



## Alko-Apu

Otto Manninen (000425931), Niko Purhonen (0494525), Oskari Kalliomäki (0494266)

# Kuvaus ohjelmasta

Applikaation tarkoituksena on auttaa käyttäjää löytämään uusia makuelämyksiä Suomen valtion alkoholimonopolin valikoimasta.

Applikaatio käy hakemassa Alkon palvelimelta excel-tiedoston, jossa on listattu kaikki Alkon tuotteet tietoineen. Tämän jälkeen käyttäjä voi määrittää haluamansa dokabiliteettipitoisuuden, jonka perusteella applikaatio suosittelee makuelämystä käyttäjälle. Käyttäjä voi tämän jälkeen palata tekemään uuden haun erilaisella dokabiliteetilla mikäli tarjotun pullon etiketti ei miellyttänyt tai tutkia tarjottua juomaa enemmän. Appi tarjoaa juomasta tarkempia tietoja tai mahdollisuuden etsiä lähimmän myymälän, josta herkkua tarjoillaan. Käyttäjä voi myös antaa tarjotulle juomalle arvostelun sekä tallentaa tämän suosikiksi.

Vuosia kuluu ja käyttäjä yhä muistaa toissavappuna nautitun suositusjuoman lempeän maun kielensä päällä, muttei millään saa mieleensä kyseisen herkun nimeä... Ei hätää, käyttäjä oli onneksi tallentanut kyseisen juoman applikaatioon ja antanut sille viisi tähteä. Applikaation suosikkilistan aukeamisen jälkeen helpotus on silmissä nähtävää, kun käyttäjä riemastuu löydöksestään ja kirmaa taas kevyin askelin kohti viinakauppaa.

Päästyään sisään Alkoon käyttäjä muistaa myös viime vappuna juodun juoman, jonka katkeran maun muisteleminenkin saa suun happamaksi. Käyttäjä päättää varmistaa, ettei vahingossakaan osta kyseistä juomaa ja tarkistaa applikaatiosta arvostelun, jossa hän on antanut kyseiselle juomalle 0 tähteä ja haukkunut sitä kuravelliksi. Käyttäjä avaa mielenkiinnosta juoman alta Alkon sivut ja katsoo mistä ainesosista tuo myrky oikein koostuikaan.

## Tekijät

Nimi	Rooli
Oskari Kalliomäki	Full Stack, Technical Lead
Otto Manninen	Frontend, Project Manager
Niko Purhonen	Backend, CEO, entrepreneur, born in 1997

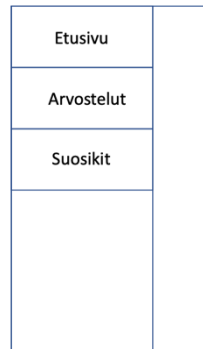
Alustava suunnitelma toteutettiin yhdessä projektin alussa, johon kuului alkuraportti sekä ohjelman luonnostelu toimintoineen. Alkupalaverin yhteydessä tehtiin karkea aikataulusuunnitelma, luokkakaavio sekä tehtävien jako.

Frontend kehittäjän rooliin kuului ohjelman ulkoasun ja käyttöliittymän tekeminen (layoutit, fragmentit, siirtymät sekä visuaalit). Backend kehittäjän vastuulle jäi analytiikka ohjelman tekeminen. Full Stack kehitti Alkon palvelimelta ladattavan excelin parsijan sekä muita Backend toimintoja ja viimeisteli ohjelman kokoonpanon. Jokainen rooli tuki toinen toistaan aina tarvittaessa sekä osa toiminnoista tehtiin yhteistyössä. Ohjelma toteutettiin alustavan suunnitelman perusteella ja suuria poikkeamia alkuperäiseen rakenteeseen ei tullut projektin aikana.

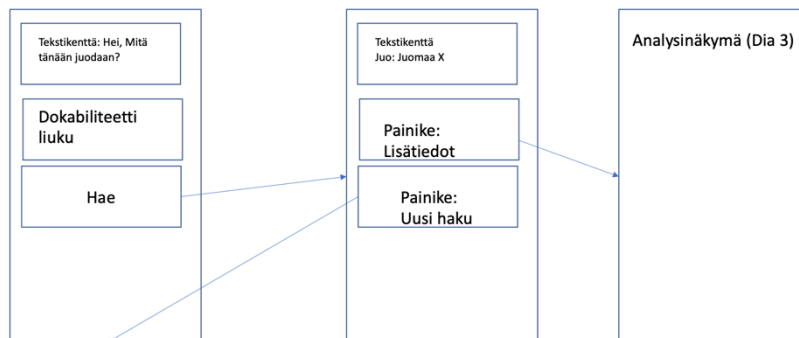
# Ohjelman toteutus

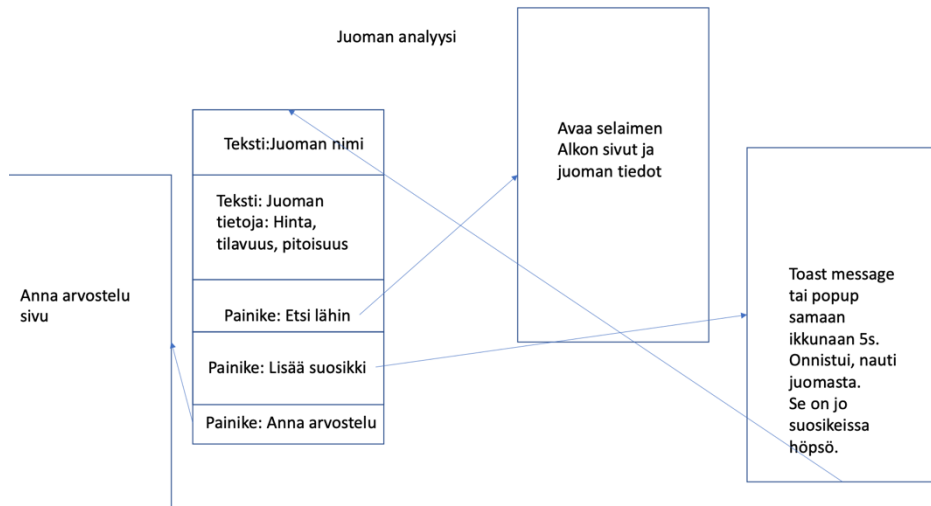
Ohjelma suunniteltiin MS Teams palaverissa ja alustavat käyttäjäpolut ja sovelluksen toiminnot mallinnettiin MS Power Pointissa sekä luokkakaavion avulla. Alla MS Powerpointin lähtösuunnitelma:

Sivupalkki



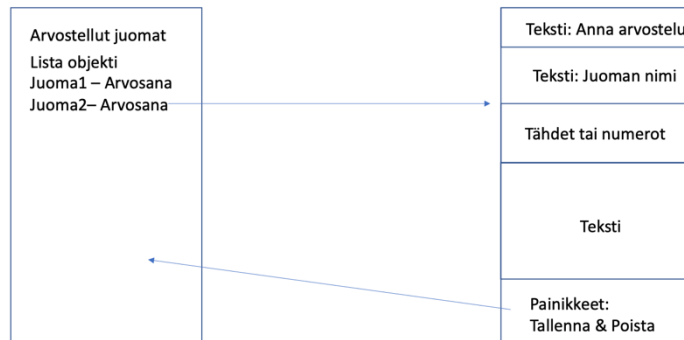
Dashboard





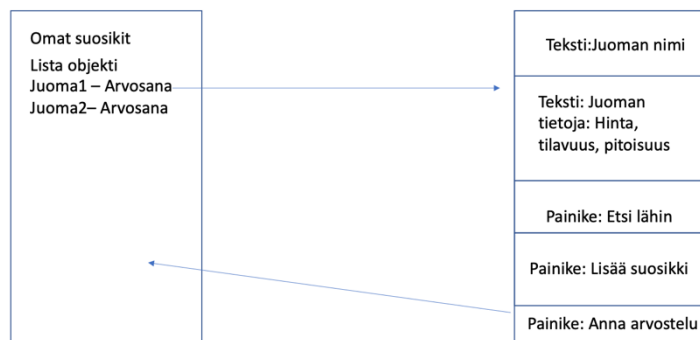
#### Arvostelut listanäkymä & Anna arvostelu

- Juomissa onclick listener. Painamalla avaa Anna arvostelu kyseiseltä juomalta.
- Tallenna ja poista palaa listanäkymään



#### Omat suosikit listanäkymä & Anna arvostelu

- Juomissa onclick listener. Painamalla avaa Juoma analyysi sivun kyseiseltä juomalta.



Ohjelma toteutettiin Android Studiolla ja versionhallintaan käytettiin GitLabia. Android Studion ohjelmointikielenä käytettiin Javaa. Ryhmätyöskentelyä varten hyödynnettiin Microsoft Teams palvelua sekä GitLabin lisäksi Sharepointtia tiedostojen jakamiseksi.

Testaamisessa käytettiin Android Studiota ja testitapaukset dokumentoitiin Microsoft Wordiin. Oheisessa taulukossa tehdyt testitapaukset

Testitapaus	Testattu
Käyttäjä lataa alkon sivulta juomien tiedot	Ok
Käyttäjä määrittää dokabiliteetin ja hakee sopivan juoman	Ok
Käyttäjä tallentaa juoman suosikiksi	Ok
Käyttäjä arvostelee juoman	Ok
Käyttäjä katsoo arvosteluita	Ok
Käyttäjä katsoo suosikeita	Ok
Käyttäjä tekee uuden haun ensimmäisen haun jälkeen	Ok
Käyttäjä katsoo haetusta juomasta lisätietoja	Ok
Käyttäjä siirtyy alkon sivuille AA-apista	Ok

Kaikki ryhmän jäsenet osallistuivat testaukseen. Testaaminen koostui käyttötapausten testauksista sekä satunnaisista savu- ja regressiotesteistä projektin aikana. Kehittäjien vastuulle jäi omien toimintojensa testaaminen kehityksen yhteydessä ennen versionhallintaan päivittämistä. Ohjelmoinnissa käytettiin Javan peruskirjastoja sekä lisäksi excelin lukemiseen apache.poi:ta.

## Luokkakaavio



Luokkakaaviossa on esitetty ainoastaan datan säilömiseen ja prosessointiin käytettävät luokat. Käyttöliittymäluokkia ei ole esitetty.

- **AlkoCrawler**
  - Periytetty luokasta ArrayList, sisältää AlkoProduct -olioita
  - Hakee tuotteet Alkon palvelimelta metodilla loadProducts()
  - Generoi URL-osoitteet tuotesivulle ja kaupan (AlkoStore) sivulle
  - Sisältää HashMap-tyyppisen listan AlkoStore -olioista, joita voi kysyägetStoreById() -metodilla
- **AlkoStore**
  - Alkon myymälää esittävä luokka
  - Sisältää tarvittavat tiedot myymälästä
  - Alkuperäisessä suunnitelmassa appi osaa etsiä lähimmän myymälän, mistä tuotetta löytää. Aikataulun takia ominaisuus jäi toteuttamatta, joten AlkoStore -luokka on toistaiseksi vailla käyttöä.
- **AlkoProduct**
  - Alkon tuotetta esittävä luokka
  - Sisältää kaikki applikaation kannalta relevantit tiedot tuotteesta
- **Rating**
  - Arvostelu
  - Sisältää viitteen AlkoProduct -olioon, sanallisen arvostelun ja numeroarvosanan väliltä 1 – 5
  - Osaa kirjoittaa itsensä PrintWriter -tyyppiseen tietovirtaan, ja lukea itsensä BufferedReader -tyyppisestä tietovirrasta.
- **FavouriteList**
  - Periytetty ArrayList -luokasta, sisältää AlkoProduct -olioita
  - Lista edustaa käyttäjän suosikiksi lisäämiä tuotteita
  - Sisältää metodit listan kirjoittamiseen ja lukemiseen tiedostosta.
    - Ainoastaan tuote-id kirjoitetaan tiedostoon. Luettaessa viittaukset AlkoProduct -olioihin luodaan uudelleen.
- **RatingList**
  - Periytetty myös ArrayList -luokasta, sisältää Rating -olioita
  - Hyvin samankaltainen kuin FavouriteList -olioon verrattuna

## Toteutetut ominaisuudet

Ominaisuus	Perustelut	Pisteet
Ulkoinen datalähde	Tehtävänanto	3
Asynkroninen HTTP kutsu data hakiessa	Tehtävänanto	2

Scoped storage käyttö tallennuksessa	Tehtävänanto	2
Excelin prosessointi	Ohjelma tarvitsee dataa, jotta voi palvella käyttäjiänsä	3
Tietojen analysointi	Käyttäjä haluaa löytää tietyn juoman dokabiliteetin perusteella (dokabiliteetti määrittyy hinnan ja alkoholipitoisuuden perusteella)	3
Tietyn sivun avaaminen mobiiliapista	Käyttäjä haluaa nähdä vielä lisää tietoja juomista	2
Suosikin tallennus	Käyttäjän tulee pystyä tallentamaan suosikki juomia	2
Arvostelun tallennus	Käyttäjän tulee tietää, kuinka paljon hän on tykännyt mistäkin juomista	2
Suosikkien katsominen	Käyttäjän tulee pystyä listaamaan suosikkijuomansa	2
Suosikit listanäkymän toiminnot	Painalluksella linkitys toiseen näkymään ja pohjaan pitäessä suosikin poisto. Suosikkien filterointi tähtien perusteella	3
Arvostelujen katsominen	Käyttäjän tulee pystyä katsomaan koostetusti arvosteluitaan	2
Fragmenttien käyttö aktiviteettien sijaan	Jotta ohjelmaa voi käyttää kivemmin.	2
Responsiivisuus	Ohjelmaa voi käyttää eri kokoisilla näytöillä	2
Käytettävyys/ Käyttäjäpolku apissa	Käyttäjän tulee pystyä liikkumaan sujuvasti eri toimintojen välillä ja ohjelman tulee olla riittävän intuitiivinen, jotta käyttäjä ei tarvitse erillisiä käyttöohjeita	3
Sivupalkki	Parempi käyttökokemus	2
Toast messaget	Parempi käyttökokemus	1+
Aivan eppinen ulkoasu	Erittäin miellyttävä käyttökokemus	4
Arvostelu: Annettujen tähtien mukaan vaihtuvat tekstit	Hiukan huumoria	2
Viimesen päälle tehty dokumentaatio ja luokkakaavio	Ylläpidettävyys ja jatkokehitys	3
Summa		45+

# Työmäärät

Tekijä	Tehtävät	Tunnit
Otto Manninen	Devaus, testaus, palaverit ja dokumenttien teko	66
Oskari Kalliomäki	Devaus, testaus, palaverit ja dokumenttien teko	71
Niko Purhonen	Devaus, testaus, palaverit ja dokumenttien teko	68
Summa		205

## Mitä opin harjoitustyöstä?

Otto Manninen: Android applikaation ohjelmointia Android Studiolla, käyttöliittymän ulkoasun tekoa, fragmenttien sekä aktiviteettien käyttöä sekä niiden välillä siirtymistä. Yleisesti Java koodaustaidot kehittyivät. Ulkoasussa erilaisten kuvien ja vektoreiden hyödyntäminen sekä visuaalien välillä siirtyminen. Lisäksi arvostelu tähtien tekstit olivat hauska lisäharjoitus.

Oskari Kalliomäki: Opin käyttämään GitLabin versionhallintaa Android Studion kanssa. Lisäksi sain kattavan kokonaiskuvan mobiiliohjelmoinnista, niin front- ja backendistä sekä fragmenttien käyttökokemusta. Olioiden yhdistelyssä pääsin oppimaan ongelmanratkaisua. Tiedon lataaminen ohjelmaan ja sen sovittaminen käyttäjäystävällisesti oli hiukan haastavaa ja tätä asiaa ei ollut huomioitu määritelmässä tarkemmin, joten pääsin tässäkin kehittämään loistavaa ongelmanratkaisukykyäni.

Niko Purhonen: Opin Android mobiilikoodausta sekä Javassa pääsin syventämään taitojani uusien tekniikkojen parissa, joista yhtenä esimerkkinä analysointi olio. Työkaluista Git Lab oli uusi ja sen kanssa oli aluksi hiukan haasteita, mutta niistä selvittiin. Ohjelmoinnissa tuli jatkuvasti erinäisiä haasteita, jotka ratkaisin tavalla tai toisella ja tiedonhaun tekniikat kehittyivät projektin aikana huomattavasti. Opin myös, että toisinaan androidin sijainnin lupien kysely on melkoisen ikävää hommaa.

Yhteisiä haasteita:

Kaikki ryhmäläiset kokivat haasteita GitLabin ja Android Studion kanssa. Fragmenttien ja aktiviteettien kanssa oli hiukan haasteita, sillä määritelmässä ei ollut otettu kantaa mitä tekniikkaa mikäkin sivusto käyttää. Lopulta päädyttiin käyttämään enemmän fragmentteja, sillä toiminnot oli suurimmaksi osaksi tehty niiden mukaan. Versionhallinnan haasteiden vuoksi ensimmäisen yhdistämiskerran kohdalla huomattiin, että joitain toimintoja oli tehty useampaan kertaan eri tavoilla ja koodin yhdistämisessä meni paljon aikaa. Työssä tulisi olla vähintään yksi työnjohtaja, joka ei



mitään muuta kuin pakota muita tekemään hommia ja katsoo, ettei plebeijit tee samaa asiaa samaan aikaan.

## Palaute harjoitustyöstä (vapaaehtoinen)

Tää oli ihan jees!

Pienellä lisäajalla applikaatio saattaisi löytyä Google Play Storestakin. Ehkä harjoitustyön voisi ensi vuonna aloittaa hiukan aikaisemmin. Ehkä iOS ohjelmointia myös. Ehkä joku ohjeistus, kuinka GitLab toimii Android Studion kanssa? Tosin vika on aina käyttäjässä.

Harjoitustyön palautusaikaa voisi myös mielellään siirtää vaikkapa kuukaudella. Olio-ohjelmointi konseptina oli helppo sisäistää, mutta sen sijaan Android Studion opetteluun meni melko paljon aikaa. Sinänsä Android-ohjelmoinnin opettelu oli mukavaa ja hyödyllistä, mutta aikataulu oli mielestämme aivan liian tiukka. Kaikki deadlinet lähellä vappua on melkoisen ikäviä ja aiheuttaa suurta määrää ikävää mieltä, kun joutuu lopettamaan juhlimisen jo klo 3 aamuyöstä, jotta ehtii pohtia mistä johtuu NullPointerException.

Tässähän jäi vielä rutkasti aikaa ylimääräiseksi.

## Lähdekoodin versionhallinta

Harjoitustyön lähdekoodi on luettavissa GitLabissa:

<https://version.lab.fi/olioohjelma-ty/alkoapu>

Harjoitustyön sovelluksen pikainen ja huono esittely

[Linkki esittelyvideoon](#)