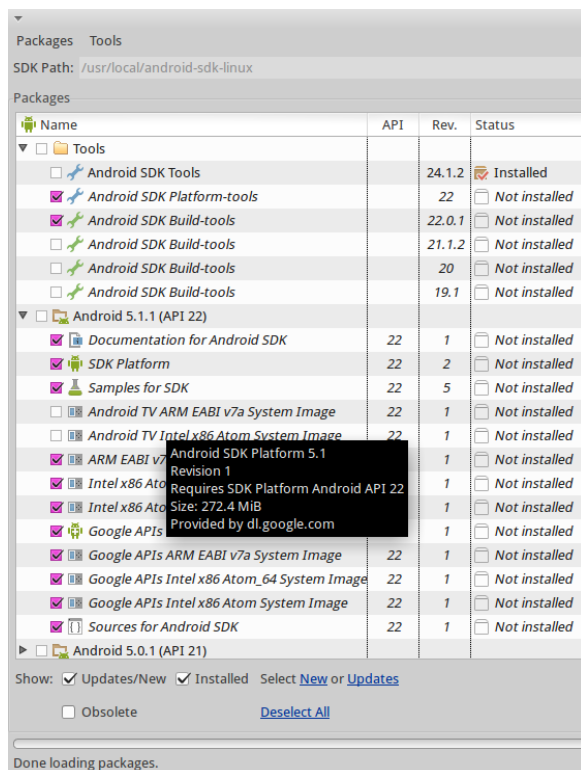
3.3 Android SDK

Androidin kehitystyökalujen asentaminen poikkesi edellisissä kohdissa käsitellystä tavasta, koska se ladattiin erikseen Googlen sivuilta (Google 2015). Linuxille ladataan asen-

nuspaketti tgz-päätteellä, ja koska tämän opinnäytetyön kehitysympäristössä käytettiin Intel XDK-sovellusta editorina, valittiin latauksen tyypiksi *SDK Tools Only*, jolloin mukana ei tullut Android Studio-kehitysohjelmaa, joka ei ollut tarpeellinen.

Android SDK-paketti purettiin Xubuntun *Archive Manager*-ohjelmalla. Tässä tapauksessa paketti purettiin kirjoittajan Xubuntun kotihakemiston juureen, kansioon nimeltä android-sdk-linux. Kansion tiedostopolku voidaan kirjoittaa muodossa ~/android-sdk-linux/, jos halutaan navigoida kansioon terminaalissa komennolla `cd` – koko navigointikomento olisi tällöin `cd ~/android-sdk-linux/`.

Ensin navigoitiin terminaalissa tools-kansioon komennolla `cd ~/android-sdk-linux/tools/`, josta voitiin käynnistää Android SDK-asennusohjelma komennolla `./android`. Tässä vaiheessa asennettiin *Platform-tools*-työkalut, Androidin tarvittavat rajapinnat Android-versioille 4.4.2 ja 5.1.1, sekä Googlen USB-ajuri testilaitetta varten. Kun valinnat oli tehty, hyväksyttiin asennus ja käyttöehdot, jonka jälkeen asennusohjelma asensi valitut komponentit. (Binstein 21.7.2014.) Alla esimerkkikuva *platform-tools*-työkalujen ja rajapinnan 5.1.1 valinnasta (Kuva 5).



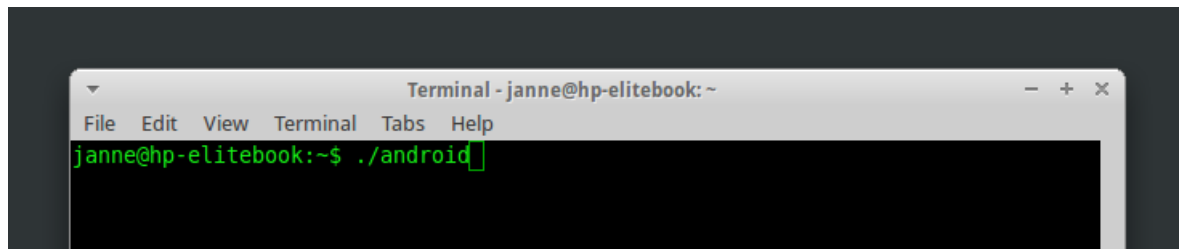
Kuva 5. Valittuina Platform Tools ja rajapinta Android-versiolle 5.1.1

Kun edellisen kohdan asennusprosessi oli valmis, lisättiin *tools*- ja *platform-tools*-kansiot Xubuntun PATH-ympäristömuuttujaan, jotta voidaan käyttää terminaalissa esimerkiksi edellä mainittua *Android*-terminaalikomentoa. Näiden kansioden lisääminen PATH-

ympäristömuuttujaan määrittää käyttöjärjestelmälle, mistä pyydettyä komentoa (ohjelmaa), tässä tapauksessa *Androidia*, oletusarvoisesti etsitään. Tiedostopolkujen lisääminen muuttujaan tapahtui avaamalla *bashrc*-tiedosto terminaalin kautta komennolla *nano ~/.bashrc*. Tämän tiedoston loppuun lisättiin tarvittavat rivit. (Ubuntu Documentation 2015.)

```
export PATH=${PATH}:/android-sdk-linux/tools  
export PATH=${PATH}:/android-sdk-linux/platform-tools
```

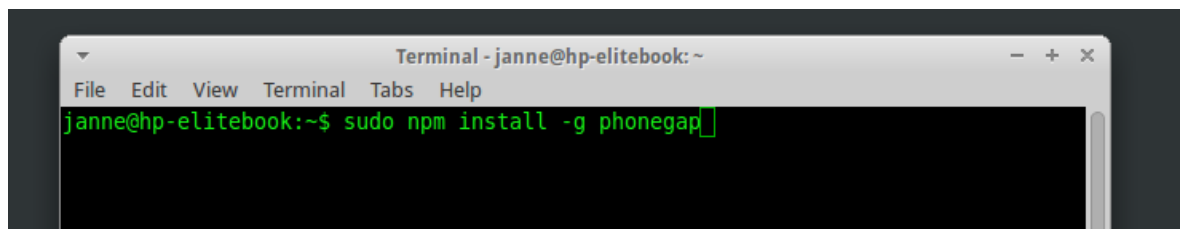
Tämän jälkeen tallennettiin muutokset *bashrc*-tiedostoon. Terminaalissa pystyi ajamaan komennon *Android*. (Ubuntu Documentation 2015.) Android SDK-asennus oli valmis kokonaisuudessaan. Alla kuva *Android*-komennosta (Kuva 6).



Kuva 6. Android-ajokomento terminaalissa

3.4 PhoneGap-sovelluskehys

PhoneGap-sovelluskehiksen komentorivi-version (engl. *command line interface*) asennus tapahtui Xubuntun terminaalin kautta komennolla *sudo npm install -g phonegap*, jossa *npm* viittaa 3. pääluvun 2. alaluvun Node.js-asennukseen, eli PhoneGap-sovelluskehiksen asennuksessa hyödynnettiin sen pakettinhallintaa. Komennon ajamisen jälkeen PhoneGap-sovelluskehiksen asennusprosessi oli valmis. Alla kuva komennosta terminaalissa (Kuva 7).



Kuva 7. PhoneGap-asennuskomento terminaalissa

3.5 Intel XDK

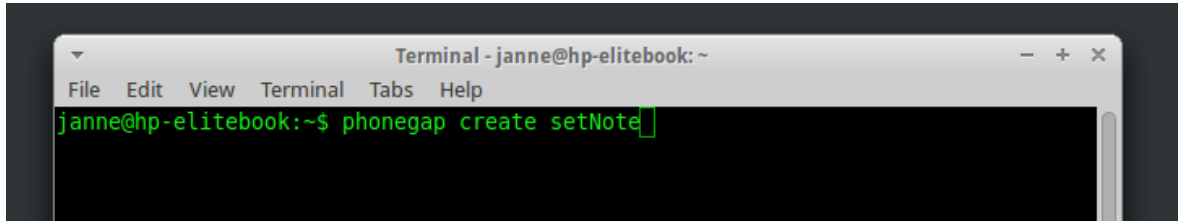
Intel XDK-ohjelmasta ladattiin ensin Linuxille sopiva 64-bittinen asennuspaketti ohjelman sivuilta. Seuraavaksi ladattu paketti purettiin Xubuntun *Archive Manager*-ohjelmalla halut-

tuun kansioon, tässä tapauksessa suoraan kirjoittajan Xubuntun kotihakemiston lataukset-kansioon. Linux-versiossa kansion nimi oli oletuksena `xdk_web_linux64`. Asennus suoritettiin loppuun suorittamalla kansiossa oleva *install.sh*-tiedosto. (Intel 2015a; Intel 2015b.)

4 Sovelluksen kehitys

4.1 PhoneGap-projektin luominen

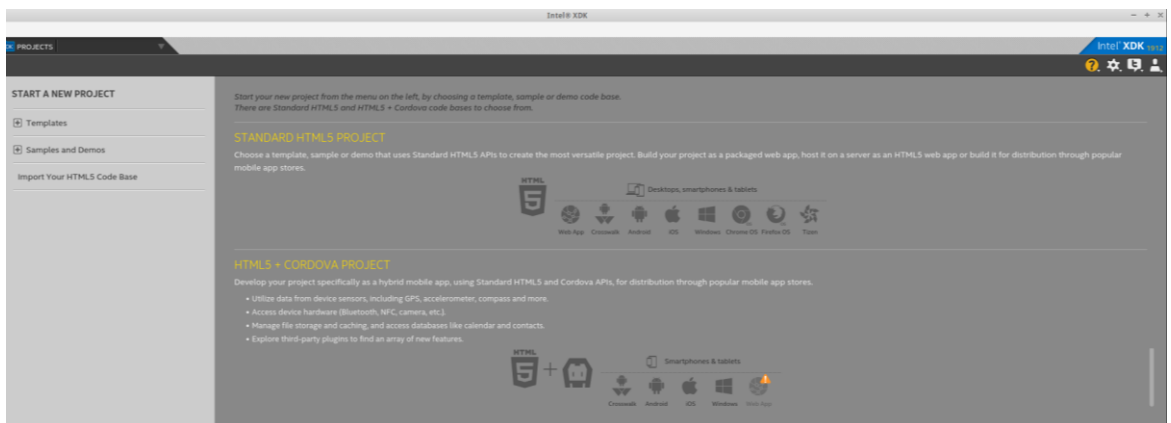
Ennen SetNote-projektin luomista sille tehtiin tyhjä työkansio kirjoittajan kotihakemiston juureen Xubuntussa nimellä Projekti. Seuraavaksi navigoitiin terminaalissa kyseiseen kansioon komennolla `cd ~/Projekti`, johon projekti luotiin PhoneGap-sovelluskehityksen avulla Xubuntun terminaalin kautta komennolla `phonegap create setNote`. (Higman 17.10.2013.) Alla kuva tilanteesta (Kuva 8).



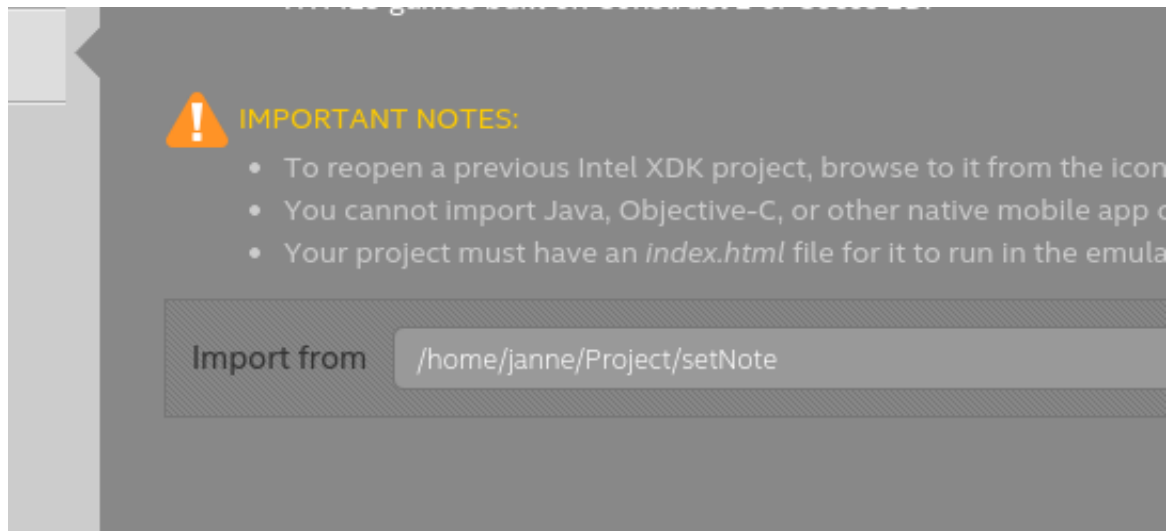
Kuva 8. PhoneGap-projektin luominen terminaalissa

4.2 Projektin lisääminen Intel XDK-ohjelmaan

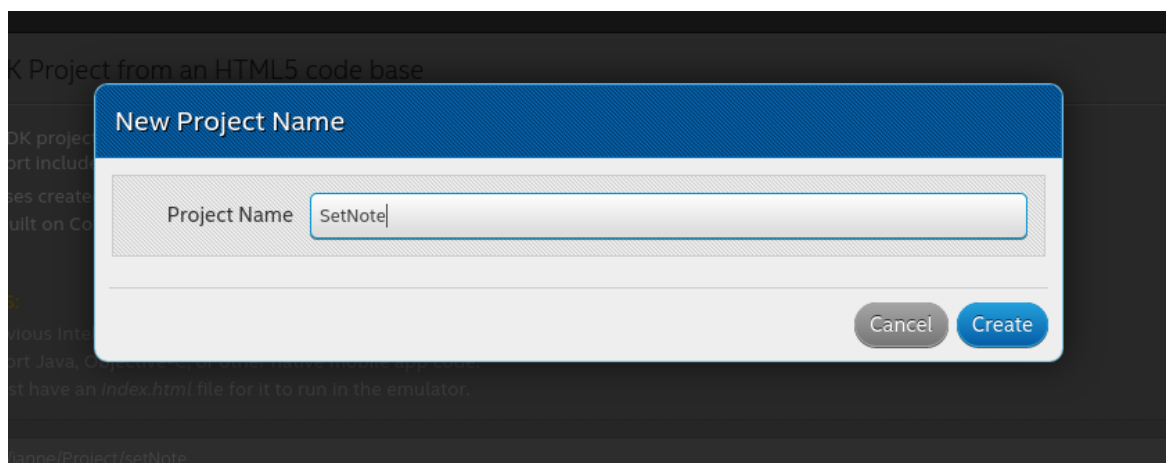
Intel XDK-ohjelman (Kuva 9) käynnistymisen jälkeen SetNote-projekti lisättiin sen projektistaan *Start a new project*-kohdan alta valinnalla *Import Your HTML5 Code Base*, jossa kansion hakemistopolku määritettiin (Kuva 10) ja projektille annettiin työnimi SetNote (Kuva 11). Seuraavaksi *Project Importer results*-ikkunassa määritettiin, että Cordova (PhoneGap)-lisäosia (engl. *plugins*) ei käytetä Intel XDK-ohjelman sisällä (Kuva 12). Tämä sen vuoksi, että kirjoittajan oman kokemuksen mukaan ne eivät toimi Intel XDK-ohjelman virtuaalisessa laite-emulaattorissa (Kuva 13) oikein, vaan aiheuttavat jatkuvia virheilmoituksia, silloin kun niitä käytetään. Eteenpäin jatkettiin klikkaamalla *Continue*-painiketta, jolloin Intel XDK-lisäsi onnistuneesti projektin projektistaansa (Kuva 14).



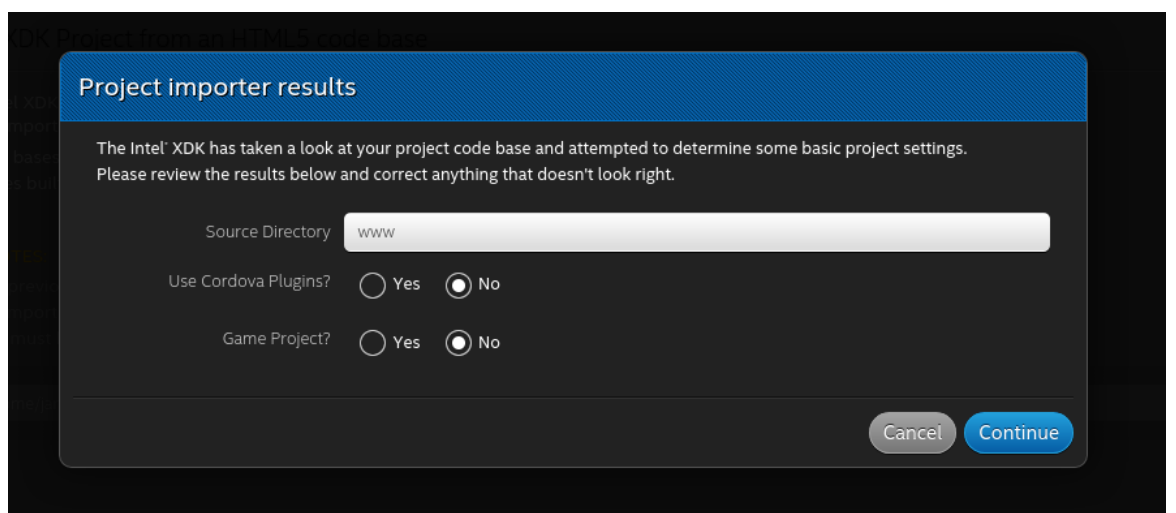
Kuva 9. Intel XDK-ohjelma



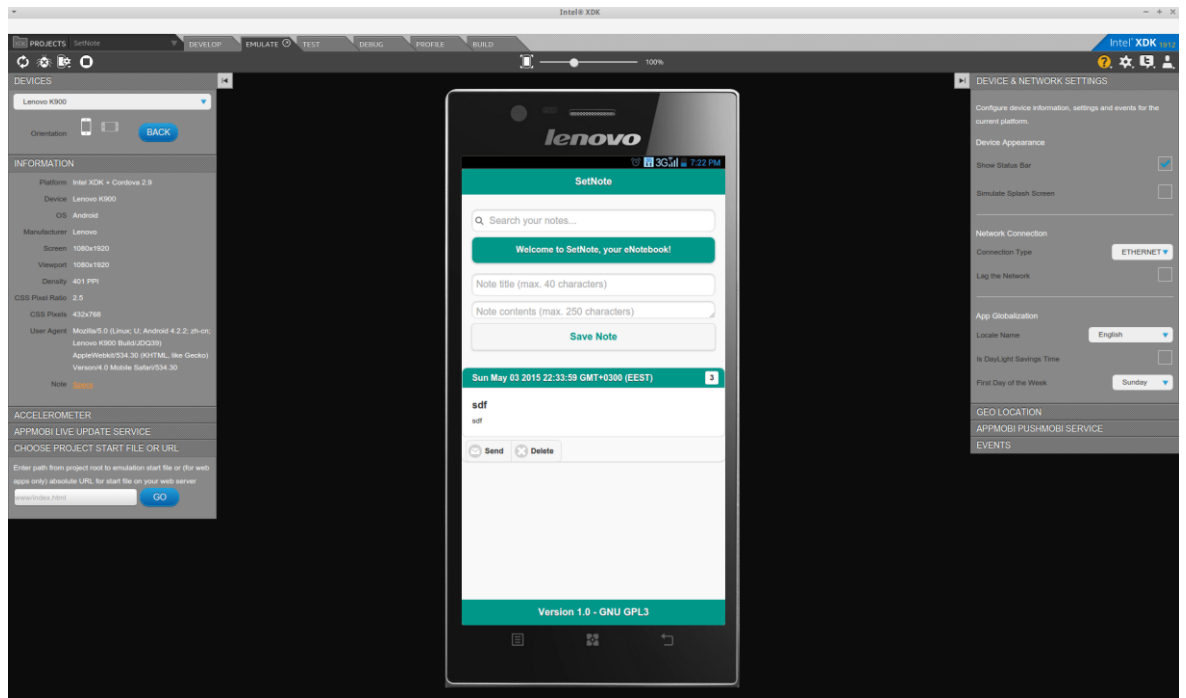
Kuva 10. Projektin juuripolun määrittäminen Intel XDK-ohjelmassa



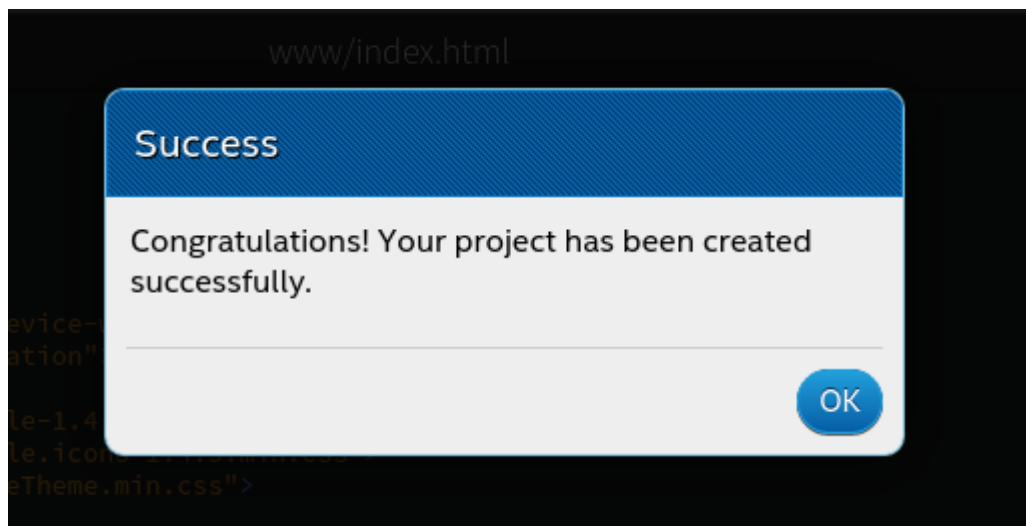
Kuva 11. Asetetaan projektille työnimi Intel XDK-ohjelmassa



Kuva 12. Poistetaan Cordova-lisäkkeet käytöstä



Kuva 13. Intel XDK-ohjelman virtuaalinen laite-emulaattori Androidille



Kuva 14. Projekti on lisätty onnistuneesti listalle

4.3 Testilaitteen liittäminen

Testilaitte eli OnePlus One-puhelin, jota käsiteltiin 2. pääluvun 1. alaluvussa, asetettiin ensin kehittäjätilaan ottamalla puhelimen tietoturvalistasta käyttöön *Unknown sources*-asetus, joka sallii sovellusten asentamisen muualta kuin Google Play-sovelluskaupasta eli tässä tapauksessa kannettavalta tietokoneelta. Kehittäjävalikosta otettiin vielä lisäksi käyttöön asetus *Android debugging*, joka sallii puhelimen ottaa yhteys tietokoneeseen virheenkorjaustilassa USB-yhteyden yli. Molemmat asetukset ovat välttämättömiä sovelluksen asentamiseksi koneelta suoraan puhelimeen USB-kaapelin välityksellä.

Seuraavaksi OnePlus One-puhelin liitettiin USB-kaapelilla kannettavaan tietokoneeseen ja näin myös Xubuntu-käyttöjärjestelmään. Tämän jälkeen Xubuntu alkoi etsiä ja löysi 3. pääluvun 4. alaluvussa asennetun Google USB-ajurin, jonka se asensi OnePlus One-puhelimen käyttöön käyttöjärjestelmässä. Ajurin asennuksen jälkeen puhelin pyysi hyväksymään tietokoneen tarjoaman RSA-avaimen, jotta turvallinen yhteys voitiin muodostaa. Tämän hyväksymisen jälkeen puhelin oli valmis testikäyttöön.

Asennuksen lopuksi SetNote-sovelluksen asentamista testipuhelimeen testattiin Xubuntun terminaalista USB-yhteyden yli. Ensin navigoitiin terminaalissa projektin kansioon komennolla `cd ~/Projekti/setNote/`. Tämän jälkeen ajettiin komento *phonegap run android*, joka asentaa sovelluksen testilaitteeseen, jos sellainen on kytkettynä ja se tunnistetaan *Android Debug Bridge* -komentorivityökalun kautta. Komento lisää myös automaattisesti Android-alustan (engl. *platform*) projektiin – tämä alusta mahdollistaa sovelluksen rakentamisen ja asentamisen Android-laitteisiin. Puhelin tunnistettiin ja sovellus asentui siihen USB-kaapelin yli ongelmitta. Sovelluksessa ei tässä vaiheessa ollut vielä sisältöä PhoneGap-sovelluskehiksen generoiman pohjan lisäksi.

4.4 Sovelluksen rakenne

PhoneGap-sovelluskehiksen mukaisesti SetNote-sovellus koostuu internet-sivujen tapaan HTML-, CSS- ja JavaScript-tiedostoista, jotka PhoneGap pakatoi omien tiedostojensa ohella asennettavaan muotoon Android-laitteille. Sovelluksen rakenne vastaa internet-sivujen rakennetta myös siltä osin, että CSS- ja JavaScript-tiedostot linkitetään HTML-sivuihin niiden head-osassa, jotta sivut voivat hyödyntää niitä.

Sovelluksen kansiorakenne setNote-projektikansiossa on PhoneGap-sovelluskehiksen määrittämä - juuressa sijaitsevat alikansiot hooks, platforms, plugins ja www. Juuressa sijaitsee myös tiedosto config.xml, joka sisältää sovelluksen tietoja sekä PhoneGap-määrittämiä sovelluksen tarvitsemien käyttöoikeuksien osalta.

Alikansioista www on sovelluksen kehityksen kannalta oleellisin, koska se sisältää aina sovelluksen kehitysosan eli HTML-, CSS- ja JavaScript-tiedostot. Alikansio sisältää omat alikansionsa css, img, js ja res. Kansioista css sisältää sovelluksen setnote.css-tyylitiedoston, img sovelluksen logon, js sovelluksen setnote-main.js JavaScript-tiedoston ja res sovelluksen kuvakkeen eli ikonin *icon*-alikansiossa sekä *screen*-alikansiossa sovelluksen käynnistyskuvan (engl. *splash screen*), joka näytetään sen käynnistyessä.

Yllä mainitussa `www`-alikansiossa sijaitsevat myös sovelluksen HTML-tiedostot. Koska SeNote on niin kutsuttu yhden sivun sovellus (engl. *single-page application*), HTML-tiedostoja on vain yksi, `index.html`. Yhdelle HTML-sivulle mahdutettuna sovellus toimii nopeammin ja kaikki sen osat voidaan ladata kerralla, jolloin sovellus toimii sulavammin ja sen käyttökokemus on lähempänä natiivia Android-sovellusta. Sovelluksen lähdekoodin rakennetta käsitellään tämän 4. pääluvun 9. alaluvussa.

4.5 jQuery- ja jQuery Mobilen lisäys

Ensimmäiseksi ladattiin sekä jQuery että jQuery Mobile niiden omilta sivuilta. Latauspaketit olivat yleissopivia, eikä Linuxille tarvinnut ladata omia paketteja. Ladatut tiedostot purettiin Xubuntun *Archive Manager*-ohjelmalla kirjoittajan omaan Xubuntun kotihakemistoon, kansioon Downloads. (jQuery 2015; jQuery Mobile 2015).

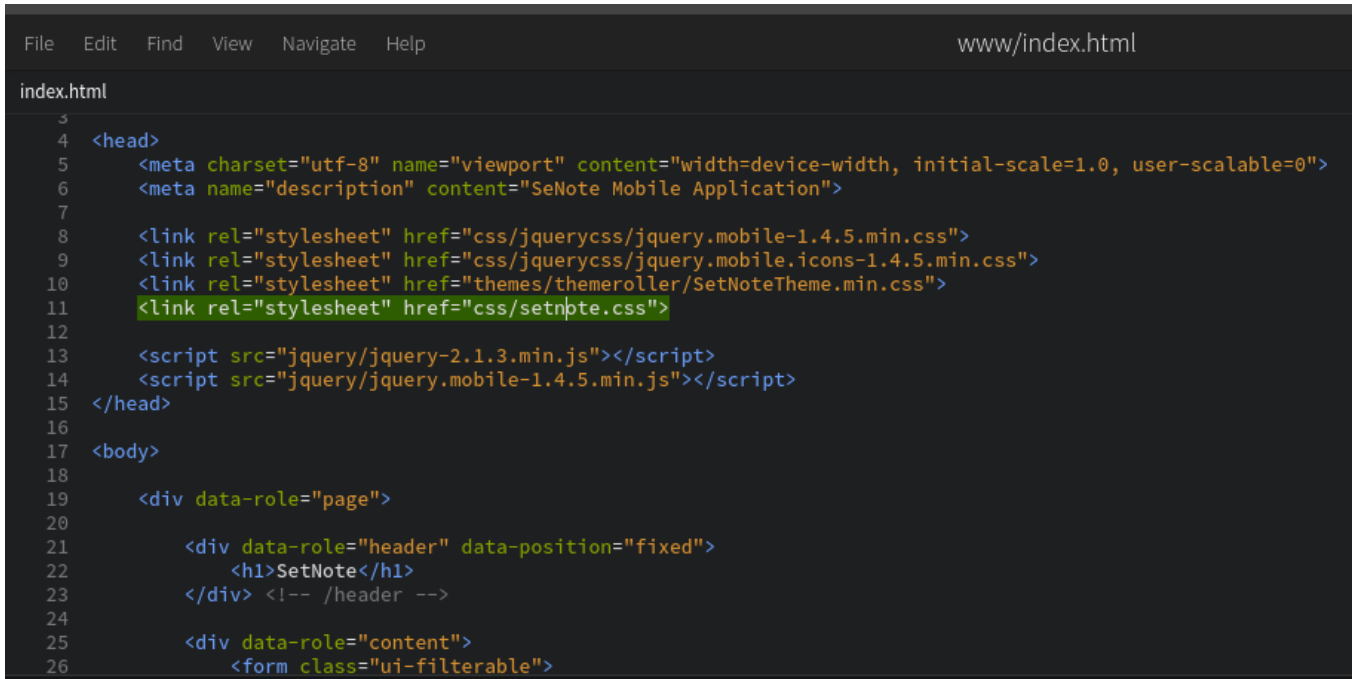
Näiden kansioden sisällöstä pystyi oman olemassa olevan tiedon turvin valitsemaan tarvittavat osat projektia varten. Tässä tapauksessa ne olivat JavaScript-tiedostot `jquery.mobile-1.4.5.min.js` ja `jquery-2.1.3.min.js` sekä tyyli-tiedostot `jquery.mobile.icons-1.4.5.min.css` ja `jquery.mobile-1.4.5.min.css`. Yleisesti voidaan todeta, että näiden neljän tiedoston avulla projektiin saatiin kaikki jQuery- ja jQuery Mobile-toiminnallisuus, jota tarvittiin. Vaikka projekti tarvitsikin lisäkkeitä kuten edellä mainitut, oli silti syytä minimoida niiden projektiin mukanaan tuomien tiedostojen määrä.

Edelleen turvautuen omaan tietoon asennuksen kulusta, oli seuraavaksi vuorossa jQuery:n ja jQuery Mobilen liittäminen SetNote-projektiin, joka käytännössä tarkoitti sitä, että ne täytyi ensin lisätä projektin kansiorakenteeseen ja sen jälkeen linkittää projektin `index.html`-tiedostoon, jotta ne voitiin ladata käyntiin projektia ajettaessa ja että niitä voitiin kutsua tarpeen mukaan.

Tämän 4. pääluvun 4. alaluvussa käsiteltyyn sovelluksen kansiorakenteeseen viitaten, edellä mainittujen jQuery- ja jQuery Mobile-komponenttien lisäys sijoittui vain ja ainoastaan `www`-kansioon, jossa muukin projektin varsinainen muokattava sisältö sijaitsi.

JavaScript-tiedostot `jquery.mobile-1.4.5.min.js` ja `jquery-2.1.3.min.js` saivat `www`-kansioon oman alikansionsa nimellä jQuery. Tyyli-tiedostot `jquery.mobile.icons-1.4.5.min.css` ja `jquery.mobile-1.4.5.min.css` siirrettiin `www`-kansion alikansion `css:n` sisälle, jonka sisällä ne vielä kertaalleen siirrettiin omaan alikansioonsa nimellä `jquerycss`. Käytännössä kansioden valinta ei tässä tapauksessa liittynyt erityisesti mihinkään standardeihin, vaan kyseessä oli projektin kehittäjän oma mielipide hyvästä kansiorakenteesta.

Seuraavassa kuva siitä miltä index.html-tiedoston head-osio näytti kun linkitykset oli tehty (Kuva 15).

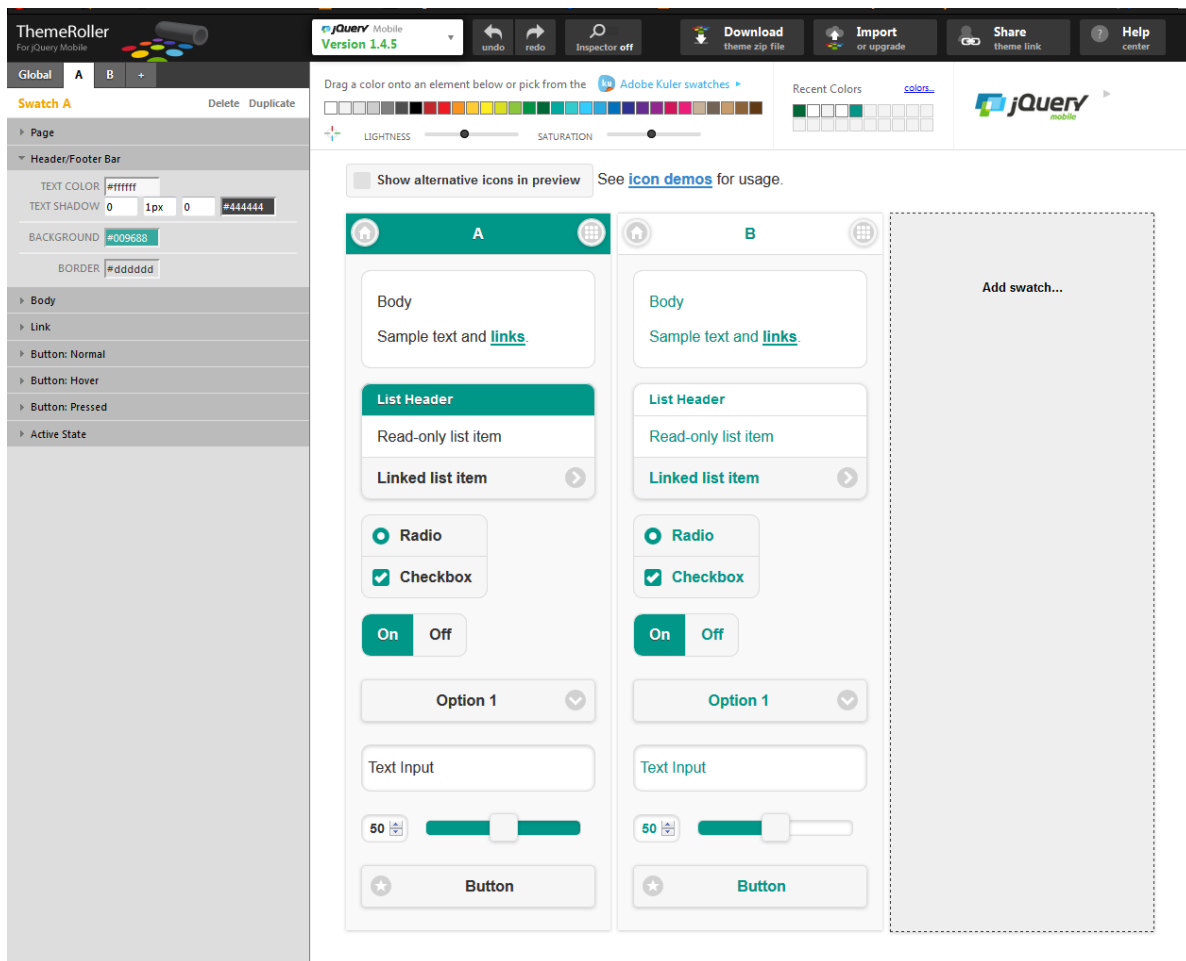


```
3
4 <head>
5   <meta charset="utf-8" name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0">
6   <meta name="description" content="SeNote Mobile Application">
7
8   <link rel="stylesheet" href="css/jquerycss/jquery.mobile-1.4.5.min.css">
9   <link rel="stylesheet" href="css/jquerycss/jquery.mobile.icons-1.4.5.min.css">
10  <link rel="stylesheet" href="themes/themeroller/SetNoteTheme.min.css">
11  <link rel="stylesheet" href="css/setnpte.css">
12
13  <script src="jquery/jquery-2.1.3.min.js"></script>
14  <script src="jquery/jquery.mobile-1.4.5.min.js"></script>
15 </head>
16
17 <body>
18   <div data-role="page">
19     <div data-role="header" data-position="fixed">
20       <h1>SetNote</h1>
21     </div> <!-- /header -->
22
23     <div data-role="content">
24       <form class="ui-filterable">
```

Kuva 15. Index.html kun jQuery ja jQuery Mobile oli lisätty sen head-osioon

4.6 ThemeRoller-teeman lisäys

ThemeRoller-teeman lisäys jQuery Mobilen päälle projektiin oli käytännössä suurin syy miksi jQuery Mobile valittiin vastaamaan SetNote-sovelluksen käyttöliittymästä ja ulkoasusta. ThemeRoller on erittäin helppokäyttöinen web-pohjainen työkalu, jolla voi nopeasti määrittää haluamansa määrän väriteemoja jQuery Mobilen käyttöliittymäelementeille ja ladata valmiit teemat suoraan ThemeRollerin sivuilta. (ThemeRoller 2015.) Alla kuva SetNote-projektin teemasta ThemeRollerissa (Kuva 16).



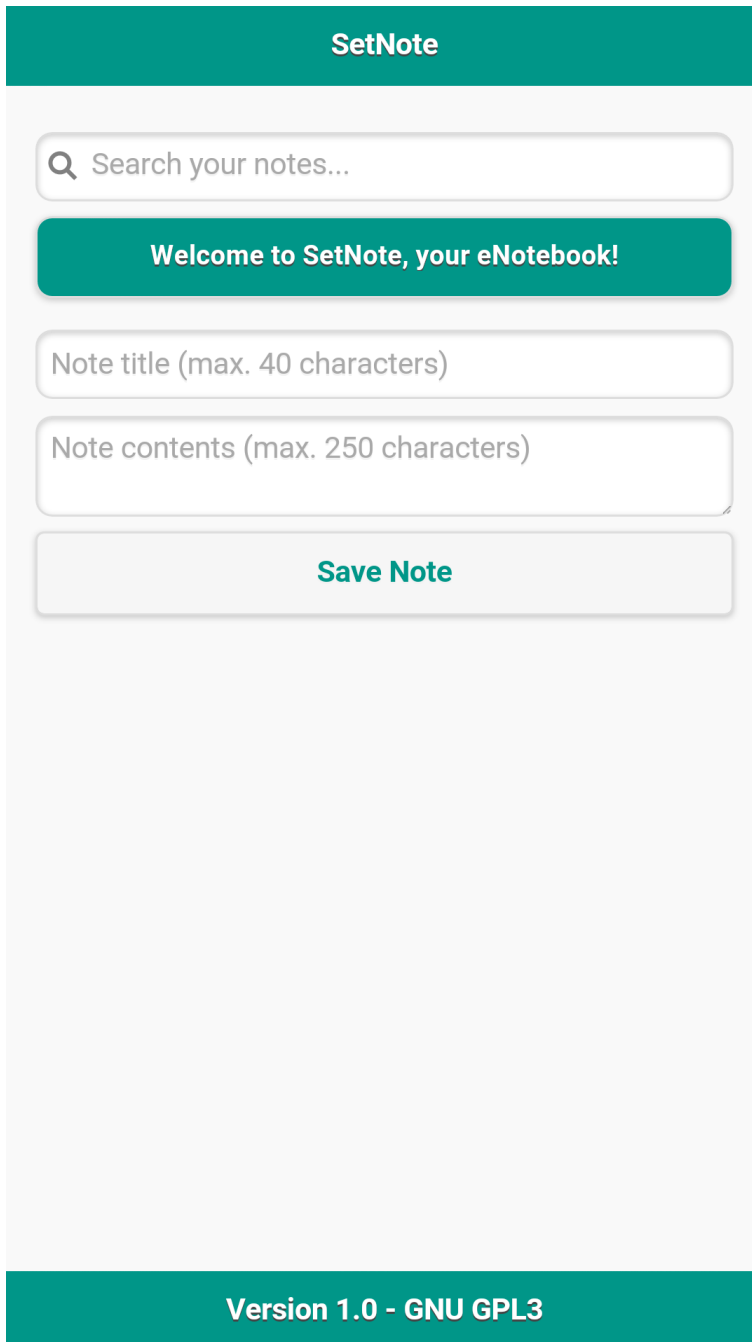
Kuva 16. SetNote-sovelluksen teema ThemeRollerin sivujen generaattorissa

ThemeRollerin teema ladattiin kuvan 16 yläosassa näkyvän Download-painikkeen kautta Downloads-kansioon Xubuntun kotihakemistossa. Oman tiedon pohjalta oli myös tiedossa, mitä ThemeRollerin latauspaketin sisällöstä otettiin mukaan. Näin projektin www-kansioon sijoitettiin themes-kansio, jonka alikansioksi määritettiin themeroller-kansio, johon SetNoteThemeMin.css-tyylitiedosto sijoitettiin images-kansion kanssa, joka taas sisälsi kaikki ikonit tai kuvakkeet tyylejä varten. ThemeRollerin linkitys projektiin tapahtui samalla tavalla kuin jQuery:n ja jQuery Mobilen eli index.html:n kautta, minkä voi nähdä kuvassa 15, jossa ThemeRoller on jo linkitettyä.

4.7 Sovelluksen käyttöliittymä

Sovelluksen käyttöliittymän ajatuksena oli olla mahdollisimman minimalistinen ja noudattaa Googlen Android 5.0 ja uudempien versioiden tiettyä väriteemaa, jonka pääosassa ovat turkoosin vihreä ja valkoinen. Idean jalostus jatkui vielä ThemeRollerissa eteenpäin, kuten kuvasta 16 voi nähdä.

Käyttöliittymän minimalistisuus ja yksinkertaistettu ulkoasu olivat myös helpompia toteuttaa käytännössä, koska animaatioihin tai muuhun dynaamiseen koristeluun ei tarvinnut käyttää aikaa. Alla kuva SetNote-sovelluksen pääsivusta (Kuva 17).



SetNote

Q Search your notes...

Welcome to SetNote, your eNotebook!

Note title (max. 40 characters)

Note contents (max. 250 characters)

Save Note

Version 1.0 - GNU GPL3

Kuva 17. SetNote-sovelluksen pääsivu

Sovelluksen ominaisuuksia työstettäessä oli myös otettava huomioon niiden asettamat vaatimukset sovelluksen toiminnallisuudelle ja funktiolle ja verrattava näitä tarpeita ideaalin ulkoasun ja käyttöliittymän määrittämiseen tasapainoisen lopputuloksen aikaansaamiseksi.

Seuraavassa kuva sovelluksesta toiminnassa, jossa käyttöliittymään on lisätty muistilappu-elementtejä (Kuva 18).

The screenshot displays the 'SetNote' application interface. At the top is a teal header with the text 'SetNote'. Below the header is a search bar with a magnifying glass icon and the placeholder text 'Search your notes...'. A teal button with white text reads 'Welcome to SetNote, your eNotebook!'. Below this are two text input fields: 'Note title (max. 40 characters)' and 'Note contents (max. 250 characters)'. A grey button with black text says 'Save Note'. Below the form is a list of three notes, each with a teal header bar containing the timestamp and a small white box with a number. The first note has a title 'PC' and content 'Update Windows', with 'Send' and 'Delete' buttons below it. The second note has a title 'Lunch?' and content 'Call to confirm', also with 'Send' and 'Delete' buttons. The third note is partially visible at the bottom.

SetNote

Q Search your notes...

Welcome to SetNote, your eNotebook!

Note title (max. 40 characters)

Note contents (max. 250 characters)

Save Note

Sun May 03 2015 17:11:21 GMT+0300 (EEST) 3

PC
Update Windows

Send Delete

Sun May 03 2015 17:10:35 GMT+0300 (EEST) 2

Lunch?
Call to confirm

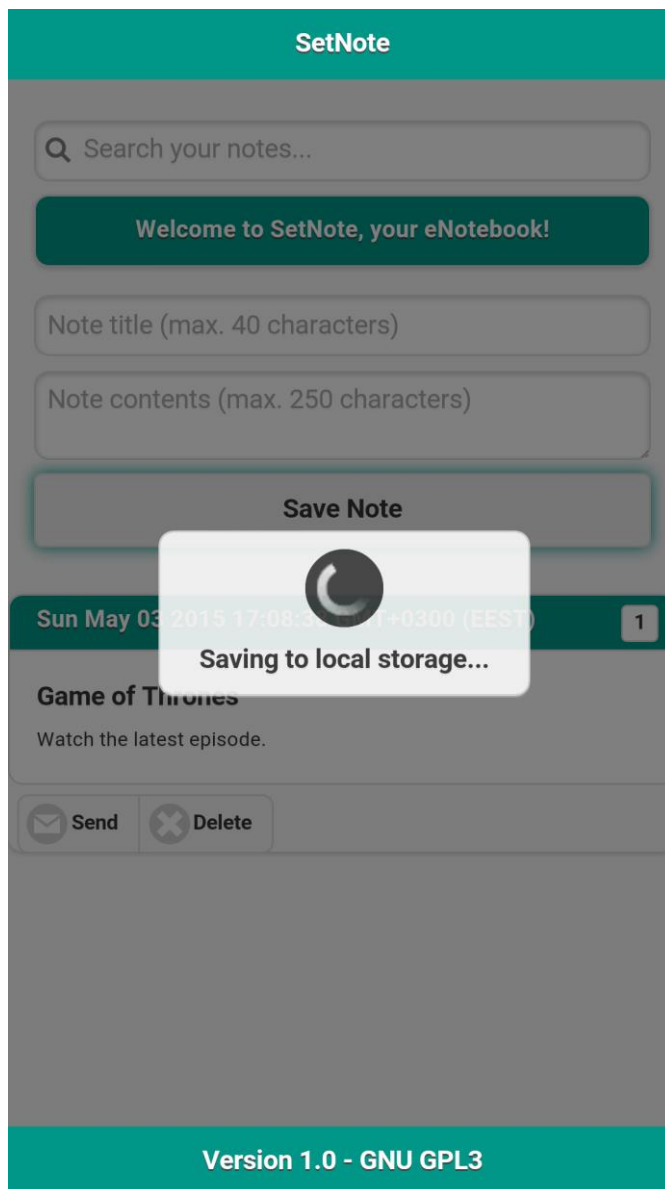
Send Delete

Sun May 03 2015 17:08:38 GMT+0300 (EEST) 1

Kuva 18. Muistilaput lisätty käyttöliittymään

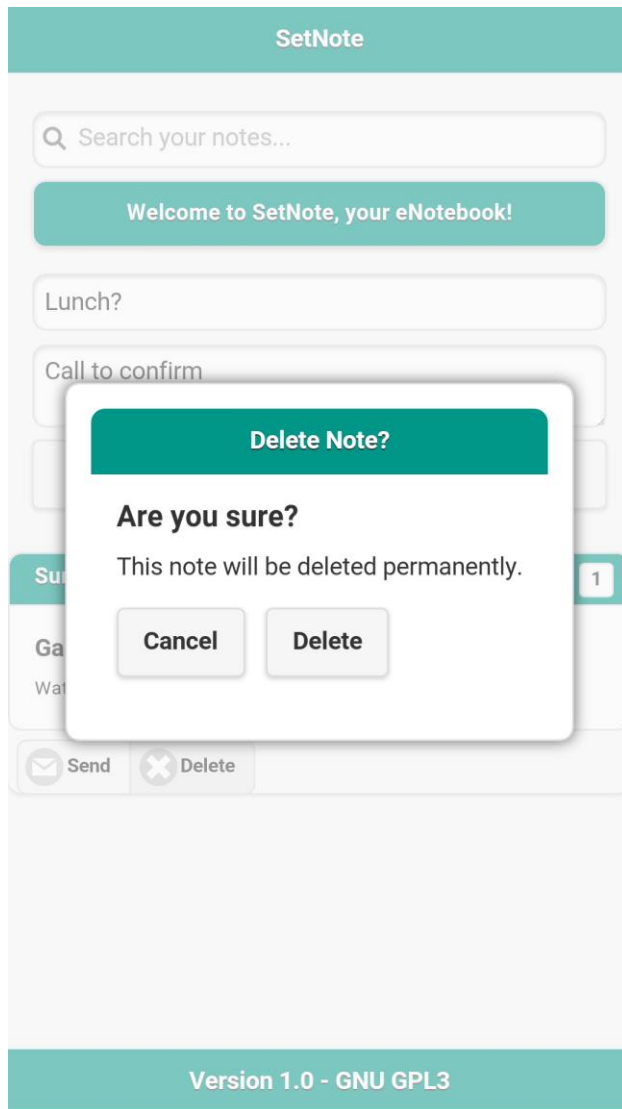
Käyttöliittymän selkeyden ja hienon ulkoasun ohella oli tietenkin myös tarpeen pohtia sovelluksen käytettävyyttä. Erilliset painikkeet (Kuva 18) jokaiselle muistilapulle tuovat vaihtelua sovelluksen virtaa ja antavat käyttäjälle mahdollisuuden keskittyä yhteen asiaan kerrallaan. Myös muistilappujen numerointi ja järjestys oli mietinnän kohteena, ja lopulta siinä päädyttiin ratkaisuun, jossa uusin muistilappu on suurimmalla numerolla aina ensimmäisenä. Ratkaisu ei ollut erityisen omaperäinen, mutta sovelluksen nykyisen käyttöliittymän kanssa se toimii erinomaisesti. Alla kuva sovelluksen tallennusanimaatiosta, joka myös

poistaa kohdistuksen sovelluksen muulta osalta ja himmentää näytön tallennuksen ajaksi, joka viestittää käyttäjälle hyvin sen, että sovellus ei ole tällöin käytettävissä (Kuva 19).



Kuva 19. SetNote-sovelluksen tallennuksen animaatio ja himmennys

Seuraavassa kuva vielä sovelluksen muistilapun poiston vahvistusdialogista (Kuva 20), jossa käyttäjää muistutetaan vielä erikseen siitä, että poisto on peruuttamaton. Vahinkopainallusten aikaansaama tuhoa voidaan siis ehkäistä käyttämällä sovelluksessa varmennusdialogeja suoran poiston sijaan.



Kuva 20. Sovelluksen vahvistuspyyntö poistolle

4.8 Sovelluksen ikoni

Sovelluksen ikoni tai pikakuvake on oleellinen, joskin pieni asia sen ulkoasun kannalta. Tämän projektin tapauksessa ratkaisu löytyi täysin vapaalla lisenssillä jaettavasta grafiikasta, jota tarjoaa esimerkiksi Openclipart, jonka valikoimista SetNoten ikoni löytyi. Ikonin väri muutettiin GIMP-kuvankäsittelyohjelmassa ja siihen lisättiin teksti SetNote. Alla kuva lopullisesta versiosta (Kuva 21). Tarvittavat koot saatiin suoraan Roman Nurikin Launcher Icon Generator-sivuston kautta, jossa valmis ikoni kopioitiin eri kokoihin valmiiksi projektia varten. (Openclipart 2015.) (Github 2015.)



Kuva 21. Ikoni

4.9 Sovelluksen lähdekoodin kommentointi

Tässä luvussa käydään läpi sovelluksen lähdekoodi siltä osin, kuin sitä on projektin aikana tuotettu. Valmiina erilaisissa komponenteissa olevaa lähdekoodia ei käsitellä. Käsiteltäviä tiedostoja on kolme; index.html, setnote.css ja setnote-main.js. Ensimmäiseksi vuorossa on index.html (Kuva 22).

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8" name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0">
6   <meta name="description" content="SetNote Mobile Application">
7
8   <link rel="stylesheet" href="css/jquerycss/jquery.mobile-1.4.5.min.css">
9   <link rel="stylesheet" href="css/jquerycss/jquery.mobile.icons-1.4.5.min.css">
10  <link rel="stylesheet" href="themes/themeroller/SetNoteTheme.min.css">
11  <link rel="stylesheet" href="css/setnote.css">
12
13  <script src="jquery/jquery-2.1.3.min.js"></script>
14  <script src="jquery/jquery.mobile-1.4.5.min.js"></script>
15 </head>
16
17 <body>
18
19   <div data-role="page">
20
21     <div data-role="header" data-position="fixed">
22       <h1>SetNote</h1>
23     </div> <!-- /header -->
24
25     <div data-role="content">
26       <form class="ui-filterable">
27         <input id="setnotesearch" data-type="search" placeholder="Search your notes...">
28       </form>
29
30       <ul data-role="listview" data-inset="true">
31         <li data-role="list-divider" class="welcome" data-theme="a">
32           <h1>Welcome to SetNote, your eNotebook!</h1>
33         </li>
34       </ul>
35
36       <input id="title" type="text" maxlength="40" placeholder="Note title (max. 40 characters)">
37       <textarea id="content" type="text" maxlength="250" placeholder="Note contents (max. 250 characters)"></textarea>
38       <input type="submit" class="save button" data-theme="b" value="Save Note">
39       <br>
40
41       <div id="hold">
42         <ul data-role="listview" id="storage-container" data-filter="true" data-input="#setnotesearch">
43         </ul>
44       </div>
45
46       <div id="hidden">
47
48         <div data-role="popup" id="confirmDialog" class="ui-content" data-transition="slideup" data-dismissible="false" data-overlay-theme="a" data-theme="a">
49           <div data-role="header" data-theme="a">
50             <h1>Delete Note?</h1>
51           </div>
52           <div role="main" class="ui-content">
53             <h3 class="ui-title">Are you sure?</h3>
54             <p>This note will be deleted permanently.</p>
55             <a href="#" class="cancelthis ui-btn ui-corner-all ui-shadow ui-btn-a" data-rel="back">Cancel</a>
56             <a href="#" class="deletethis ui-btn ui-corner-all ui-shadow ui-btn-a" data-rel="back">Delete</a>
57           </div>
58         </div>
59
60         <div data-role="popup" id="emptyFields" class="ui-content" data-transition="slideup" data-overlay-theme="a" data-theme="a">
61           <div data-role="header" data-theme="a">
62             <h1>Missing Details</h1>
63           </div>
64           <div role="main" class="ui-content">
65             <h4 class="ui-title">Please fill in the required fields for your note.</h4>
66             <a href="#" class="ui-btn ui-corner-all ui-shadow ui-btn-a" data-rel="back">OK</a>
67           </div>
68         </div>
69
70         <div data-role="popup" id="emptyTitle" class="ui-content" data-transition="slideup" data-overlay-theme="a" data-theme="a">
71           <div data-role="header" data-theme="a">
72             <h1>Missing Details</h1>
73           </div>
74           <div role="main" class="ui-content">
75             <h4 class="ui-title">Please enter a descriptive title for your note.</h4>
76             <a href="#" class="ui-btn ui-corner-all ui-shadow ui-btn-a" data-rel="back">OK</a>
77           </div>
78         </div>
79
80         <div data-role="popup" id="emptyContent" class="ui-content" data-transition="slideup" data-overlay-theme="a" data-theme="a">
81           <div data-role="header" data-theme="a">
82             <h1>Missing Details</h1>
83           </div>
84           <div role="main" class="ui-content">
85             <h4 class="ui-title">Please add some content for your note.</h4>
86             <a href="#" class="ui-btn ui-corner-all ui-shadow ui-btn-a" data-rel="back">OK</a>
87           </div>
88         </div>
89       </div> <!-- /hidden -->
90     </div> <!-- /content -->
91
92     <div data-role="footer" data-position="fixed">
93       <h4>Version 1.0 - GNU GPL3</h4>
94     </div> <!-- /footer -->
95     <script src="js/setnote-main.js"></script>
96   </div> <!-- /page -->
97 </body>
98
99 </html>
100
101
```

Kuva 22. Index.html

Viitaten kuvaan 22, rivit 1-15 sisältävät HTML-määrittelyn ja sivuston linkitykset head-osiossa, jos myös aiemmin linkitetty jQuery ja jQuery Mobile ovat. Rivit 17–98 sisältävät sivun body-osion, joka kattaa myös sen sisällä olevat elementit, kuten jQuery Mobilen header- ja footer-elementit, jotka näkyvät sovelluksessa palkkeina ylhäällä ja alhaalla. Sivu sisältää riveillä 36–38 kentät, joihin muistilapun tiedot syötetään sovelluksessa. Sivulla näkymättömissä olevia osia ovat esimerkiksi rivien 46–90 popup-viestien määrittelyt, jotka eivät ole suoraan käyttäjän nähtävillä, vaan ne ilmaantuvat vasta jotakin nappia painettaessa.

Setnote.css-tyylitiedosto (Kuva 23). Tyylitiedosto sisältää määrittelyjä edellisen index.html sivun elementeille. Div-elementti rivillä 1 on piilotettu näkyvistä rivin 2 tyylimäärittelyllä, jonka vuoksi sen omaava elementti on index.html:ssä näkymättömissä. Riveillä 26–40 ovat määrittelyt, joilla tallennuksen aikana tummennetaan sovelluksen näkymä.

```
1  #hidden {
2      display: none;
3  }
4
5  .welcome h1 {
6      line-height: 110%;
7      text-align: center;
8  }
9
10 div[data-role="header"] {
11     border: none;
12 }
13
14 div[data-role="header"] [data-role="controlgroup"] {
15     margin: 0;
16 }
17
18 div[data-role="footer"] {
19     border: none;
20 }
21
22 div[data-role="footer"] [data-role="controlgroup"] {
23     margin: 0;
24 }
25
26 .dark-overlay {
27     width: 100%;
28     height: 100%;
29     top: 0;
30     padding: 0;
31     margin: 0;
32     background: rgba(0, 0, 0, 0.5);
33     display: none;
34     position: fixed;
35     z-index: 100;
36 }
37
38 .ui-loading .dark-overlay {
39     display: block;
40 }
```

Kuva 23. Setnote.css

Viimeiseksi käsitellään sovelluksen tärkein lähdekoodi eli JavaScript, joka vastaa sovelluksen toiminnallisuudesta (Kuva 24). Riveillä 1-2 määritellään muuttuja `noteID`, joka vastaa siitä, että jokaisella muistilapulla on oma numeronsa. Rivit 4-20 pitävät sisällään tiedon hakemisen HTML5 local storage-säilöstä aina kun sovellus käynnistyy. Tässä myös lajitellaan haetut muistiot niiden numeroiden mukaan sovelluksen etusivulle. Riveillä 22–56 käsitellään ensin käyttäjän syöttämien tietojen hakeminen kun tämä painaa Save Note-painiketta tallentaakseen muistilapun. Tämän jälkeen tiedot viedään eteenpäin ja niiden pohjalta luodaan sovelluksen etusivulle muistilappu-elementti, joka pitää sisällään kyseiset tiedot. Tämän jälkeen tiedot tallennetaan vielä HTML5 local storage-varastoon, jotta ne eivät katoa sovelluksen muistista. Lopuksi ajetaan vielä tallennusanimaatio, jotta voidaan varmistua siitä, että sovellus ehtii tallentaa tiedon ilman, että käyttäjä pääsisi vahingossa katkaisemaan oman tallennuksensa. Riveillä 58–63 on sähköpostin lähetys eli muistilapun lähettäminen toiselle henkilölle – toiminto käyttää mobiililaitteen omaa sähköpostiohjelmaa eli avaa sen pyydettyssä. Riveillä 65–74 käsitellään muistion poistaminen kokonaan sovelluksen muistista. Tässä tapauksessa avataan myös dialogi-ikkuna, joka vahvistaa käyttäjältä, haluaako tämä todella poistaa valitun muistilapun.

```

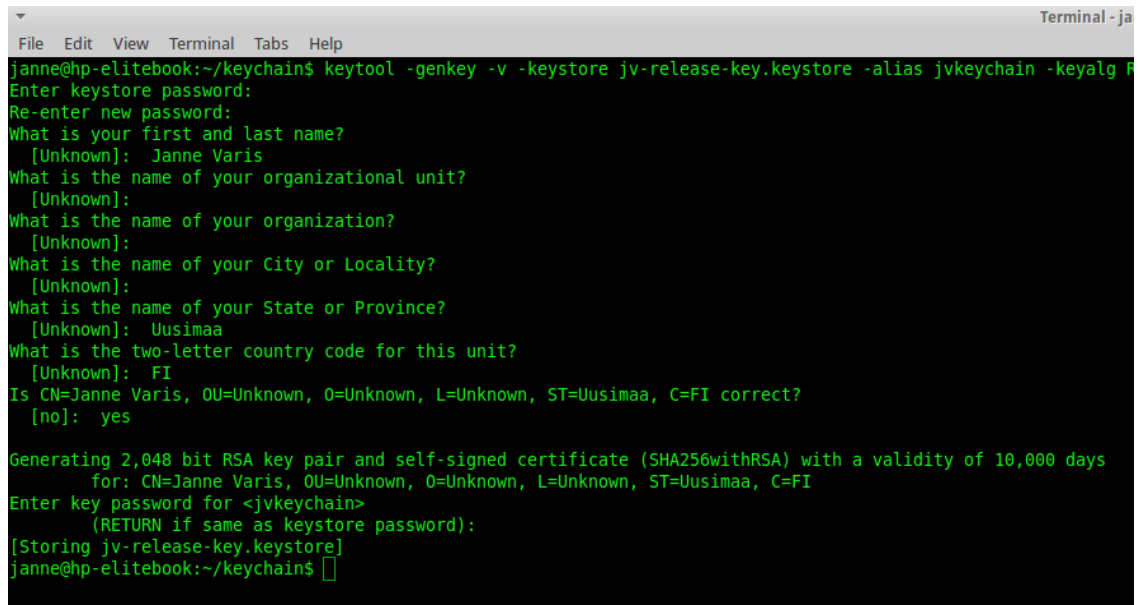
1 var noteID = 0;
2 var noteID = localStorage.getItem("number");
3
4 for (var i = 0, len = localStorage.length; i < len; ++i) {
5   $$('#vault').empty();
6
7   var storedNotes = localStorage.getItem(localStorage.key(i));
8   $('div id="vault" style="display:none"').append(storedNotes + '</div>').appendTo('#storage-container');
9   var message = $('#vault .savedNote');
10  $('#storage-container').prepend(message).enhanceWithin();
11
12  list = $('#savedNote');
13  parent = list.parent();
14  list.sort(function (a, b) {
15    var aProp = parseInt($(a).attr('id'));
16    bProp = parseInt($(b).attr('id'));
17    return (aProp < bProp ? 1 : aProp > bProp ? -1 : 0);
18  });
19  parent.append(list);
20 }
21
22 $(document).on("click", ".save", function () {
23
24   if (document.getElementById('title').value == '' && document.getElementById('content').value == '') {
25     $('#emptyTitle').popup('open');
26   } else if (document.getElementById('title').value == '') {
27     $('#emptyTitle').popup('open');
28   } else if (document.getElementById('content').value == '') {
29     $('#emptyContent').popup('open');
30   } else {
31     noteID++;
32
33     var noteTitle = document.getElementById('title').value;
34     var noteContent = document.getElementById('content').value;
35     var todayStamp = new Date();
36
37     noteID + "<span></li></div>" + noteTitle + "</h2><p style='overflow: visible !important; white-space: normal !important;'>" + noteContent + "</p></li></div data-role='controlgroup' data-type='horizontal' data-mini='true'><a href='#' class='ui-shadow ui-corner-all ui-btn-icon-left ui-icon-mail ui-btn-a send'>Send</a><a href='#' id='delete' class='ui-shadow ui-btn ui-corner-all ui-btn-icon-left ui-icon-delete ui-btn-a del'>Delete</a></div></ul>";
38     localStorage.setItem(noteID, stringNote);
39     localStorage.setItem("number", noteID);
40
41     $('#storage-container').prepend(stringNote).enhanceWithin();
42     $('#title, #content').val('');
43   }
44
45   $('body').append("<div class='dark-overlay'></div>");
46   $.mobile.loading('show', {
47     text: "Saving to local storage...",
48     textVisible: true,
49     theme: "a"
50   });
51
52   setTimeout(function () {
53     $('body').find(".dark-overlay").remove();
54     $.mobile.loading('hide');
55   }, 4100);
56 });
57
58 $(document).on("click", ".send", function () {
59   var mailTitle = $(this).closest('ul').find('h2').html();
60   var mailContent = $(this).closest('ul').find('p').html();
61
62   window.location.href = "mailto:subject=" + mailTitle + "&body=" + mailContent;
63 });
64
65 $(document).on("click", ".del", function () {
66   $(this).addClass('tagged');
67   $('#confirmDialog').popup('open');
68   var noteToDelete = $(this).closest('ul').find('span').html();
69
70   $(document).on("click", ".deletethis", function () {
71     localStorage.removeItem(noteToDelete);
72     $('#tagged').closest(".savedNote").remove();
73   });
74 });
75
76 $(document).on("click", ".cancelthis", function () {
77   $('#tagged').removeClass('tagged');
78 });
79

```

Kuva 24. Setnote-main.js

4.10 Julkaisu ja GNU GPL3-lisenssiehdot

Ennen julkaisua täytyi luoda RSA-avainpari, jotta sovelluksesta pakattava asennustiedosto voitaisiin luotettavasti allekirjoittaa – joka on välttämätöntä sovelluksen julkaisemiseksi. Tämä vaihe tehtiin Xubuntu terminaalissa kautta seuraavasti (Kuva 25). (Android Developers 2015.)



```
janne@hp-elitebook:~/keychain$ keytool -genkey -v -keystore jv-release-key.keystore -alias jvkeychain -keyalg RSA
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Janne Varis
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]: Uusimaa
What is the two-letter country code for this unit?
[Unknown]: FI
Is CN=Janne Varis, OU=Unknown, O=Unknown, L=Unknown, ST=Uusimaa, C=FI correct?
[no]: yes

Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
for: CN=Janne Varis, OU=Unknown, O=Unknown, L=Unknown, ST=Uusimaa, C=FI
Enter key password for <jvkeychain>
(RETURN if same as keystore password):
[Storing jv-release-key.keystore]
janne@hp-elitebook:~/keychain$
```

Kuva 25. Allekirjoitusavaimen luominen terminaalissa

Tämän jälkeen projektiin kansiorakenteesta (kuten kuvattu 4. pääluvun 4. alaluvussa) tehtiin zip-paketti, joka ladattiin Adoben omistamaan PhoneGap Build-palveluun, jossa voi rajattuun määrään saakka pakata asennusmuotoon omia yksityisiä sovelluksiaan ilmaiseksi. Tähän tarvittiin myös edellä luotu allekirjoitusavain, joka ladattiin myös palveluun. Avaimen salasanojen syötön jälkeen oli mahdollista rakentaa sovelluksen zip-paketista asennuskelpoinen, allekirjoitettu versio. Tämän jälkeen APK-asennuspaketin pystyi lataamaan PhoneGap Build-palvelusta ja julkaisemaan. (Adobe 2015.)

Sovellus on avoimen lähdekoodin sovellus, joka on julkaistu lähdekoodeineen kirjoittajan omassa blogissa sekä myös Google Play-sovelluskaupassa kirjoittajan nimellä. Julkaisulisenssinä oli GNU General Public License versio 3. (Varis 4.5.2015.) (Google Play 2015.)

GNU General Public License version 3:n ehdot keskittyvät täysin vapaaseen jakamiseen muokkaamiseen ja käyttämiseen. Sen määrittelemät neljä vapautta ovat suomeksi seuraavat: vapaus käyttää ohjelmaa mihin tahansa tarkoitukseen, vapaus muuttaa ohjelmaa sopimaan omiin tarpeisiisi, vapaus jakaa ohjelma ystäviesi ja naapureidesi kanssa ja vapaus jakaa muutokset, jotka teet ohjelmaa. (GNU 2007.)

5 PhoneGap-sovelluskehityksen hyödyt kehityksessä

5.1 Ohjelmointi

PhoneGap-sovelluskehityksen hyödyt tulivat selvästi esille tämän projektin aikana SetNote-sovellusta kehitettäessä, tukien kirjoittajan omaa näkemystä siitä, että hybridikehitys web-teknologioilla kuten HTML, CSS ja JavaScript on tullut jäädäkseen. Hybridikehityksen saralla PhoneGap edustaa pisimmälle vietyä ajattelua ja toteutusta, koska sen käyttöönotto-vaatimukset ovat hyvin matalat verrattuna natiiveihin ohjelmointikieliin. On helppoa kuvitella esimerkiksi nykypäivän web-kehittäjien vaihtavan sovelluskehitykseen, koska kynnys siihen on niin matala ja markkinat niin suuret.

SetNote-hybridisovelluksen kehityksen kuluessa kävi yhä selvemmäksi, että web-teknologioihin nojaava sovelluskehitys PhoneGapilla on erittäin joustavaa, eli on mahdollista ratkaista ongelmia esimerkiksi yksinkertaisella CSS-muutoksella, joka ei välttämättä ole kovinkaan vaikea toteuttaa, mutta säästää natiiveihin ohjelmointikieliin verrattuna huomattavasti kehittäjän työaikaa, joka taas vaikuttaa positiivisesti koko kehitysprosessiin, kun se pysyy mielekkäänä.

Sovelluksen jatkokehitys ja päivittäminen ovat tämän projektin aikana kerättyihin tietoihin nähden erittäin mahdollisia ja kannattavia, koska tehdyn koodin voi hybridikehityksen ansiosta käyttää uudelleen vaikka siirtyisi kehittämään sovelluksia toiselle käyttöjärjestelmälle Androidin sijaan. On kuitenkin huomautettava, että natiivit sovellukset ovat usein edelleen sulavampia käyttää, varsinkin, jos puhutaan mobiilipeleistä. Tämäkin asia voi muuttua sen myötä mitä HTML5 ja JavaScript jatkossa tarjoavat, kun kehittäjien kiinnostus niitä kohtaan kasvaa hybridikehityksen yleistyessä yhä enemmän.

Kehitysaikaan PhoneGap vaikutti selvästi eniten. Web-teknologioilla kehittäminen on erittäin nopeaa verrattuna natiiveihin ohjelmointikieliin, joissa toisissa on vanhanaikainen tai vaikea syntaksi, tai joissa joutuu näkemään huomattavasti enemmän vaivaa ongelmien ratkaisemiseksi, koska ne ovat juuri koostumukseltaan ja rakenteeltaan tietynlaisia, jolloin tiettyyn ongelmaan saattaa olla hyvin rajallinen määrä ratkaisuja. Tähänkin on paljon poikkeuksia, mutta monissa tapauksissa hybridikehitys on silti erittäin varteenotettava vaihtoehto.

Muutokset sovellukseen oli projektin aikana helppo toteuttaa, koska sovellusta pystyi koodaamaan myös pienemmissä spurteissa, eikä yhden ominaisuuden kehittämiseen kulunut liian paljon aikaa, jolloin kehitysprosessi pysyi monipuolisena, eikä jäänyt junnaamaan

paikalleen mihinkään tiettyyn pisteeseen. Kaiken kaikkiaan arvio PhoneGapin eduista voidaan summata jotakuinkin näin: etuja ovat kehityksen nopeus, matala käyttöönottokyn-
nys, mukautuvuus, koodin uudelleenkäyttömahdollisuus ja sopivuus esimerkiksi yritysten
lyhyisiin projekteihin, joiden toteuttaminen ei välttämättä olisi edes järkevää natiiveilla oh-
jelmointikielillä palkkakustannusten vuoksi. PhoneGap on selvästi hyödyllinen väline hyb-
ridikehityksessä ja on tämän kirjoittajan mielestä tullut jäädäkseen.

6 Pohdinta

Tämän projektin tavoitteita olivat Android-sovelluksen kehittäminen PhoneGap-sovelluskehityksen avulla, sovelluksen ja sen lähdekoodin julkaisu, sen kehityksen dokumentointi ja sitä kautta hyvän dokumentointitavan omaksuminen sekä PhoneGap-sovelluskehityksen etujen tutkiminen ja omien taitojen kartuttaminen tulevaa sovelluskehitystä silmälläpitäen.

Avoimen lähdekoodin Android-sovelluksen kehitys PhoneGap-sovelluskehityksen avulla sujui hyvin onnistuneeseen loppuunsa saakka, ja SetNote-sovellus ja sen lähdekoodi saatiin julkaistua tavoitteen mukaisesti. Tähän voi olla tyytyväinen. Sovelluksen dokumentointi meni myös hyvin. PhoneGapin etujen tutkiminen oli myös yksi onnistuneesti täytetty tavoite, josta oli hyötyä myös tulevaa sovelluskehitystä ajatellen.

Työllä ei ollut toimeksiantajaa, joten kehitystyö sujui oman aikataulun mukaan. Tässä oli sekä hyvää että huonoa – toisaalta kehityksessä oli oma rauha tehdä omassa tahdissa asioita, mutta toisaalta olisi voinut olla hyvä tehdä töitä myös toimeksiantajan aikataulun mukaan, että työn tulos olisi ollut paremmin ennustettavissa, vaikka lopputulos tässä tapauksessa olikin hyvä.

Työn aikana oppia tuli lisää PhoneGapin eduista ja hyödyistä ja siitä, miten niitä voi hyödyntää sovelluskehityksessä tulevaisuudessa tämän projektin lisäksi. Tietomäärä karttui myös hyvästä dokumentointitavasta ja sen tarpeellisuudesta projektien osana, koska sen hyödyllisyys oli havaittavissa jo yksin toteutetussa opinnäytetyöprojektissa.

Kokonaisuutena projekti meni erittäin hyvin.

Lähteet

Adobe 2015. Adobe PhoneGap Build. Luettavissa: <https://build.phonegap.com/>. Luettu: 4.5.2015.

Android Developers 2015. Signing Your Applications. Luettavissa: <http://developer.android.com/tools/publishing/app-signing.html>. Luettu: 2.5.2015.

Apache Ant 2015. Apache Ant. Luettavissa: <http://ant.apache.org/>. Luettu: 8.4.2015.

Binstein, J. 21.7.2014. Installing PhoneGap/Cordova on Windows and Linux. Made by Knight. Luettavissa: <http://madebyknight.com/installing-phonegapcordova-windows-linux/>. Luettu: 8.4.2015.

Github 2015. Launcher Icon Generator. Luettavissa: <https://romannurik.github.io/AndroidAssetStudio/icons-launcher.html>. Luettu 3.5.2015.

GNU 2007. A Quick Guide to GPLv3. Luettavissa: <http://www.gnu.org/licenses/quick-guide-gplv3.html>. Luettu: 25.4.2015

Google 2015. Android Studio. Luettavissa: <http://developer.android.com/sdk/index.html>. Luettu: 8.4.2015.

Google Play 2015. SetNote - Notes Simplified. Luettavissa: <https://play.google.com/store/apps/details?id=com.jdev.setnote>. Luettu: 4.5.2015.

GSMarena 2014. OnePlus One. Luettavissa: http://www.gsmarena.com/oneplus_one-6327.php. Luettu: 8.4.2015.

Higman, J. 17.10.2013. Setting up PhoneGap on Ubuntu for Android app development. Julian Higman : Blog. Luettavissa: <http://julianhigman.com/blog/2013/10/17/notes-on-setting-up-phonegap-on-ubuntu-for-android-app-development/>. Luettu: 8.4.2015.

Intel 2015a. Intel XDK. Luettavissa: <https://software.intel.com/en-us/html5/tools>. Luettu: 8.4.2015.

Intel 2015b. Intel XDK Install Instructions. Luettavissa: <https://software.intel.com/html5/install-instructions>. Luettu: 9.4.2015.

jQuery 2015. What is jQuery?. Luettavissa: <https://jquery.com/>. Luettu: 8.4.2015.

jQuery Mobile 2015. A Touch-Optimized Web Framework. Luettavissa: <https://jquerymobile.com/>. Luettu: 8.4.2015.

Node.js 2015. About documentation. Luettavissa: <https://nodejs.org/documentation/>. Luettu: 24.4.2015.

Openclipart 2015. Sticky Note. Luettavissa: <https://openclipart.org/detail/48931/sticky-note>. Luettu: 3.5.2015.

Oracle 2015. Learn About Java Technology. Luettavissa: <https://www.java.com/en/about/>. Luettu: 8.4.2015.

PhoneGap 2015. Platform Guides. Luettavissa: http://docs.phonegap.com/en/edge/guide_platforms_index.md.html. Luettu: 8.4.2015.

ThemeRoller 2015. ThemeRoller. Luettavissa: <http://jqueryui.com/themeroller/>. Luettu 28.4.2015.

Ubuntu Documentation 2015. AndroidSDK. Luettavissa: https://help.ubuntu.com/community/AndroidSDK#Modifying_the_PATH_Environment_Variable. Luettu: 8.4.2015.

Varis, J. 4.5.2015. Android App Release: SetNote – Note Taking Simplified. The Penguin Habitat – Things You Can Accomplish With Linux (And Android) – Now With Arduino!. Luettavissa: <https://jvaris.wordpress.com/2015/05/04/android-app-release-setnote-note-taking-simplified/>. Luettu: 4.5.2014.

Xubuntu 2015. Get Xubuntu. Luettavissa: <http://xubuntu.org/getxubuntu/>. Luettu: 8.4.2015.