

Oljo-ohjelmointi harjoitustyö: Yliopiston ruoka-arvosteluapplikaatio

Jesse Pasanen 0545937

Jesse Mustonen 0541805

Vili Raunola 0543366

Taneli Mykrä 0560909

Kuvaus

Ohjelma on Yliopistojen ruoka-arvosteluapplikaatio, josta näkee useiden eri yliopistojen eri ravintoloiden päivittäisiä ruokalistoja. Ruokat sisältävät käyttäjäkohtaisia arvosteluja, joita pystyy tekemään lisää käyttäjätileillä. Ruoka arvosteluja voi lisätä vain ruuille, jotka ovat olleet aiemmin tarjolla, ja käyttäjä voi lisätä arvostelun vain kerran samalla ruoalle. Sovelluksessa löytyy sekä normaaleja, että Admin -käyttäjiä. Admin -käyttäjillä on lisäominaisuuksia, kuten muiden käyttäjien arvosteluiden poistaminen ja piilottaminen, sekä kaksivaiheisen tunnistautumisen ohittaminen. Käyttäjät voivat tarkastella tekemiään arvosteluja "Omat arvostelut" -sivulta, jossa näkyvät käyttäjän jo julkaistut arvostelut ja arvostelu luonnokset. Arvosteluja voidaan myös muokata tältä sivulta. Ne voidaan myös asettaa julkaistuksi tai takaisin luonnoksiksi. Kuitenkaan arvosteluja, joiden kaikki arviointikriteerit ovat 0 ja tekstikenttään ei ole syötetty mitään, ei voi julkaista. Sovelluksen "Tili" -sivulla voidaan tarkastella käyttäjän tietoja ja muokata niitä. Tietoja vaihtaessa vaaditaan vanha salasana. Salasanaa muuttaessa ovat samat kriteerit kuin käyttäjätiliä luodessa, jotka ovat merkitty tekstikentän yläpuolelle. "Asetukset" -sivulta löytyy kielenvaihto suomen ja englannin välillä.

Kuka teki, mitä teki:

Arvostelut: Raunola, Pasanen

Navigointi: Raunola, Mustonen

Iopullinen UI: Mykrä

Salaus: Mykrä

Json kirjoitus ja luku: Kaikki

Xml kirjoitus ja luku: Raunola, Pasanen

Account creation, login ja authentication: Mustonen, Mykrä

User: Mykrä, Raunola, Pasanen

Kielen vaihto: Mustonen

Käännökset: Mykrä

Food review: Mykrä, Raunola, Pasanen

University, ParseClass: Raunola, Pasanen

Olioiden siirto fragmenttien välillä: Raunola, Pasanen

DateClass: Pasanen

Sorting: Mustonen, Mykrä

Ominaisuuksia, joita ei ole mainittu listassa kaikki ovat tehneet tasapuolisesti

Sovellukselle tehtiin alun perin yksinkertainen suunnitelma ominaisuuksineen ja luokkakaavioineen ja toteutus lähtikin alunperin suunnitelman ja sen jälkeen tulleiden ideoiden pohjalta. Kävi kuitenkin nopeasti ilmi, että alkuperäinen suunnitelma ei ollut kovinkaan perinpohjainen, sen tekemisen aikaisen vähäisemmän tietämyksen ja osaamisen vuoksi. Toteutus tapahtui loppupuoleissa "trial and error" -tyylillä. Ryhmämme työskentely toimi kuitenkin hyvin ja kaikilla oli hyvä kuva siitä mitä pitää tehdä, joten vajaavaisesta suunnittelusta ei lopulta ollut paljoakaan haittaa ja työskentely sujui hyvin. Toteutuksessa hyödynsimme Android Studiota sovelluksen tekemiseen ja GitHub:ia sen versionhallintaan.

[illegible]

<https://drive.google.com/file/d/1N7ht-xhSrlOrntdsIa6EeLQSt8kKtGTZ/view?usp=sharing>

Pakolliset:

- olio-ohjelmoitu
- Vähintään viisi erilaista luokkaa & oliota (käyttöliittymäluokkia ei lasketa)
- Tärkeiden tietojen kirjoitus tiedostoon (XML, JSON tai CSV)
- Voi muokata käyttäjän tietoja (nimi, yhteystiedot, jne.)
- Voi tehdä arvosteluita
- 1. Anna numeerinen arvio ruoalle (1-5, 1-10, useampi arviointikriteeri, jne. saatte päättää itse)

- 2. Anna kirjallinen arvio ruoalle (vapaasti syötettävä teksti)
- Voi muokata arvostelua ennen lähetystä (eli arvostelussa on tallenna JA lähetä painikkeet)
- Voi tarkastella ruokalistoja (eri ravintolat, eri päivät, useampi ruoka per päivä, ruokalistat voitte tehdä itse)
- Voi tarkastella ruokalistoja eri ravintoloista ja niiden arvosteluita

Yleiset:

- Tietojen **kirjoitus ja lukeminen** tiedostosta (XML, JSON tai CSV) (muu muoto kuin pakolliseksi listatussa ominaisuudessa)
- Ohjelma on rakennettu hyvin suunnitelluista UI-komponenteista
- Admin-käyttäjä
- Useampi käyttäjä (ja niiden luominen), tietojen tallennus järkevästi jonnekin
- Kirjautuminen applikaatioon
- Kirjautumisen salasana noudattaa hyvän salasanan sääntöjä (sisältää vähintään yhden numeron, erikoismerkin, ison ja pienen kirjaimen, on vähintään 12 merkkiä pitkä). ja Kirjautumisen jälkeen, ruudulle tulee myös satunnainen 6-numeroinen luku, joka käyttäjän pitää syöttää (simuloi avainlukukoodia).
- Salasanan tallennus käyttää jonkinlaista hash-menetelmää ja suolausta (esim SHA-512 + salt)
- Jokin oma hieno ominaisuus tai toiminto (tai useampi)
 - Kielen vaihto
 - Admin ominaisuudet (ei avainlukua, poista ja piilota arvosteluja)
 - Käyttäjän nimi näkyy side menussa
 - Käyttäjä ei voi julkaista samalle ruoalle toista arvostelua
 - Yläpalkin otsikko muuttuu riippuen sovelluksen tilasta
 - Käyttäjä voi palata nykyiseen päivään ruokalistoja läpikäydessään

Aiheesta:

- Voi muokata / poistaa arvioita (muokkaus voi sisältää esim. 5 minuutin aikarajan)
- Arvosteluita voi antaa vain kun ruoka on ollut saatavilla
- Voi suodattaa / järjestää arvosteluita
- Voi äänestää muiden arvosteluita (esim. oliko tämä hyödyllinen vai ei)
- Järjestelmässä voi olla useampi yliopisto, joilla kaikilla on omat ravintolansa ja ruokansa. Käyttäjän tiedoissa näkyy mikä on käyttäjän "koti" yliopisto
- Ruokalistoista tai ruoista voi tehdä "vakioita" eli samat listat voivat toistua esim. kerran kuussa tai ruoat joka kolmas viikko joku päivä.

Pistearvio ominaisuuksien perusteella

40 (olettaen, että omista ominaisuuksista tulisi 4 pistettä)

Työmääräarvio

Harjoitustyön tekemiseen kului 2 viikkoa jokapäiväistä työstämistä. Työhön kului aikaa kullakin ryhmän jäsenellä 5-10 tuntia per päivä eli yhteensä +100 tuntia.

Mitä opittiin:

Vili Raunola

Harjoitustyön aikana opin käyttämään versionhallinta ohjelmaa GitHub. Opin myös paljon erityylisten tiedostojen lukemisesta ja kirjoittamisesta Javalla. Opin myös hyödyntämään olio-ohjelmoinnissa käytettyä singleton luokkaa järkevänä osana ohjelmistoa sekä siirtämään olioita fragmenttien välillä. Yleinen kommentti: Stack Overflow on erittäin hyödyllinen ongelmien ratkaisuun.

Jesse Pasanen

- Opin käyttämään GitHubin versionhallintaa.
- Opin käyttämään Android Studion UI elementtejä.
- Opin kuinka sovellusten käännösversiot eri kielille käytännössä toimii.
- Opin käyttäjätilien toiminnasta puhelinsovelluksissa.
- Opin ymmärtämään kommentoinnin tärkeyden koodatessa.
- Kurssilla annetut ohjeet varsinkin hyvään ohjelmointityyliin eivät olleet riittävät.

Taneli Mykrä

Opin harjoitustyössä miten ohjelmointi käytännössä tapahtuu osana pientä ryhmää. Opin myös miten salasanojen salaus toimii käyttämällä SHA-512 + salt salausta. Opin paljon javan, Android Studion ja GitHubin käytöstä. Harjoitustyö oli aikaa vievä, mutta suoriuduimme siitä suurimmalta osin ongelmitta.

Jesse Mustonen

Opin miten versionhallintaa käytetään Githubin avulla ja kuinka se on oikeasti tärkeä työkalu kun jotain menee pieleen. Opin myös, että lopullinen koodi on usein monen iteraation lopputulos ja usein ensimmäinen ja viimeinen versio eroavat hyvin paljon toisistaan. Opin, että onClick:in käyttäminen napeissa on paljon hyödyllisempi, selkeämpi ja nopeampi tapa tehdä napin toiminen kuin onClick:n lisääminen layout.xml tiedostossa. Yleinen kommentti: YouTubessa on erittäin hyviä tutoriaaleja android studion käyttämiseen varsinkin Coding in Flow.

Bonus:

- Hyvän ohjelmointityyli opetus varsinkin Android ympäristössä, mutta myös olio-ohjelmoinnissa yleisesti.
- Vaikeaa toteuttaa:
 - Json parsiminen
 - Spinner joka määrää toisen Spinnerin sisällön(oma, ei vaadittu)
 - onBackPressed()-metodin saaminen toimimaan halutulla tavalla(oma, ei vaadittu)
 - Version control oli vaikeaa toteuttaa ryhmässä joten voisi olla erillinen pieni opetusvideo asiasta

Ohjeita sovelluksen käyttämiseen

Sovelluksessa olevia testikäyttäjiä:

Admin -käyttäjä

Email: admin

Salasana: admin

Email: user@test.com

Salasana: T3stuserpass!!

Email: user1@test.com

Salasana: T3stuserpass!!

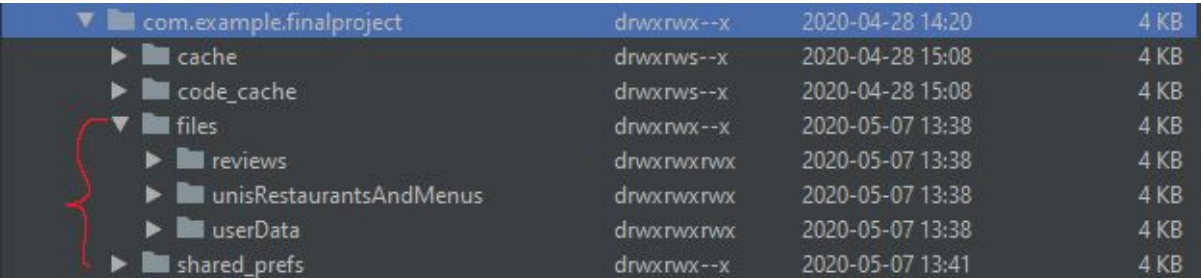
Email: user2@test.com

Salasana: T3stuserpass!!

Mahdollista tehdä myös uusia sovelluksen kautta.

Assteissa olevat tiedostot tulee siirtää puhelimen muistiin ohjelmalle dedikoituun "Files" kansioon sovelluksen toiminnan mahdollistamiseksi.

Path: /data/data/com.example.finalproject/files/



▼ com.example.finalproject	drwxrwx--x	2020-04-28 14:20	4 KB
▶ cache	drwxrws--x	2020-04-28 15:08	4 KB
▶ code_cache	drwxrws--x	2020-04-28 15:08	4 KB
▼ files	drwxrwx--x	2020-05-07 13:38	4 KB
▶ reviews	drwxrwxrwx	2020-05-07 13:38	4 KB
▶ unisRestaurantsAndMenus	drwxrwxrwx	2020-05-07 13:38	4 KB
▶ userData	drwxrwxrwx	2020-05-07 13:38	4 KB
▶ shared_prefs	drwxrwx--x	2020-05-07 13:41	4 KB

Linkki GitHub:iin

<https://github.com/ViliRaunola/Finalproject>

Sovellusta tehdessä käytetyt lähteet

Side menun tekeminen:

<https://www.youtube.com/watch?v=G9yi92qdxBg>

Uuden aktiviteetin tekeminen (part 1 - 4):

<https://www.youtube.com/watch?v=fGcMLu1GJEc>

Salauksen tekeminen:

<https://howtodoinjava.com/security/how-to-generate-secure-password-hash-md5-sha-pbkdf2-bcrypt-examples/>

Salasanan oikeanlainen tyyli

<https://stackoverflow.com/questions/1795402/check-if-a-string-contains-a-special-character>

Json tiedostoon kirjoittaminen

<https://howtodoinjava.com/library/json-simple-read-write-json-examples/>

Nykyisen päivämäärän parsiminen:

<https://stackoverflow.com/questions/5175728/how-to-get-the-current-date-time-in-java>

Päivämäärän muuttaminen:

<https://stackoverflow.com/questions/428918/how-can-i-increment-a-date-by-one-day-in-java>

Päivämäärien vertailu:

<https://www.tutorialspoint.com/how-to-compare-two-dates-in-java>

Olioiden siirtäminen fragmentista toiseen:

<https://stackoverflow.com/questions/42266436/passing-objects-between-fragments>

Käyttäjän nimi side menuun:

<https://stackoverflow.com/questions/34973456/how-to-change-text-of-a-textview-in-navigation-drawer-header>

<https://stackoverflow.com/questions/32806735/refresh-header-in-navigation-drawer/35952939#35952939>

Activityn hyödyntäminen fragmentissa:

<https://stackoverflow.com/questions/9343241/passing-data-between-a-fragment-and-its-container-activity>

Kielen vaihtaminen:

<https://www.youtube.com/watch?v=zILw5eV9QBQ>

<https://stackoverflow.com/questions/43092248/how-to-get-application-language-and-device-language-separately-in-android>

XML tiedostoon kirjoittaminen:

<https://stackoverflow.com/questions/17022221/openfileoutput-how-to-create-files-outside-the-data-data-path>

(Myös opettajan youtube video)

Sorting:

https://www.youtube.com/watch?v=Mguw_TQBExo

<https://stackoverflow.com/questions/9941890/sorting-arraylist-of-objects-by-float>

Fragment transaction:

<https://developer.android.com/reference/android/app/FragmentManager>