

<RateMovies>

Tekijät: Thomas Kaatranen

# Kuvaus ohjelmasta

Ohjelma alkaa kirjautumisella. Jos tunnuksia ei ole, käyttäjän on mahdollisuus rekisteröityä ohjelman sisällä, mutta rekisteröityminen on vahvistettava annetusta sähköpostista.

Ohjelma antaa käyttäjälle mahdollisuuden selata läpi Finnkinon tarjoamia elokuvia ja antaa niille tähti-arvosteluja. Elokuvia voi hakea elokuvan nimen perusteella. Arvosteltuja elokuvia käyttäjä kykenee katsomaan ohjelman sisällä ja käyttäjä pystyy myös uudelleen arvostelemaan jo arvostellun elokuvan. Arvostellut elokuvat ovat selattavissa paremmuusjärjestyksessä.

Ohjelmaan pystyy rekisteröityä monta eri käyttäjää. Ohjelmassa hyödynnetään Firebase-databasea, joka päivittyy automaattisesti, kun uusi käyttäjä rekisteröidään.

Ohjelmaan on implementoitu sormenjälkitunnistin, jonka avulla käyttäjä voi kirjautua ohjelman sisään antamatta omia tunnuksiaan.

Salasanan vaihto on myös mahdollista, jos käyttäjä on sen unohtanut. Salasanan vaihto tapahtuu käyttäjän antaman sähköpostin kautta.

## Tekijät

- Thomas Kaatranen

## Ohjelman toteutus

- Millaisella teknisellä alustalla ohjelma toimii?
  - Ohjelma on suunniteltu Pixel 2 -näytöllä
  - Appi toimii ongelmitta myös Pixel 3 ja Pixel 3XL -näytöillä
  - Jos näytönkoko laskee liian pieneksi (esimerkiksi Nexus 5) appin kaikki toiminnot eivät välttämättä näy niin kuin suunniteltu, mutta toimii Pixel 2 ja suurimmilla näytöillä
- Mitä kirjastoja on käytetty?
  - Firebase
  - Fingerprint Authentication
  - Gson
  - Glide

Build.gradle:

```
dependencies {

    implementation 'androidx.appcompat:appcompat:1.4.1'
    implementation 'com.google.android.material:material:1.5.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.3'
    // Firebase
    implementation 'com.google.firebase:firebase-auth:19.2.0'
    implementation 'com.google.firebase:firebase-database:20.0.0'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'

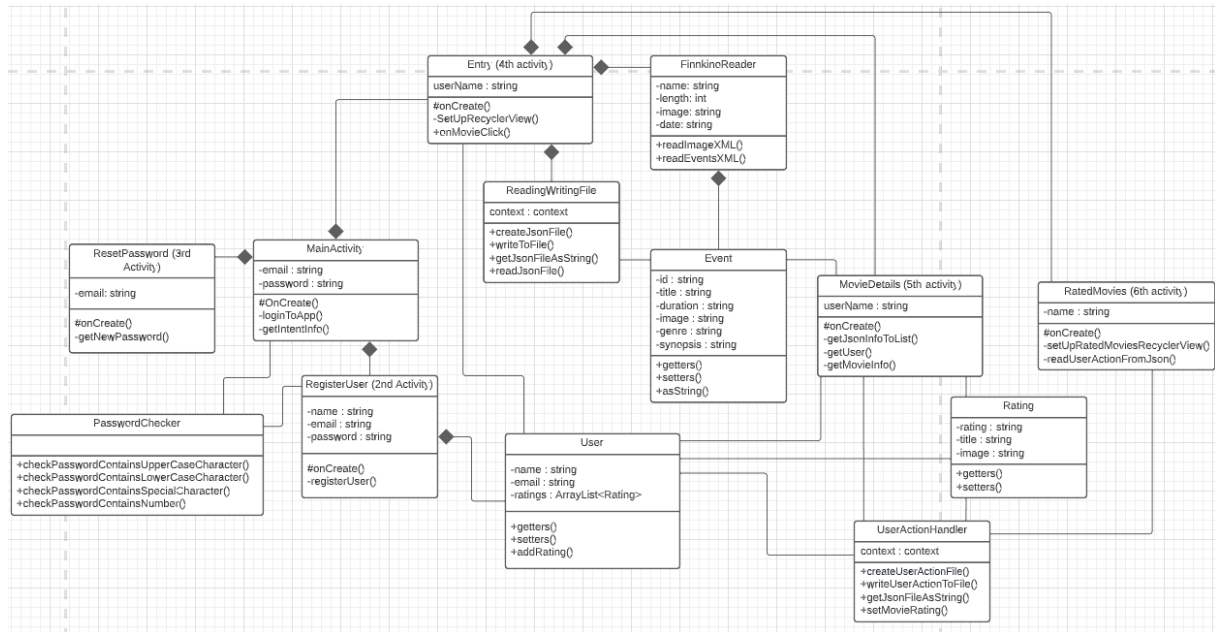
    // Fingerprint authentication
    implementation 'androidx.biometric:biometric:1.1.0'

    // Gson
    implementation 'com.google.code.gson:gson:2.8.7'

    // Glide
    implementation 'com.github.bumptech.glide:glide:4.11.0'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.11.0'
}
```

- Mitä työkaluja on käytetty?
  - Ei tietääkseni mitään.

## Luokkakaavio



## Toteutetut ominaisuudet

Ominaisuus	Huomioita	Pisteet
Olio ohjelmoitu	Pakollinen	-
Väh. 5 erilaista luokkaa	Pakollinen	-
API:n käyttö	Pakollinen	-
Sovellus tallentaa käyttäjien toiminnan JSON tiedostoon	Pakollinen	-
Käyttäjä voi selata omia arvosteltuja elokuvia	Pakollinen	13
Ohjelma on rakennettu hyvin suunnitelluista UI-komponenteista	Ohjelmassa hyödynnetään paljon erilaisia UI-komponentteja järkevästi. Osa UI-komponenteista on muokattu näyttämään paremmilta esimerkiksi tekstikentät ja jotkut painikkeet	5
Kirjautuminen applikaatioon	Toimii Firebase datan kautta. Rekisteröitymisessä hyödynnetään käyttäjän sähköpostia, josta pitää varmistaa rekisteröityminen päästäkseen appiin.	3

Sovelluksella voi olla useampi käyttäjä. Tietojen tallennuksessa hyödynnetään Firebase databasea	Päivittyy automaattisesti, kun sinne uusi käyttäjä rekisteröityy ohjelmaan	3
Salasana noudattaa hyvän salasanan sääntöjä	Sisältää vähintään yhden numeron, erikoismerkin, ison ja pienen kirjaimen, on vähintään 12 merkkiä pitkä	2
Käyttäjä voi vaihtaa salasanaa	Salasanan vaihto tapahtuu käyttäjän sähköpostin kautta	2
Kirjautuminen onnistuu biometrisellä tunnisteella	Tämä onnistuu vain, jos jokin käyttäjä on jo kirjautunut appiin, muuten tulee error.	2
Elokuvia voi etsiä nimen perusteella	RecyclerView päivittyy aina, jos käyttäjä antaa haku parametrejä, jos ei parametrejä ohjelma näyttää kaikki elokuvat	2
Käytetty Glide kirjastoa näyttääkseen kuvia ohjelmassa	Kuvat tekevät ohjelmasta miellyttävämmän ja ymmärrettävämmän käyttäjälle	2
Summa		34

## Työmäärä

Työmäärä tuntui isolta, mutta siihen varmaan vaikuttaa se, että tein harjoitustyön yksin. En osaa sanoa tarkkaa tunti numeroa.

## Mitä opin harjoitustyöstä?

- Hyödyntämään erilaisia Java kirjastoja ja, miten niitä implementoidaan android studio ohjelmaan
- Ohjelmoimaan Java:lla paremmin
- Olio-ohjelmoimaan entistä paremmin
- Suunnittelemaan android appin toimintoja ja hyödyntämään erilaisia UI-komponentteja appin rakentamisessa
- Json tiedoston käsittelyä, kuten tiedostoon kirjoitus ja sieltä lukeminen
- Opin hakemaan tietoa paremmin internetin kautta ja ratkaisemaan tyypillisimpiä Java errora.

## Palaute harjoitustyöstä

- Mitkä ominaisuudet / toiminnot olivat helppoja / vaikeita toteuttaa?
  - Toteutettavista toiminnoista helpoimmat olivat kurssin viikkotehtävissä käsitellyt asiat.
  - Vaikeinta oli toteuttaa käyttäjän toimintojen arkistointi Json tiedostoon jne.
- Oliko jokin asia aivan syvältä?
  - NullPointerException
- Oliko jokin asia todella hyvää tässä työssä?
  - Omasta mielestä työn vapaus oli hyvä. Tekijä sai itse kehittää oman appinsa.

## Huomioita

Tämänhetkiset käyttäjätunnukset Firebase databasesessa ovat:

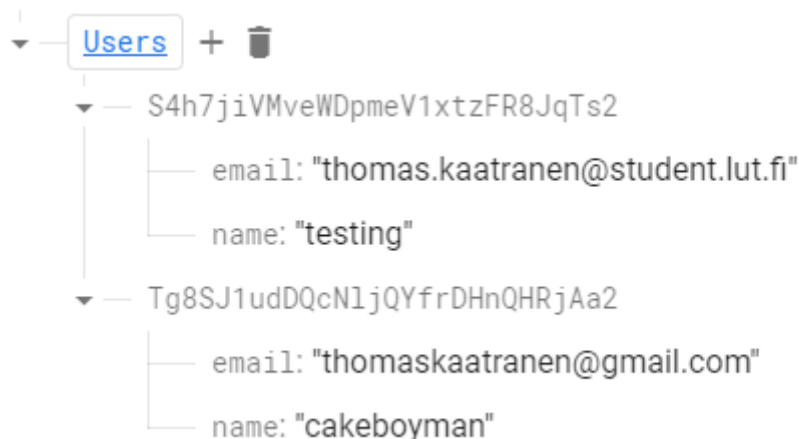
Email: thomaskaatranen@gmail.com / thomas.kaatranen@student.lut.fi (Saa käyttää)

Password: Olioohjelmointi1?

**Käyttäjä voi rekisteröidä täysin uuden käyttäjän, mutta hänen on vahvistettava tunnus annetusta sähköpostista.**

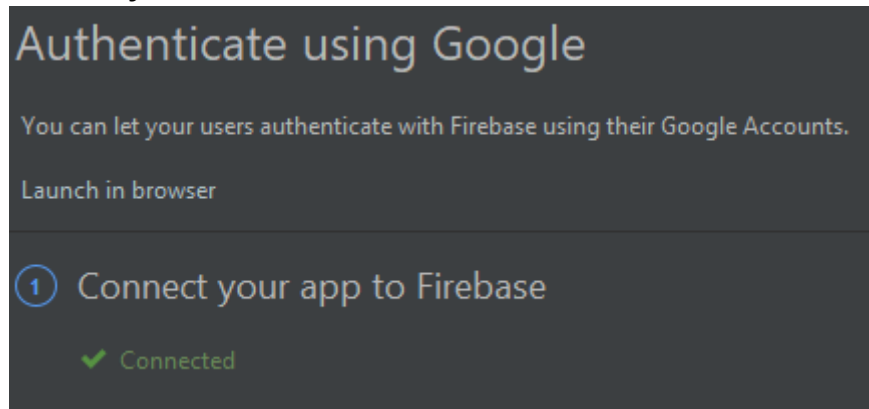
Kirjautuminen sormenjälkitunnistimella on toteutettu niin, että jos ohjelmalla ei ole tietoa sen hetkisestä käyttäjästä niin sormenjälkitunnistin ei toimi. Eli jos käyttäjä kirjautuu ohjelmasta ulos ei sormenjälkitunnistin toimi.

Firebase database:



Jos ohjelman testaus ei toimi se voi johtua, että firebase authentication ei ole päällä. Tools  
→ Firebase → Authentication → Authentication using Google

Pitäisi näyttää tältä:



Video demo: <https://drive.google.com/file/d/1M4E82YCZ91Wn7-O2yIgbpW3Ak4wiP8AR/view?usp=sharing>

Työssä on hyödynnetty seuraavia linkkejä:

<https://www.youtube.com/watch?v=3l4OShOR8jw&t=16763s>

<https://www.youtube.com/watch?v=Z-RE1QuUWPg&t=30s>

[https://www.youtube.com/watch?v=KB2Blm\\_m1Os](https://www.youtube.com/watch?v=KB2Blm_m1Os)

<https://www.youtube.com/watch?v=15WRCpH-VG0>

<https://www.youtube.com/watch?v=DRBLazxi6Eq>

<https://www.youtube.com/watch?v=-plgl1EQ21Q>

[https://www.youtube.com/watch?v=w-Uv-ydX\\_LY&t=1s](https://www.youtube.com/watch?v=w-Uv-ydX_LY&t=1s)