

Vesistöjen veijarit

Tekijät Jan-Peter Kauppinen, Matti Lankinen ja Valtteri Lausala

Kuvaus ohjelmasta

Vesistöjen veijarit on jokaisen vesitieteilijän, lintuharrastajan, sekä luontoihmisen suosikki Android-applikaatio. Ohjelman tarkoituksena on toimia tietopankkina jokaiselle järvi-intoilijalle, sekä tarjoaa mahdollisuuden kirjata havaintojaan ylös ohjelman muistiin. Käyttäjä pystyy jälkikäteen tarkastelemaan kirjaamiaan havaintoja synkkänä talvi-iltana lämpimän takkatulen äärellä kaakaokuppi kädessään.

Mitä ohjelma sisältää

Ohjelma avautuu kirjautumisnäyttöön. Kirjautumisnäytöstä käyttäjä voi syöttää oman käyttäjätunnuksensa, sekä salasanan erillisiin laatikoihin. Aloitusnäytöstä löytyy myös mahdollisuus siirtyä kolmeen eri näyttöön.

Sign in-nappulasta käyttäjä pääsee kirjautumaan käyttäjätilillensä syötettyään oikean salasanan ja käyttäjätunnuksen. Change language-nappulasta käyttäjälle aukeaa valikko, josta hän voi vaihtaa kieltä. Painamalla kielivalinnassa kielten edessä olevia palloja, käyttäjä voi vaihtaa kieltä suomen ja englannin kielen välillä. Painalluksen jälkeen kielivalinta päivittyy koko appiin ja säilyy, kunnes käyttäjä seuraavan kerran vaihtaa valintaa.

Näytön alalaidasta löytyy pitkä teksti, jonka kautta uusi käyttäjä voi luoda uuden käyttäjätunnuksen. Painalluksen jälkeen käyttäjä ohjataan uudelle Create account-näyttö, jossa on kolme laatikkoa username ja kaksi password laatikkoa, sekä Register here-painike. Käyttäjä syöttää käyttäjätunnuksen kerran ja salasanan kahteen kertaan. Salasanojen tulee olla identtisiä ja vahvasti suojattuja (sisältää vähintään 12 merkkiä ja tulee sisältää numeroita, kirjaimia, sekä erikoismerkkejä). Käyttäjälle ilmoitetaan salasanan hylkäämisestä, sekä rekisteröinnin onnistumisesta keskelle näyttöä.

Kirjautumisen jälkeen käyttäjälle avautuu näyttö, jossa on listaus kaikista Suomen järvistä, sekä kaksi painiketta. Ensimmäinen painikkeista (REFRESH) päivittää listan ja toinen painikkeista (INPUT INFORMATION) ohjaa käyttäjän kommenttien tekemiseen luodulle näytölle. Listan avulla käyttäjä pystyy tarkastelemaan tietoja suomalaisista järvistä. Listauksessa on esiteltynä järven nimi, sijaintikunta, rantaviivan pituus, keskimääräinen syvyys, sekä järven pinta-ala.

Kommentit näytöltä käyttäjä pystyy kirjaamaan ylös järven nimen ja siihen liittyvän kommentin, tallentamaan kommentin sovelluksen muistiin SAVE-nappulasta, sekä selaamaan aikaisemmin tekemiään kommentteja painamalla BROWSE COMMENTS-nappulaa. BROWSE COMMENTS-nappula aukaisee käyttäjälle näytön, johon on tallennettu kaikki käyttäjän luomat kommentit kronologiseen järjestykseen. Ohjelma tallentaa kommentit käyttäjäkohtaisesti SQLite:n avulla, joten kommentit näkyvät ainoastaan kyseiselle kirjautuneelle käyttäjälle.

Miksi asiat on sisällytetty ohjelmaan?

Ohjelmaan on sisällytetty ryhmämme mielestä kaikista tärkeimmät ominaisuudet toimivan luontosovelluksen kannalta. Alun henkilökohtaisen tunnistautumisen koimme olevan äärimmäisen tärkeä ominaisuus käyttäjien erottelun ja tietojen oikeaoppisen säilömisen kannalta. Käyttäjätunnuksen luomisessa emme kysyneet käyttäjän nimeä/muita henkilötietoja vaaliaksemme käyttäjien yksityisyyttä. Salasanan osalta tahdoimme luoda jokaiselle käyttäjälle turvallisen salasanan, sekä varmistaa salasanan muistaminen kirjaamalla se kahdesti. Turvallinen salana on edellytys käyttäjän tietojen säilymiseen ja tästä syystä on pakollinen ominaisuus sovellukselle.

API:n kautta tuleva järvidata oli pakollinen ominaisuus harjoitustyölle, mutta samalla hyödyllinen lisä käyttäjälle. Esitetyt tiedot toimivat perustana käyttäjän yleissivistyksen kasvattamiselle, sekä antavat tärkeää tietoa käyttäjän kiinnostuksen kohteena olevista järvistä. Tiedot järven pinta-alasta, sijainnista ja keskisyyvyydestä toimivat hyvänä lisänä järvellä liikkuville, kalastaville, sekä kumppaniansa aamupalan aikana mielenkiintoisilla faktoilla sivistäville henkilöille.

Kommenttiominaisuuden koimme myös äärimmäisen tärkeäksi käyttömukavuuden kannalta. Käyttäjä voi ominaisuuden avulla tallentaa tietoja tietyistä järvistä ja kerätä itseään kiinnostavimpia faktoja kommenttikenttään. Kommenttikenttää voi hyödyntää myös järvellä vierailukalenterina, kalasaaliin tallentajana, tai järvien arvosteluun käytettävänä muistiona.

Kehitysehdotuksia tulevaan

Sovellus tarjoaa käyttäjälle lukuisia käytännöllisiä ominaisuuksia, mutta kehitettävää siitä löytyy vielä paljon, eikä se ole missään nimessä tuotantovalmis. Sovelluksen perustoiminnot ovat hyvällä tasolla, mutta käyttäjää avustavia ominaisuuksia voitaisiin luoda lisää. Tarkastelimme sovelluksemme toimintoja kriittisesti ja kasasimme tähän kappaleeseen lukuisia ehdotuksia sovelluksen tulevaisuuden kehittämiseen.

Suurimpana kehityskohtana apin osalta voidaan pitää datan pyörittämistä verkon kautta. Tällä hetkellä database on lokaali, joten sovellusta voidaan käyttää ainoastaan yhdeltä laitteelta. Tämä ominaisuus pilaa sovelluksen tämänhetkiset maailmanvalloitus suunnitelmat.

Sovelluksessa olisi hyvä olla myös muita järvi-intoilijoita avustavia ominaisuuksia. Kommenttiosioon voisi lisätä spinnerin, jonka avulla järven nimen voisi valita helposti. Kommenttiosion kautta voisi olla hyvä myös lisätä kuvia järveltä, sekä antaa arvosteluja ja lisätä suosikkeja listaan. Jonkin asteinen kommenttien hakutoiminto voisi olla myös kätevä.

Käyttäjää voitaisiin myös avustaa mahdollistamalla hänelle salasanan vaihtaminen. Tämä kuitenkin vaatisi tarkempien tietojen varmistamisen käyttäjältä, sekä esimerkiksi

sähköpostivarmistuksen implementoimista sovellukseen. Käyttäjän kannalta toinen hyödyllinen ominaisuus olisi Google Mapsin lisääminen käyttäjien hyödynnettäväksi. Sen avulla järvien hakeminen olisi yksinkertaista ja käyttäjien sovelluksesta saavuttama hyöty kasvaisi.

Tekijät

Työ jakautui ryhmäläistemme osaamistason, kiinnostuksen ja elämäntilanteet huomioon ottaen melko tasaisesti ja roolitus saatiin luotua yhteisymmärryksessä. Päävastuussa koodauksesta oli Valtteri, jolla oli selkeästi eniten kokemusta ryhmäläisistämme olio-ohjelmoinnista. Valtteri hoiti siis haastavimpien kokonaisuuksien toteuttamisen, sekä yhdisteli Githubissa muiden tekemiä luokkia yhdeksi yhtenäiseksi ohjelmaksi. Valtterin päävastuuna oli kirjautuminen, kaksi adapteria, SQLiten implementointi projektiin, sekä responsiivinen käyttöliittymä.

Jan-Peter ja Matti toimivat apuna ideoinnissa ja tiedonhaussa haastavampien ominaisuuksien osalta. He auttoivat Valtteria myös ongelmien ratkaisussa, sekä niiden luomisessa. Jan-Peterin vastuulle jäi JSON-luokan ja kaksikielisyyden luominen. Matille tehtäväksi jäi vahvan salasanan sekä järvi- ja kommenttioloiden luominen. Jan-Peter ja Matti vastasivat yhdessä myös sovelluksen visuaalisesta puolesta ja suunnittelivat layout-tiedostoja. Kyseiset sankarit ottivat myös isompaa roolia dokumentaation ja työn kommentoinnin osalta.

Pidimme viikoittain nopeita kokouksia missä tarkastelimme työn edistymistä ja päätimme että mitä tehdään seuraavaksi. Näissä kokouksissa päätimme myös ominaisuuksien toteuttamisesta. Tämän lisäksi ryhmämme jäsenet pitivät yhteisiä koodaus- ja ongelmanratkaisutuokioita, joissa ratkottiin esille nousseita ongelmia porukassa.

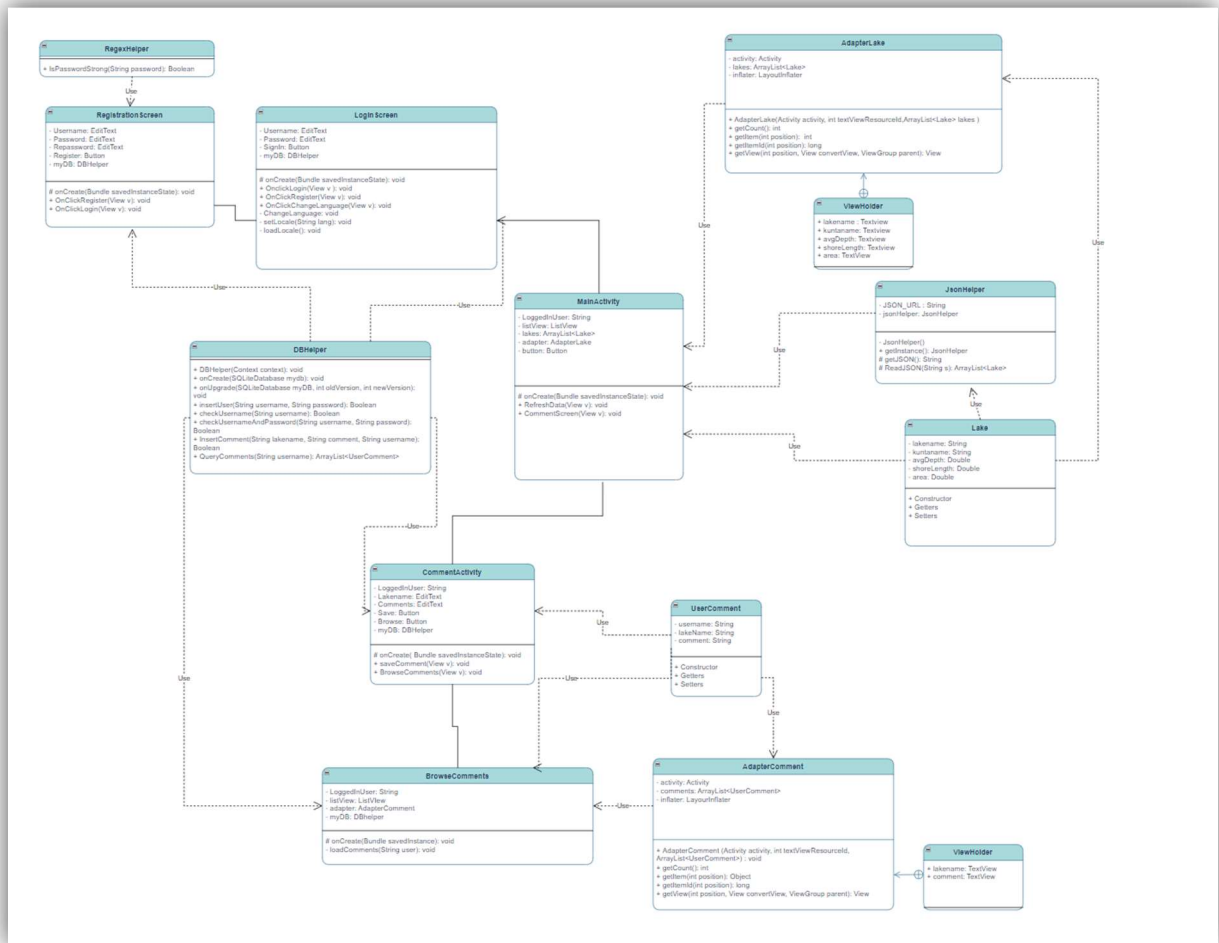
Työn alkuvaiheessa totesimme alkuperäisen suunnitelman mukaisen elokuvat-sovelluksen olevan tylsä toteuttaa, joten päätimme vaihtaa projektia lennosta. Tämä johti sovelluksen uudelleen suunnitteluun, mutta mahdollisti tekijöitä laajemmin motivoivan työn tekemisen. Lopputulokseen olemme kutienkin tyytyväisiä.

Ohjelman toteutus

Sovelluksemme toimii Android sekä Java alustalla. Työssä on käytetty muun muassa Androidin default kirjastoja, Java.generics, Java.io ja Java.collections kirjastoja. Työtä tehdessämme olemme hyödyntäneet AndroidStudio lisäksi lukuisia ryhmätyökaluja, kuten Github Desktop, Whatsapp ja MS Teams. Githubin kautta keräsimme ja jaoimme koodia. WhatsApp ja Teams toimivat keskustelu- ja kokousalustoina työn tekemisen aikana.

Projektiä toteuttaessamme emme hyödyntäneet testaustyökaluja, vaikka niistä olisi varmasti ollut hyötyä. Testauksemme pohjautui koodarien itse toteuttamaan testaukseen työn edetessä. Dokumentoinnissa olemme hyödyntäneet Wordia, sekä Githubia.

Luokkakaavio



Luokkakaavio Android-sovelluksen toiminnasta, Suurempi kaavio löytyy githubista

Toteutetut ominaisuudet

Ominaisuus	Perustelut	Pisteet
Olio-ohjelmoitu	CT60A2411 Olio-ohjelmointi	-
Min. 5 erilaista oliota	pakollinen	-
Järvirajapinta	pakollinen	-
Tallentaa käyttäjän toiminnan	pakollinen	-
Tallennettujen tietojen tarkastelu	pakollinen	-
UI-komponentit	Mm. Listaa olisi voinut parantaa, kommentteja filtteröidä ja yleisesti tehdä käyttäjäystävällisemmäksi	3
Kirjautuminen aplikaatioon	Kirjautumisessa ei ole ongelmia	3
Useampi käyttäjä ja niiden tietojen tallennus	Useamman käyttäjän luominen onnistuu ja tiedot tallennetaan järkevästi	3
Kirjautumisen salasana	Salasana täyttää kaikki vaatimukset (erikoismerkit, pituus...)	2
Responsiivinen käyttöliittymä	Ruudun koko ei vaikuta ohjelman toimivuuteen. Käyttäjäkuntaa saadaan kasvatettua ja käyttömukavuutta lisättyä merkittävästi.	2
Käyttöliittymä usealla kielellä	Ohjelmasta voi valita kieleksi suomen tai englannin. Mahdollistaa englannin kieltä osaaville turisteille sovelluksen hyödyntämisen.	2
Hieno ominaisuus: Sqlite database	JSON tai XML:än sijaan hyödynsimme ohjelmistoyrityksille tuttua Sqlite databasea tietojen tallentamiseen	5
Summa		33

Työmäärät

Tekijä	Tehtävät	Tunnit
Valtteri Lausala	Master developer, testaus, ongelmien ratkominen	60
Jan-Peter Kauppinen	Junior developer, Layout developer , dokumentaatio, ongelmien aiheuttaminen	49
Matti Lankinen	Baby developer, Layout developer , dokumentaatio, ongelmien aiheuttaminen	30
Summa		114

Henkilökohtaisista syistä Matin osuus jäi tavallista pienemmäksi. Tarvittaessa voin avata aihetta enemmän. Asiasta keskusteltu ryhmän sisällä ja tilanne kaikille ok.

Mitä opin harjoitustyöstä?

Valtteri: Opin taistelemaan frameworkkiä vastaan, pyörittämään tiimiä + isompaa projektia ja android sovellusten luontiprosessin

Jan-Peter: Opin harjoitustyötä tehdessäni melko laaja-alaisesti AndroidStudioon käytöstä, sekä olio-ohjelmoinnista Javalla. Työ kehitti erityisesti tiedonhakutaitoa ohjelmointiongelmien ratkaisuun liittyen, sekä tunnistamaan omia virheitä komentorivien virhekoodien perusteella. Tiedonhakutaito kehittyi myös etsiessä esimerkkejä tehtävänä olevaa ominaisuutta vastaavista ominaisuuksista. Ohjelmistokehityksestä työ opetti myös sen, että projekti voi muuttua merkittävästikin alun perin suunnitellusta, sekä muutoksiin on vain pakko mukautua. Hyvällä suunnittelulla helpotetaan omaa työntaakkaa merkittävästi.

Matti: Opin todella paljon Javasta yleisesti ja AndroidStudioon käytöstä. Pidin itse harjoitustyön ideasta eli siitä että koodataan (mahdollisesti (50/50)) toimiva android sovellus. Työ opetti paljon siitä, että mitä oikeasti sovellusten tekeminen vaatii. Kekseliästä ongelmanratkaisua, luovuutta ja ryhmätyöskentelytaitoja. Minulle henkilökohtaisesti tämä oli ensimmäinen ryhmässä tehty ohjelmointityö, tästä opin mm. Selkeän koodaustyylin ja muuttujien fiksun nimeämisen tärkeydestä.

Palaute harjoitustyöstä (vapaaehtoinen)

Harjoitustyö oli kokonaisuudessaan melko haastava. Mielestämme sen toteuttaminen kokonaisuudessaan Javalla olisi järkevämpää, kuin opetella AndroidStudioon kaltaista uutta ohjelmistoa kesken kurssin. AndroidStudio ei opeta suoranaisesti opeta ohjelmoinnin konsepteja, vaan keskittyy lähinnä framework taisteluun. Jos kurssi toteutettaisiin

pelkästään Javalla, voitaisiin osaamista syventää myös Java-ohjelmoinnin konsepteihin (esimerkiksi Networking ohjelmointi, databaset, multithreading, sekä Collections.

Helppoja ominaisuuksia työssä olivat kielten muuntaminen, salasanojen sääntöjen luominen, sekä rajapinnan implementointi työhön. Haastavaa työssä oli adapterien luominen tietojen saamiseksi listaan, sekä datan luominen tietojen tallentamiseksi. Aivan syvältä työssä oli kokonaisuudessaan AndroidStudio käyttäminen ja intialaisten opetusvideoiden perusteella ominaisuuksien luominen. Työssä riemua aiheutti sen loppuminen. Jälkikäteen pientä auringonpaistetta myrskypilvien sekaan tuo myös uuden alustan (AndroidStudio) käytön oppiminen.

Ensi vuoden harjoitustyön osalta ryhmältämme ei tule kovinkaan suuria lisätoiveita. Työ oli mielenkiintoinen kokonaisuus ja valinnanvapautta oli välillä jopa liikaakin. Koska kyseessä on seuraavan vuosikurssin opiskelijat, voisi meidän ryhmämme mielestä työtä mielellään vaikeuttaa. Tähän kuitenkin lisäämme ehdon:

```
if(😞.getGrade() > 0) {  
    while( junnutkärsii() ){  
        harkkatyoGetsHarder();  
    }  
} else{  
    Erno.removeHarkkatyo();  
}
```

Ohjeistuksessa voitaisiin keskittyä myös esimerkiksi Githubin hyödyntämiseen, sekä esitellä järjeviä toimintamalleja ryhmäprojektin tekemiseen ohjelmoinnissa. Siinä koimme alkuvaiheessa haasteita, koska Githubin ryhmässä käyttäminen ei meille kovinkaan tuttua ollut. Ohjelmoinnin porukassa tekeminen on myös melko erilaista, joten kurssin aikana voitaisiin harjoitella yhdessä pienten projektien tekoa.