



**Lappeenrannan teknillinen yliopisto**

**Tietotekniikan koulutusohjelma**

**CT60A2511 – Olio-ohjelmointi**

Jiri Musto

Olio-ohjelmointi

**Harjoitustyö - Android applikaatio**

25.7.2019

Tomi Enberg - 0518456

tomi.enberg@student.lut.fi

Markus Strandman - 0521605

markus.strandman@student.lut.fi

Aarne Savolainen - 0521838

aarne.savolainen@student.lut.fi

# Sisällys

1 Johdanto.....	3
2 Suunnittelu ja toteutus .....	4
2.1 Suunnittelu.....	4
2.2 Luokkakaavio .....	5
3 Toiminnallisuus .....	7
3.1 Lista toiminnallisuuksista .....	7
3.2 Korvaukset tietokannalla .....	8
4 Ryhmämme.....	9
4.1 Työnjako.....	9
4.2 Henkilökohtaiset kokemukset.....	9
4.2.1 Tomi .....	9
4.2.2 Markus.....	10
4.2.3 Aarne .....	10
4.3 Pisteytys.....	10

# 1 Johdanto

Loimme Android applikaation nimeltä Food Review, missä voi tarkastella eri yliopistojen ravintoloiden ruokalistoja ja arvostella ruokia. Projekti on tähän mennessä reilusti suurin projekti, mitä olemme luoneet, sekä ensimmäisiä kokemuksia ryhmässä tällaisen projektin luomisesta, sekä tietokannan sisällyttämisestä ohjelmaan.

Valitettavasti emme saaneet oikeita ruokalistoja noudettua, joten ohjelmamme käyttää meidän itsemme luotuja ruokalistoja. Applikaatioon luodaan käyttäjätunnus, joka voi antaa yhden arvostelun per ruoka, mutta omia arvosteluja voi tarkastella, muokata tai poistaa myöhemminkin. Omia tietoja, kuten nimimerkkiä tai kotiyliopistoa, voi muokata asetuksista.

Applikaatiossa on käyttäjä, jolla on ylläpitäjän oikeudet, joka voi muokata ruokalistoja muokkaamalla ravintoloita, ruokia, sekä lisätä tai poistaa niitä. Ylläpitäjä voi myös hallita muita käyttäjiä, esimerkiksi jakamalla niille ylläpitäjän oikeuksia, sekä niiden arvosteluitaan. Kaikki tiedot voi myös tallentaa CSV-tiedostoon, sekä ruuat ja niiden arvostelut myös JSON-tiedostoon.

## 2 Suunnittelu ja toteutus

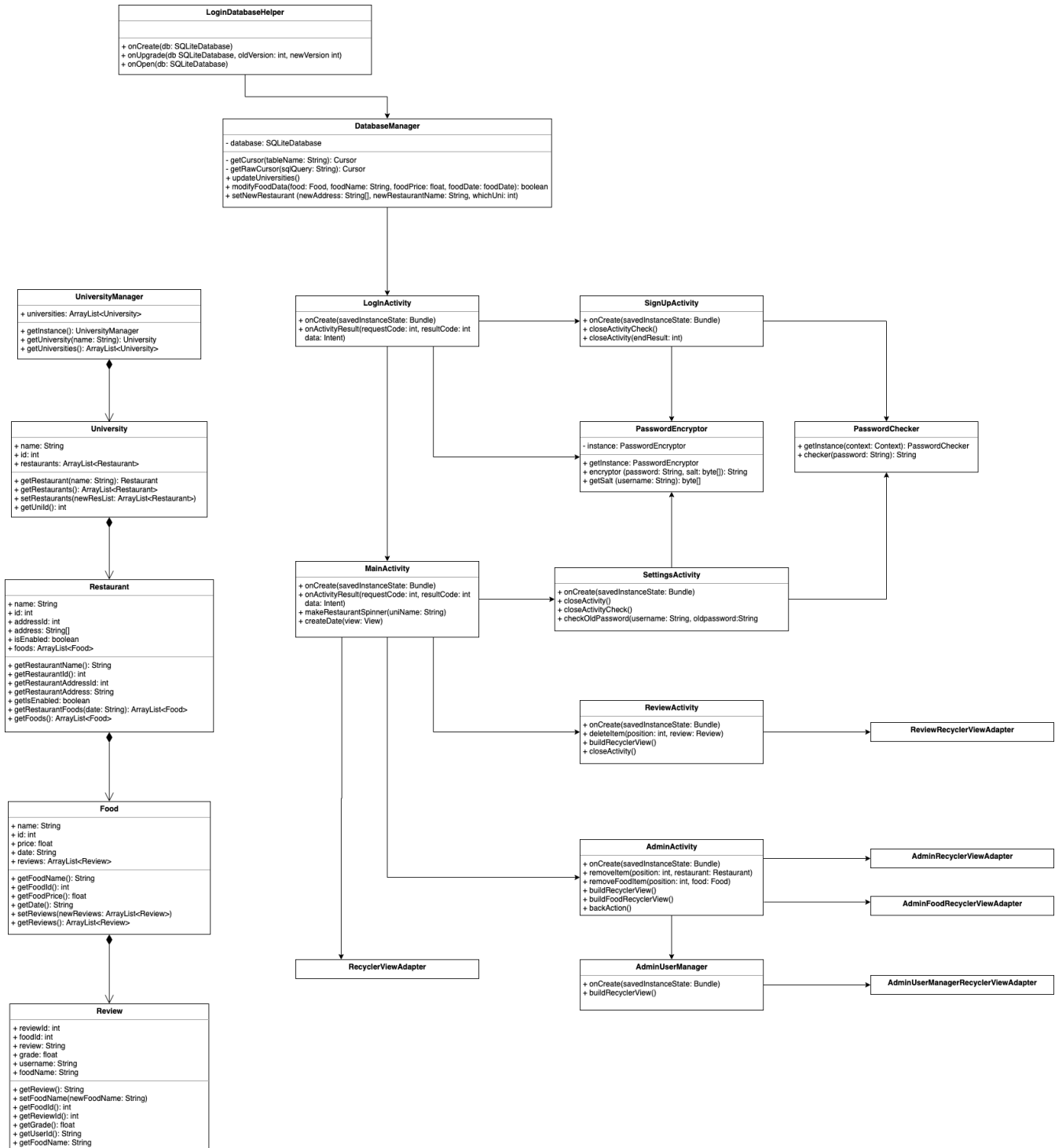
### 2.1 Suunnittelu

Kun ryhdyimme suunnittelemaan, loimme samaan aikaan prototyyppiä käyttöliittymästä, käyttäen apuna Android Studion omia pohjia, jotta nähtäisiin mitä projektimme tulisi vaatimaan. Kun saimme vision, että miltä haluamme käyttöliittymän näyttävän, ryhdyimme suunnittelemaan tietokantaa ohjelman pohjalle, mikä myös kertoi mitä meidän pitäisi lisätä käyttöliittymään. Kun saimme tietokantasuunnitelman luotua, Tomi alkoi sitä toteuttamaan, samalla kun Aarne ja Markus jatkoi käyttöliittymän kehittämistä.

Kun tietokannan tai käyttöliittymän luonti eteni, vastaan tuli monta ongelmaa, jotka johtivat suunnitelmien muutoksiin, suunnitelmat kun ovat pohja muutoksille. Pidimme monta kokousta, jossa kävimme suunnitelmat uudestaan ja uudestaan, selvittäen mitä niillä tarkoitimme ja mitä tulisimme haluamaan jatkossa, ja totta kai, miten ratkoa sen hetkiset ongelmat.

Projektia luodessa testasimme ohjelmaa jatkuvasti. Se, mitä meille kerrottiin ohjelmistotestauskurssilla, piti aika lailla paikkansa, puolet ajasta meillä meni ohjelmistokoodin kirjoittamiseen ja puolet sen testaamiseen. Projekti saatiin kuitenkin monen mutkan kautta valmiiksi.

## 2.2 Luokkakaavio





## 3 Toiminnallisuus

### 3.1 Lista toiminnallisuuksista

- Ruokalistojen tarkastelu (useampi yliopisto, useampi ravintola, eri päivät, useampi ruoka per päivä)
- Käyttäjän luominen, omien tietojen muokkaaminen (nimimerkki, salasana, kotiyliopisto – käyttäjänimeä ei voi muokata, sen täytyy olla kaikilla erilainen)
- Salasana täytyy noudattaa hyvän salasanan merkkejä, sekä tallentuu tietokantaan käyttäen hash-menetelmää ja suolausta
- Kirjautuminen applikaatioon
- Kirjautuessa pyydetään syöttämään 6 satunnaista numeroa (simuloi autentikaattoria)
- Omien arvostelujen tarkastelu ja muokkaaminen tai poistaminen
- Arvostelujen antaminen ruuille (numeerinen ja sanallinen)
- Ylläpitäjä
  - Voi luoda/poistaa ravintoloita
  - Voi luoda/poistaa ruokia
  - Voi poistaa arvosteluja
  - Näkee kaikkien ravintoloiden ja ruokien tiedot ja voi muokata/poistaa niitä
  - Voi jäädyttää ravintoloita
  - Näkee käyttäjien perustiedot ja voi hallita niitä (esim. antamalla ylläpitäjän oikeudet)
- Ruokalistat ovat vakioita (viikoittain, eli joka maanantai samat ruuat, jne.)
- Kaikkien tietojen tallennus CSV-tiedostoon tai ruokien ja niiden arvostelujen tallennus JSON-tiedostoon
- Kaikki tiedot (käyttäjät, yliopistot, ravintolat, ruuat, arvostelut) tallentuvat tietokantaan

## 3.2 Korvaukset tietokannalla

Korvasimme arvostelujen äänestämisen käyttämällä tietokantaa. Koimme tämän ominaisuuden tuottavan eniten ongelmia.



## 4 Ryhmämme

### 4.1 Työnjako

Sovellusta kehittäessämme, emme jakaneet vastuita aluksi lähes ollenkaan. Myöhemmin huomasimme, että työnjakoa on, ja se on tapahtunut aika lailla itsestään. Markus keskittyi pääsääntöisesti eri näkymien tekemiseen, joita Android Studiossa nimitetään activity -nimellä. Aarne keskittyi myös käyttöliittymän toteuttamiseen, mutta erilaisten ”Fragment” popup -ikkunoiden osalta. Tomi puolestaan keskittyi sovelluksen taustalla toimivaan tietokantaan. Raportti tehtiin suurilta osin yhdessä.

Suunnittelimme kuitenkin sovelluksen ulkonäköä, tietokantaa ja toteutustapaa pääsääntöisesti ryhmänä. Kesäleirin luokassa pystyimme helposti kommunikoimaan ja sopimaan asioita yhteisesti. Loimme myös whatsapp -ryhmän, jonka kautta pystyimme jatkamaan ryhmänä työskentelyä kesäleirin loppumisen jälkeen.

### 4.2 Henkilökohtaiset kokemukset

#### 4.2.1 Tomi

Työaika 30 x 7h.

Olio-ohjelmoinnin ja tietokantojen yhteinen harjoitustyö oli ehdottomasti opiskeluaikani laajin työ. Työ vaati aikataulutusta, työnjakoa ja laajaa itsenäistä tiedonhakua. Työ myös opetti todella paljon näistä kaikista. Opin käyttämään Android Studion sisäänrakennettua tietokantajärjestelmää, versionhallintajärjestelmää ja yleisesti ottaen ryhmätyötä.

Yhteistyö ryhmämme kanssa oli helppoa, vaikka itse pääsinkin olemaan läsnä vasta muutaman päivän työn aloittamisen jälkeen. Korvasin kuitenkin tuntini ja tein työtä itsenäisesti korvaamaan poissaoloni. Aikataulutus oli omalta osaltani haastavaa, sillä olin sopinut etukäteen erilaisia työtehtäviä muualle, ja tämän myötä jouduin tekemään myös suuren osan loppuvaiheen työstä etänä.

#### 4.2.2 Markus

Työaika 30 x 7h.

Olio-ohjelmoinnin harjoitustyö oli tähän mennessä opettavaisin projekti mitä olen tehnyt. Harjoitustyö kiteytti hyvin yhteen olio-ohjelmoinnin ja tietokantapuolen sekä näytti ensimmäistä kertaa millaista on tehdä ohjelmistoprojektia ryhmässä. Työtä tehdessä oppi myös käyttämään paremmin GitHubia ja työskentelemään useiden branchien kanssa yhdistellessä muiden tekemiä muutoksia.

#### 4.2.3 Aarne

Työaika 30 x 7h.

Harjoitustyön ollen ensimmäinen tällainen ryhmäprojekti, tässä oppi nopeasti kirjoittamaan koodia ryhmässä, sekä oppi GitHub-versionhallintajärjestelmän käytön ja miten hallita konflikteja toisen koodia yhdistäessä omaan koodiin. Harjoitustyö totta kai myös kiteytti oppimista Java-ohjelmointikielen suhteen. Projektin tekoajan venyessä kuukauteen, ymmärsi myös miten suurempia projekteja ei luoda ihan parissa viikossa.

### 4.3 Pisteytys

Mielestämme olemme noudattaneet tehtävänantoa tarkasti, rakentaneet koodista toimivan, laajennettavan ja ennen kaikkea noudattaneet käyttöliittymän rakentamisessa olio-ohjelmoinnin hyviä käytänteitä ja periaatteita. Rakensimme tietokannan vastaamaan sovelluksen tarpeita ja käytimme suunnittelussa ER-kaaviota ja transformointia hyväksi.

Rakensimme sovellukseen toiminnallisuuksia kaikkein laajimpien vaatimusten mukaisesti ja täytimme myös kaikki ryhmätyön tuomat lisävaatimukset. Ajattelimme, että työ on täyden 40:n pisteen arvoinen.