

# Tuloverolaskuri

Tekijät: Veeti Engblom, Emil Haapalainen ja Teemu Harinen

## Kuvaus ohjelmasta

Ohjelma sisältää tiedot Suomen yritysten verotuksista. Verotiedot ovat listattu rullattavana listana nimen perusteella, ja käyttäjä voi hakea haluamansa yrityksen tiedot kirjoittamalla yrityksen nimen hakukenttään. Käyttäjä voi napauttaa haluamaansa yritystä avatakseen laajemman sivun sisältäen yrityksen tietoja, kuten Y-tunnus, sijainti, sekä laajemmat verotiedot. Käyttäjä voi myös tallentaa haluamansa yrityksen napauttamalla suosikki nappia, jolloin yritys siirtyy erilliseen "suosikit" näkymään. Ohjelma myös sisältää light-dark moden käyttömukavuuden parantamiseksi. Rakensimme ohjelmamme verotietokannan ympärille, sillä yritysten verotiedot olivat kiinnostavia meille.

Ohjelmaan on myös sisällytetty ns. Mainoskangas, joka näyttää tällä hetkellä Finnkinossa pyörivät elokuvat satunnaisjärjestyksessä. Ohjelma hakee teatteritiedot Finnkinon avoimesta datalähteestä. Mainoskankaan sisällytys ohjelmaan oli välttämätöntä, jotta saisimme yhden API:n käytön sisällytettyä ohjelmaan. Tämä johtuu siitä, että verotietojen hakeminen reaaliajassa ei onnistunut, vaan vaadittava data on pakattu ohjelman mukana.

## Tekijät

Veeti Engblomin tehtävänä oli laskennalliset toiminnallisuudet, haku toiminnallisuus, sekä suosikkeihin lisäämis- toiminnallisuus. Teemu Harinen puolestaan hoiti datan ja tiedostojen luvut sekä Finnkinon mainoksen implementoinnin käyttäen Finnkinon APIa. Emil Haapalaisen vastuulle puolestaan jäi UI-komponenttien kasaus kuten sivuvalikon rakentaminen ja teemojen vaihtaminen, sekä dokumentointi ja versionhallinnan hoito.

## Ohjelman toteutus

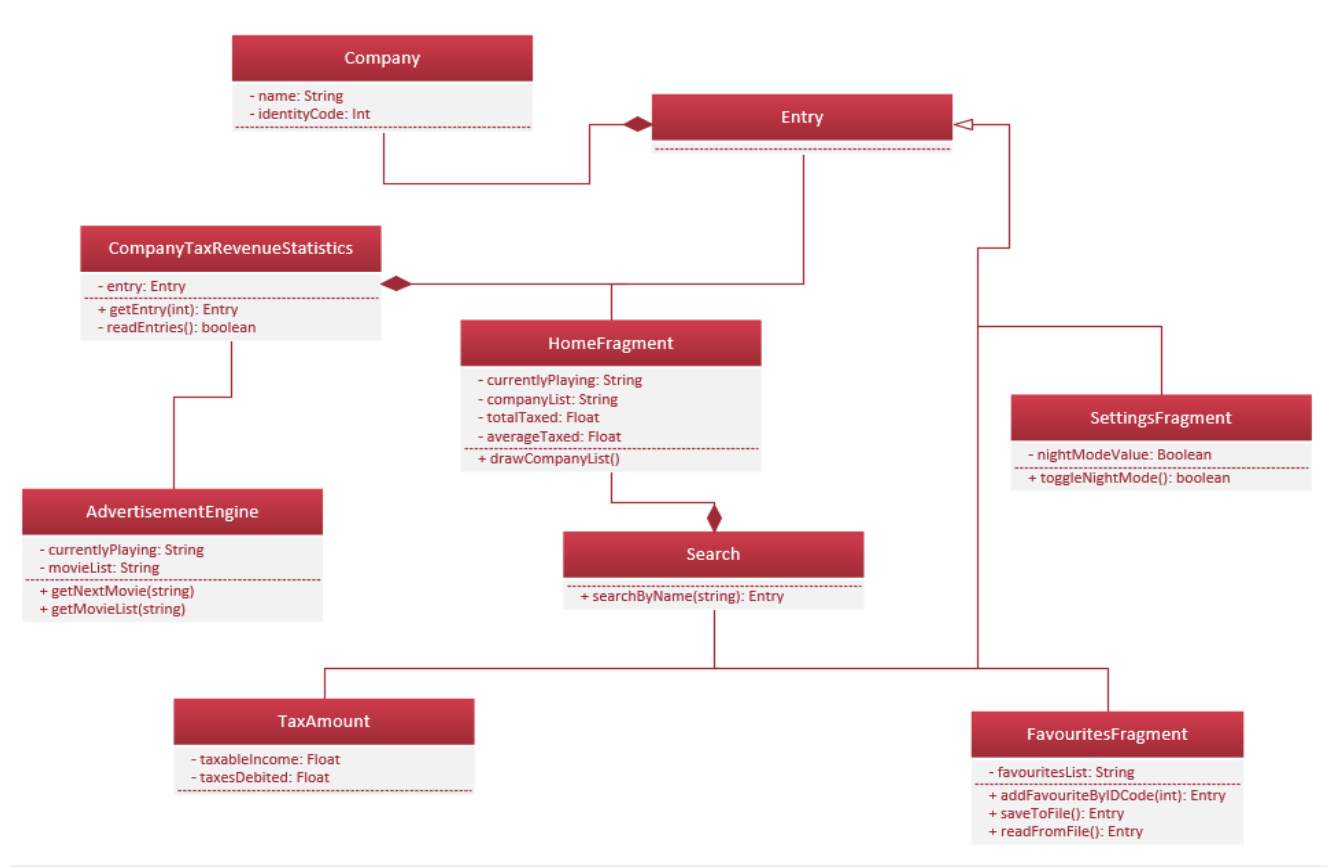
Ohjelma on rakennettu Android SDK-versiolle 32 (Android 12L). Minimi tuettu versio on 21 (Android 5.0).

Niin sanotusti erikoisempia käytettyjä kirjastoja (mm. XML lukeminen):

```
org.w3c.dom.Document;  
org.w3c.dom.Element;  
org.w3c.dom.Node;  
org.w3c.dom.NodeList;  
org.xml.sax.SAXException;  
javax.xml.parsers.DocumentBuilder  
javax.xml.parsers.ParserConfigurationException
```

Ohjelman toteutuksessa käytimme Discordia kommunikointiin ja githubia versionhallintaa. Ohjelmistokehitysympäristönä käytimme Android Studiota ja testauksessa emulaattoria sekä oikeaa puhelinta (adb). Dokumentointityökaluna toimi MS-Word.

## Luokkakaavio



## Toteutetut ominaisuudet

Ominaisuus	Perustelut	Pisteet
Olio-Ohjelmoitu	Pakollinen	Pakollinen
Väh. 5 erilaista luokkaa & oliota	Pakollinen	Pakollinen
Väh. yhden API:n käyttö	Pakollinen, applikaatio käyttää Finnkinon datalähteenä, josta se hakee tällä hetkellä teattereissa pyörivät elokuvat ja näyttää ne etusivulla mainoksena	Pakollinen
Tallentaa käyttäjän suosikit .csv tiedostoon	Pakollinen	Pakollinen

Käyttäjä voi tarkastella tallentamiaan tietoja myöhemmin	Pakollinen	Pakollinen
Ohjelma on rakennettu hyvin suunnitelluista UI-komponenteista	Hyvin suunnitellut ja toteutetut UI-komponentit ovat tärkeitä ohjelman käytön kannalta	4
Jokin toinen datalähde implementoituna	Yritysten tuloverotuksen julkisten tietojen hakeminen ja näyttäminen applikaatiossa. Yritysten nimi, sijainti, Y-Tunnus, verotettu tulo, maksetut verot	5
Fragmenttejen hyödyntäminen aktiviteettien sijasta käyttöliittymiä rakentaessa	Käyttöliittymiä rakentaessa oli mielestämme järkevämpää käyttää fragmenttejä. Halusimme myös oppia käyttämään fragmenttejä	2
Responsiivinen käyttöliittymä	Sovelluksen toimiminen eri kokoisilla ruuduilla on välttämätöntä hyvän käyttökokemuksen turvaamiseksi. Sovellus ei toimi landscape-tilassa	2
Light-Dark mode	Dark mode vähentää silmien rasitusta ja näyttää mielestämme tyylikkäältä. Tästä syystä lisäsimme sovelluksen asetuksiin kytkimen, jolla voi vaihtaa näiden kahden teeman välillä	2
Dynaaminen hakukenttä kotisivulla, joka päivittää listaa samalla kun käyttäjä kirjoittaa yrityksen nimeä kenttään	Verotietolista on liian suuri siihen, että käyttäjä voisi helposti listaa selatessa löytää haluamansa yrityksen tiedot. Tästä syystä lisäsimme kotisivulle hakukentän.	3
Singleton	Luodaan lista ja kutsutaan sitä singletonin avulla	2
Sivuvalikko	Lisäsimme käyttöliittymään sivuvalikon, josta pystyy siirtymään fragmenttien välillä (Home, Favourites, Settings)	4
Summa		37

## Työmäärät

Tekijä	Tehtävät	Tunnit
Veeti Engblom	Suunnittelu, Devaus (UI komponentit, laskenta), testaus	35

Emil Haapalainen	Devaus (UI komponentit), VSC hoito, suunnittelu, testaus, dokumentointi	30
Teemu Harinen	Devaus (UI komponentit, datan käsittely (CSV ja XML luku), testaus	35
Summa		100

## Mitä opin harjoitustyöstä?

Emil Haapalainen: Opin rakentamaan paremman näköisiä käyttöliittymiä, kuten sivuvalikon lisäys, iconien käyttö käyttöliittymässä. Opin rakentamaan käyttöliittymiä fragmenttien avulla ja ymmärsin ylipäättään miten fragmenttejä käytetään. Opin prosessoimaan suuria määriä dataa, sekä avoimen datan haku API:n avulla.

Teemu Harinen: Opin paljon fragmenttien käytöstä ja mm. Niiden rajoituksista. CSV-tiedoston käsittely oli myös uutta Javalla ja Android studiolla. CSV-datan suuren määrän vuoksi joutui hieman myös kikkailemaan emulaattori asetuksien kanssa, mikä on sitten hyvä tulevaisuuden projekteille, jos ja kun tulee samankaltaisia ongelmia. Opin myös paljon luokkien välisestä tiedonsiirrosta kantapään kautta (private, public yms.), saattoivat aiheuttaa muutaman kerran ongelmia.

Veeti Engblom: Opin paljon uutta Android studion käytöstä sekä Javalla koodaamisesta. Opin myös versionhallintaa. Projekti oli ensimmäinen ryhmässä tehtävä koodaus projekti, joten versionhallinta tuotti alussa hieman haasteita. Opin tiedoston käsittelyä (CSV), fragmenttien käyttöä. Opin myös avoimesta rajapinnasta asioita.