



TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

VEERA KANSIKAS
PLA-32820 MOBIILIOHJELMOINTI

Oppimispäiväkirja

4. maaliskuuta 2018

TIIVISTELMÄ

VEERA KANSIKAS: PLA-32820 Mobiiliohjelmointi

Tampereen teknillinen yliopisto

Oppimispäiväkirja, 10 sivua, 0 liitesivua

Maaliskuu 2018

2018 kevään Tampereen Teknillisen Yliopiston 5op. kurssi mobiiliohjelmoinnista. Kurssi järjestetään pääasiassa Porista käsin, joten suurena osana kurssin suoritusta on oppimispäiväkirjan pito. Dokumentoituna tulee olla kaikista tehtävistä työselostukset, tehtävään käytetty aika ja kommentit suorituksesta.

Itse harjoitukset koostuvat kolmesta osasta: harjoitustehtävät, Googlen Udacity-kurssi Android sovelluksien kehityksestä ja laaja omatoiminen harjoitustyö. Aloitin kurssin suorituksen 16.2 ja kaiken tulee olla valmista toukokuun loppuun mennessä.

SISÄLLYSLUETTELO

1.	AJANKÄYTTÖ	ERROR! BOOKMARK NOT DEFINED.
2.	HARJOITUSTEHTÄVÄT	
2.1	Tutustuminen mobiiliympäristöön	Error! Bookmark not defined.
2.2	Git ja versionhallinta	Error! Bookmark not defined.
2.3	Android-ympäristön asennus	
2.4	Androidilla perus käyttöliittymä	
2.5	Tietokantasovellus	
2.6	Tietokantasovellus: lisää toiminnallisuuksia	
2.7	Firebase tietokanta	
2.8	Firebase tietokanta: autentikointi	
2.9	Paikannus ja kartta	
2.10	Paikannus ja kartta: ?	
3.	MOOC-AINEISTON SUORITUS	ERROR! BOOKMARK NOT DEFINED.
3.1	Create Project Sunshine	Error! Bookmark not defined.
3.2	Connect to the Internet	Error! Bookmark not defined.
3.3	RecyclerView	Error! Bookmark not defined.
3.4	Intents	Error! Bookmark not defined.
3.5	Lifecycle	Error! Bookmark not defined.
3.6	Preferences	
3.7	Storing Data in SQLite	
3.8	Content Providers	
3.9	Building a Content Provider	
3.10	Background Tasks	
3.11	Completing the UI	
3.12	Polishing the UI	
4.	LAAJA HARJOITUSTYÖ	ERROR! BOOKMARK NOT DEFINED.
4.1	Harjoitustyön idea	Error! Bookmark not defined.
4.2	Vaatimusmäärittely	
4.3	Työkalut	
5.	YHTEENVETO	ERROR! BOOKMARK NOT DEFINED.
	LÄHTEET	10

1. AJANKÄYTTÖ

Käytin kurssiin yhteensä X tuntia aikavälillä 16.2.2018 – XX.X, keskimäärin X tuntia kerrallaan X päivää viikossa.

Päivä	Aika (h)	Kuvaus
16.2.	0.5	Android-ympäristön asennus ja testaus
18.2.	5	Oppimispäiväkirjan muotoilu, tehtävät 1 ja 4

Liityin kurssille kuukauden myöhässä, joten aloitin tehtävien kanssa 16.2 asentamalla Android-ympäristön ja tutustumalla siihen. Opintopäiväkirjan aloituksen ja loput ensimmäisistä neljästä tehtävistä tein määräpäivänä 18.2.

21.2	0.5	Harjoitustyön ideointi
26.2	5	MOOC-kurssin 1 osio ja ensimmäinen puolisko 2:sta
27.2	4	MOOC-kurssin 2 osio loppuun
1.3	1	Tehtävän 5 layout
3.3	4	MOOC-kurssin osio 6
4.3	7	Tehtävän 5 logiikka

Seuraavalla viikolla suunnittelin harjoitustyön vaatimuksia tarkemmin ja mietin vaihtoehtojani käytettävien työkalujen suhteen. Loppuviikosta aloitin MOOC-kurssin kanssa ja huomasin että 4.3 mennessä pitäisi ehtiä SQLiteä käsittelevään osioon, jotta tehtävien 5 ja 6 teko sujuisi helpommin. Suunnitelman sabotoi kuitenkin puolessavälin viikkoa iskenyt flunssa ja päädyin hyppäämään 4 ja 5 osioiden yli ehtiäkseni käydä osion 6 läpi.

2. HARJOITUSTEHTÄVÄT

Osana kurssin suoritusta on 10 harjoitustehtävää. Varsinaiset palautukset ovat erillisinä versionhallinnassa ja tässä osiossa ovat kuvaukset tehtävien suorituksista ja niihin liittyvät kommentit.

2.1 Tutustuminen mobiiliympäristöihin

Tehtävänä oli valita Android-laite tutustuttavaksi ja selvittää sen ohjelmointimahdollisuuksia. Valitsin laitteekseni Oneplus X puhelimeni. Se toimii hyvin, omaa sopivasti ominaisuuksia ja on helposti saatavilla.

Tietojen kerääminen hoitui helposti valmistajan sivuilta ja parilla google-haulla.

2.2 Git ja versionhallinta

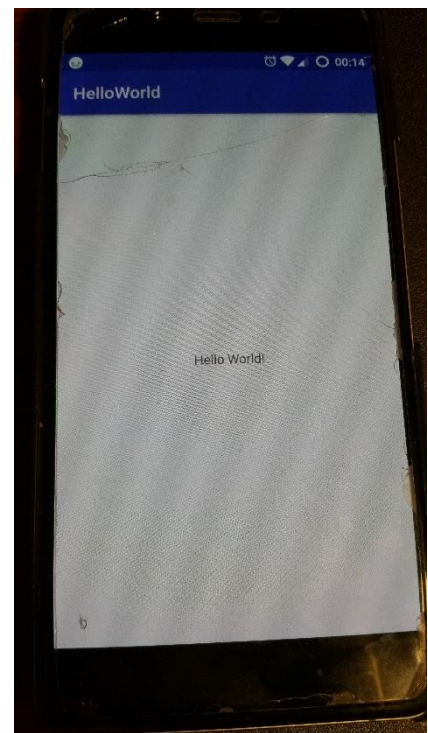
Päätin käyttää versionhallintana kurssilla GitHubia, johon omistin jo tunnuksen. Pääsyynä valinnalle oli kuitenkin GitHubin sisällön helppo jakaminen työnhaun liitteenä ja tunnuksen samanhenkinen sisältö kurssin harjoitustyöhön nähden.

Itse kurssin suorituksen osalta totesin kuitenkin että on ehkä varmempaa käyttää virallista kurssin GitLabia, jonka sitten lisäsin jo luotuun Git-hakemistoon uudeksi remoteksi. Sain siis työt hyvin näkyville sekä GitHubiin omaa käyttöäni varten, että GitLabiin kurssia varten.

2.3 Android-ympäristön asennus ja testaus

Android Studion asennus Windows pöytäkoneelle sujui pitkälti mutkattomasti asennuksen ohjeistuksen mukaan.

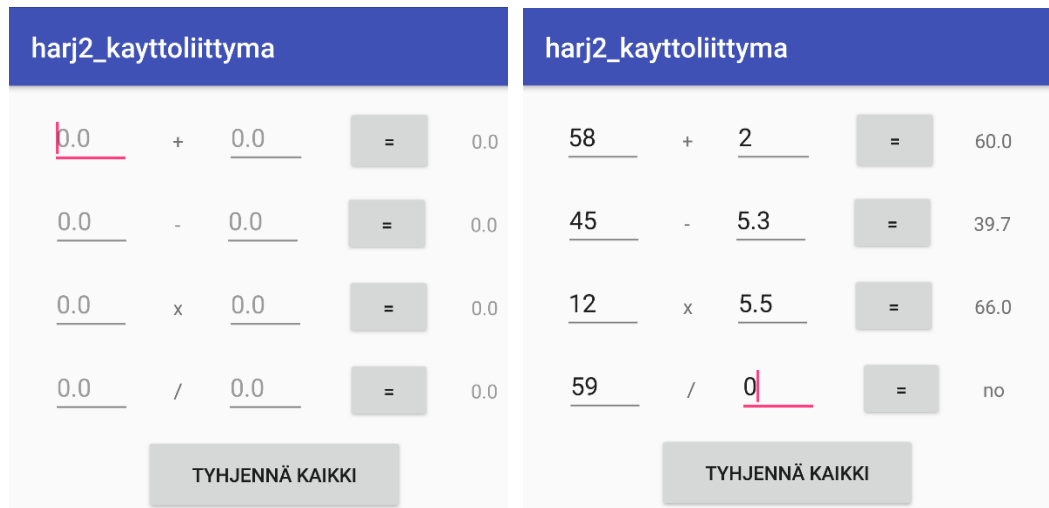
Ensimmäisen Hello World -ohjelman ajaminen puhelimella onnistui heti, emulaattorin asennuksessa tuli vastaan *"VT-x is disabled in BIOS."* Ensimmäiset tehtävät hoituvat hyvin ilman, joten jätän selvittämisen hieman myöhemmälle.



2.4 Androidilla perus käyttöliittymä

Tehtävänä oli tehdä Android Studiolla yksinkertainen laskinsovellus. Android Studiossa navigointi ja käyttöliittymän teko vaati hieman totuttelua, mutta toimi ihan kätevästi ja kaikesta oli helppo tehdä skaalautuvaa ankkurointien avulla.

Javan kirjoittaminen ei myöskään ollut tuttua ennestään, joten googlailua sai tehdä monessa vaiheessa syntaksien varmistamiseen ja aikaa kului siten tavallista enemmän. Mitään ylitsepääsemätöntä ei kuitenkaan tullut vastaan.



2.5 Tietokantasovellus

Tehtävänantona oli luoda SQLite tietokantaa käyttävä sovellus. Vaatimuksina tietojen lisäys tietokantaan ja kannan sisällön näyttäminen ruudulla.

Aloitin ensin ehdotetulla Aku Ankkoja listaavalla sovelluksella, mutta koska en aikonut käyttää annettuja esimerkkikodeja, totesin että on ehkä selvempää jos otan toisen aiheen ja päädyin jääkaapin sisältöä seuraavaan sovellukseen.

MOOC-kurssin SQLite osion jälkeen kannan naputtelu tehtävää varten tuntui sujuvan sujuvasti, kunnes tuli aika testata tietokantahakua ja sovellus kaatui. Monen tunnin etsinnän ja uudelleenkirjoittamisenkaan jälkeen en ole vielä onnistunut paikantamaan syyllistä. Kuitenkin sovellus myös kaatuu jos contractissa määritellyt kolumnien arvoja vaihtaa, mille en myöskään näe syytä? Jokatapauksessa, tässä kohtaa pitänee luovuttaa tämän päivän osalta, toivoa armahdusta deadlinesta jäämisestä, ja jatkaa huomenna. ☹️

2.6 Tietokantasovellus: lisää toiminnallisuuksia

Edellisen tehtävän laajennus lisäämällä poista- ja järjestä-ominaisuudet sovellukseen.

Poisto-ominaisuus on käytännössä myös tehty, kunhan edellinen kohta toimisi.

2.7 Firebase tietokanta

2.8 Paikannus ja kartta

3. MOOC-AINEISTON SUORITUS

Kurssin toinen osa on suorittaa Googlen rakentama online kurssi 'Developing Android Apps' Udacityssä: <https://www.udacity.com/course/new-android-fundamentals--ud851>.

Udacityn kurssi koostuu 12:sta osiosta, joiden suorituksen dokumentointi on tässä jaoteltu omiin alaotsikkoihinsa. Tehtäviin olennaisesti liittyvät tiedostot löytyvät versionhallinnasta *MOOC-kurssi/exercises* ja *MOOC-kurssi/sunshine* osioittain jaoteltuina.

3.1 Create Project Sunshine

Kurssi alkaa perusasioilla layoutista, ja selvästi parempi suoritusjärjestys olisi ollut tehdä tämä ennen ensimmäisiä harjoitustehtäviä.

Osiossa käydään läpi TextView ja ScrollView, näiden ominaisuuksia ja miten TextView:iin lisätään Javan kautta tekstiä. Harjoituksena on ensin 90-luvun leluja listaava sovellus, jonka jälkeen opittuja asioita hyödynnetään kurssin läpi jatkuvan Sunshine sovelluksen aloittamiseen. Lopputuloksena Sunshine näyttää skrollattavan listan mallisääennusteita.

Sunshine
Today, May 17 - Clear - 17°C / 15°C
Tomorrow - Cloudy - 19°C / 15°C
Thursday - Rainy- 30°C / 11°C
Friday - Thunderstorms - 21°C / 9°C
Saturday - Thunderstorms - 16°C / 7°C

3.2 Connect to the Internet

Osio alkaa lupien (permissions) käsittelyllä ja miten mahdollisimman vähäinen lupien hyödyntäminen on käyttäjän kannalta parempi. Seuraavaksi tehdään http-kysely, ja siirretään se main threadista tausta-ajoon. Lopuksi esitellään nopeasti hieman JSON:in syntaksia ja lisätään layouttiin menubutton.

Staattisen listan sijaan Sunshine siis hakee nyt sääennusteensa kurssin palvelimelta. Sääennusteet voi myös päivittää, ja lataamisen ajan näytöllä näkyy lataussymboli.

Sunshine	REFRESH
Today, 27 February - Light Rain - 18°C / 13°C	
Tomorrow - Moderate Rain - 17°C / 13°C	
Thursday - Clear - 16°C / 12°C	
Friday - Scattered Clouds - 12°C / 12°C	

3.3 RecyclerView

blaablablaa

3.4 Intents

3.5 Lifecycle

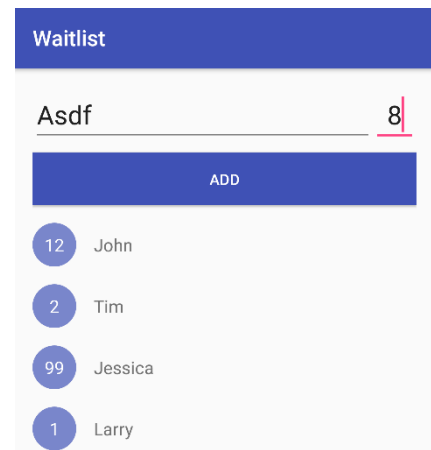
3.6 Preferences

3.7 Storing Data in SQLite

Hyppäsin 3 osiosta suoraan tähän tehtävien 5 ja 6 deadlineien lähestyessä. Varsinkin kun jo kevään toinen flunssa puolitti työtahdin ja pakotti nukkumaan suuren osan päivistä. Onneksi osio ei rakentunut juurikaan aikaisempien päälle, joten hyppy ei johtanut ongelmiin.

Harjoitustehtävänä oli luoda jonotussovellus ravintolalle, jossa tietokantaan lisätään varaajan nimi, seurueen koko ja lisäyshetken kellonaika. Sovelluksen avulla vieraita vastaanottava henkilö voi sitten helposti tarkastella varausjonoa ja poistaa siitä vieraita sitä mukaan kun näille vapautuu pöytä.

Ensimmäisenä osiossa käsiteltiin sopimuksen (contract) teko ja miten se määrittelee tietokannalle ja muulle sovellukselle yhteisen termistön. Seuraavaksi luotiin tietokanta ja haettiin sen koko sisältö, jota varten myös Adapteria piti päivittää. Lopuksi käytiin läpi vieraiden lisäys ja poisto tietokannasta ja kursorin käsittely ja vaihto.



Waitlist	
Asdf	8
ADD	
12	John
2	Tim
99	Jessica
1	Larry

3.8 Content Providers

3.9 Building a Content Provider

3.10 Background Tasks

3.11 Complementing the UI

3.12 Polishing the UI

4. LAAJA HARJOITUSTYÖ

Viimeisenä osana kurssia on oman idean pohjalta tehty sovellus, joka käyttää jotakin mobiililaitteen ominaisuuksista. Omana suunnitelmanani on tehdä augmented reality sovellus puhelimelle, ja siten käytän harjoitustyössä vähintään puhelimen kameraa, paikannusta ja karttaa.

4.1 Harjoitustyön idea

Innostuin 2017 syksyllä VR/AR kehityksestä, mutta AR:n koodamista en ole vielä päässyt yrittämään, joten harjoitustyön tekeminen siitä oli heppo päätös. Vuodenvaihteessa aloitin myös kiipeilyn, ja kesäksi on ollut puhetta lähteä koittamaan oikeita lohkareita ja kallioita. Nämä kaksi yhdistämällä siis syntyi idea tehdä AR-sovellus, johon voi merkitä kallioiden kiipeilyreitit – mikä mahdollisesti olisi kätevämpi käyttää kuin paperiset reittikartat.

Sovelluksessa on kartta, jossa näkyvät sovellukseen syötetyt kalliot ja muut sopivat kiipeilypaikat. Jokaisella paikalla on oma sivunsa, jossa näkyvät sen mahdollinen nimi, kuvat, sen reittien vaikeusteet, kuvaus paikasta ja tarvittaessa ohjeet lähestymiseen. Sivulla olisi myös hyvä olla perinteiset reittikartat kuvamuodossa.

Paikanpäällä käyttäjä voi siirtyä sovelluksen AR-tilaan, ja osoittamalla kamerallaan kalliota näytöllä näkyvät mahdolliset reitit vaikeusasteineen. Reittejä voi suodattaa mm. vaikeusasteen mukaan tai valita vain yhden näkyviin. Myös reittien muokkaus ja uusien lisääminen on mahdollista.

Idea saattaa olla tarvittua haastavampi, mutta pidän siitä ja koska henkilökohtaisesti tavoitteenani on kurssin suorituksen lisäksi saada mahdollisimman paljon hyötyä tulevalle uralleni – joka toivottavasti liittyy VR/AR kehitykseen – harjoitustyö on mielestäni todella osuva. Mietin kuitenkin myös muita mahdollisia ideoita sille varalle, että työssä tulee ylitsepääsemättömiä esteitä vastaan.

4.2 Vaatimusmäärittely

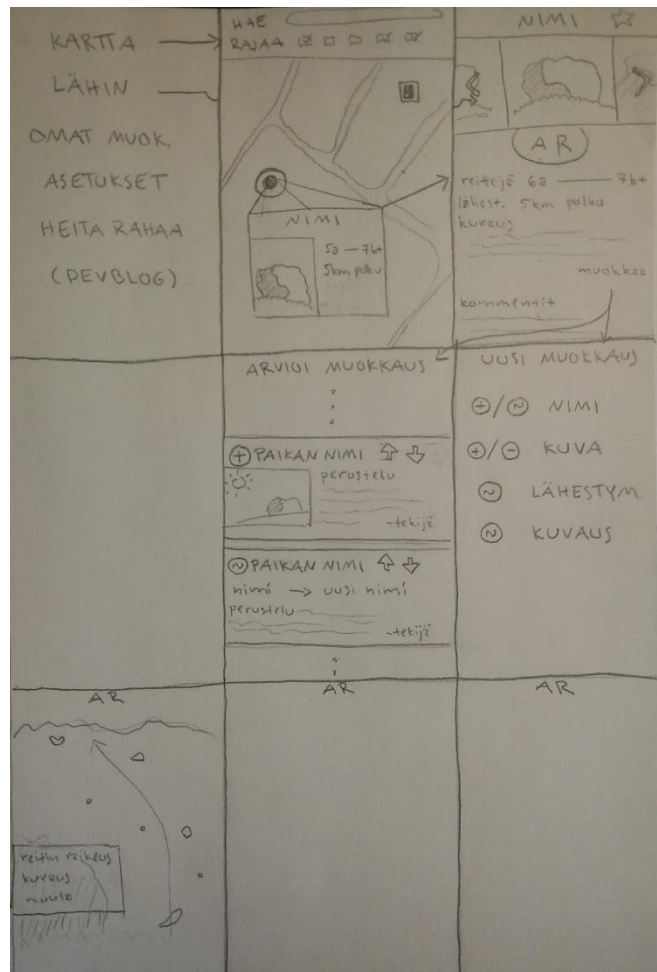
Oheisessa kuvassa on ensimmäisiä suunnitelmia asioiden jaottelusta käyttöliittymään.

4.2.1 Kartta

Kartalla näkyvät kaikki sovellukseen lisätyt kiipeilypaikat, joista eri tyyppisillä paikoilla on omat kuvakkeensa.

Paikkaa klikatessa esiin tulee pikakatsaus paikasta: nimi, kuva, reittien lukumäärä ja vaikeustasoskala, mahdolliset aukioloajat tai muu huomioitava asia.

Paikkoja voi hakea nimen perusteella tai suodattaa ominaisuuksien mukaan. Esim. vain sisätilat ja 7b reittejä sisältävät.



4.2.2 Kiipeilypaikan sivu

Kiipeilypaikan sivulla näkyvät kaikki siihen liittyvät tiedot: nimi, kuvat, reittien määrä ja vaikeusasteskaala, huomioitavat asiat, osoite ja kuvaus. Lisäominaisuutena sivulle voi myös tarvittaessa lisätä kommentoinnin.

Sivulla keskeisenä linkkinä on siirtyminen AR-tilaan, mahdollisesti vain kun laite on tarpeeksi lähellä sijaintia. Muulloin mahdollisuutena on tarkastella reittejä kuvina, mikäli sellaisia on lisätty.

4.2.3 AR-tila

AR-tilassa näkyvät oletusarvoisesti kaikki paikan reitit oikeilla fyysisillä paikoillaan oletusväreillä. Väritys on mahdollista muuttaa kuvastamaan vaikeusasteita.

Reittejä voi suodattaa mm. vaikeusasteen mukaan, tai niistä voi valita näkyviin vain yhden. Reitin ollessa valittuna sen lähelle ilmestyy infolaatikko, joka kertoo sen vaikeusasteen ja mahdolliset lisähuomiot tai rajoitukset.

Reiteistä on AR-tilassa myös mahdollista ottaa kuva, jolloin reittiä voi tarkastella myös ilman AR-tilaa, esimerkiksi akun ollessa vähissä tai kun käyttäjä ei ole reitin luona.

4.2.4 Paikkojen ja reittien muokkaus

Jotta paikkojen ja reittien tietoja olisi mahdollista muokata ilman pelkoa laatua huonontavista muutoksista tai poistoista, muokkausten olisi hyvä käydä nopean arvioinnin läpi. Ehdotettu muokkaus lisittäisiin arvioitavien muokkausten listaan, josta käyttäjät voivat äänestää muokkausta alas tai ylöspäin ja tietyn rajan ylittäneet muokkaukset hyväksytään.

Pienellä käyttäjämäärällä pelkkä adminin kautta hyväksyttäminen toimii myös ja on todennäköisesti ensimmäinen versio, käyttäjä-arvioinnin voi lisätä mikäli resurssit (= oma aikani) riittävät ja sille on tarvetta.

4.2.5 Laitteistovaatimukset

Android 7.0 thanks. Ja kamera ja GPS.

4.3 Työkalut

Android Studio ja Googlen 2017 syksyllä julkaisema ARCore, joka valitettavasti vaatii Android 7.0:n toimiakseen, jota mm. oma puhelimeni ei tue. Tästä huolimatta luultavasti helpoin ja tyylikkään ratkaisu. Testaukseen pitää vain kehitellä jokin ratkaisu.

<https://developers.google.com/ar/>

LÄHTEET

Opiskelussa hyödynnetyt verkkoaineistot:

- [1] Udacity Developing Android Apps: <https://www.udacity.com/course/new-android-fundamentals--ud851>
- [2] W3Schools SQL Tutorial: <https://www.w3schools.com/sql/>
- [3] Google Firebase: <https://firebase.google.com/docs/android/>
- [4] Google ARCore: <https://developers.google.com/ar/>