

AjoLoki

Yleistä

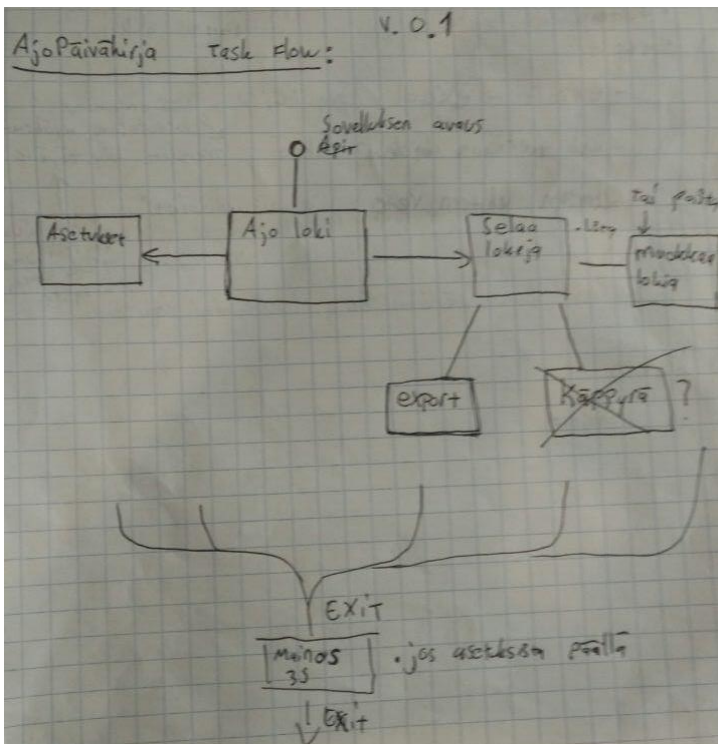
Sovelluksen on ajopäiväkirja. Sovelluksen tarkoitus on olla yrittäjän apuna työ ja oman ajon seurannassa. Alun perin ajattelin suunnata sovelluksen laajemmalle kohderyhmälle, mutta lopulta päätin keskittyä vain suomalaisiin yksityisyrittäjiin. Lähes jokainen projektin kompromissi ja mutkan oikominen johtui omasta aikataulutuksen sössimisestä.

Toiminnallisuus

Tässä versiossa toiminnot on jätetty minimiin. Tulevaisuudessa ajoin toteuttaa GPS pohjaisen matkan pituuden tallentamisen ja matkojen muokkauksen. Tässä versiossa voi tallentaa matkan pituuden matkamittarin lukemien perusteella, ajon päivämäärän ja ajan sekä tyyppin (oma/työ). Tietoja voi myös selailla ja poistaa yksittäisiä rivejä sekä tallentaa koko tietokannan CSV-tiedostoon.

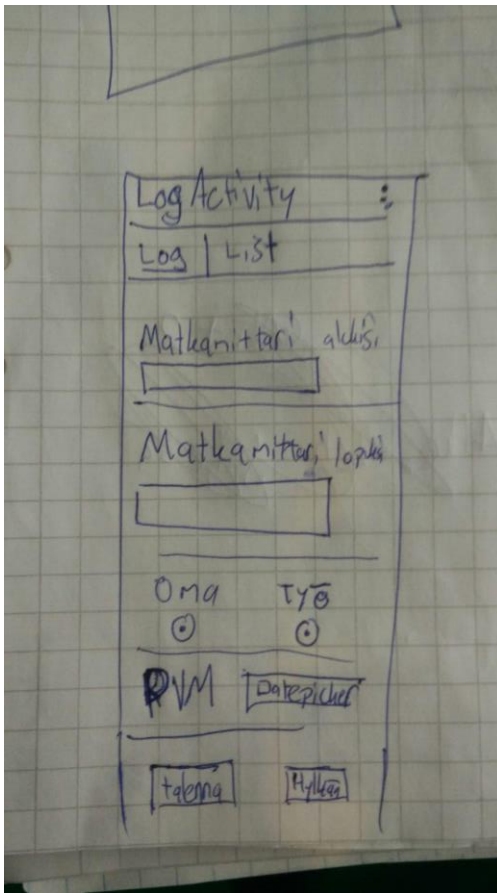
Suunnittelu

Aloitin suunnittelun listaamalla paperille haluttuja toimintoja. Listan jälkeen hahmottelin muutaman version task flow diagrammeja (Kuva 1). Seuraavaksi piirsin wire frame kuvia kaksi versiota. Käsin piirtelyn jälkeen tein FuiidUI:lla toimivan UI demon (https://www.fluidui.com/editor/live/preview/p_HEA6iG4UpdW2LmAJ5VMKSUKTgahRxQ3F_1461940344134).



Kuva 1: Task Flow ensimmäinen versio

FluidUI:n jälkeen piirsinkin vielä yhden wire frame version käsin (Kuva 2). Näiden wire framen perusteella



Kuva 2: low res wireframe

toteutin lopullisen ulkoasun.

Datan säilömistä ei tarvinnut juuri miettiä. Valitsin SQLite – tietokannan koska se tuntuu olevan standardi tapa säilöä tietoa Androidilla.

Käyttäjäryhmäkseni valitsin suomalaiset yksityisyrittäjät. Mietin vähän user story pohjalta potentiaalisia sovelluksen käyttäjiä. Rami on omalla autolla liikkuva, hajamielinen yksityisyrittäjä. Hän pitää ajopäiväkirjaa kynäpaperilla, mutta unohtaa välillä vihon kotiin ja hukkaa tilapäiset laput. Hän on kiireinen, päiväkirjan pitoon tuntuu menevän liikaa aikaa. Niinpä sovelluksesta on tärkeä saada mahdollisimman helppo ja nopea.

Pyrin mahdollisimman nopeaan ja sulavaan käyttöliittymään, joten päätin virtaviivaistaa suunnitelmia. Alkuperäisissä suunnitelmissa käyttäjän olisi pitänyt mennä aloitusruudun kautta kirjaamaan ajot. Nyt sovellus aukeaa suoraan ajon kirjaus näkymään (Kuva 3).

Toteutus

Sovelluksen värit on valittu materialpalette.com sivuston työkalulla ja ovat material design ohjeiden mukaiset (<https://www.google.com/design/spec/style/color.html#color-color-palette>). Sovellus noudattaa muutenkin material design ohjeita mahdollisimman tarkkaan kokonaisuudessaan - fonttien koosta marginaaleihin.

Tietokantatoteutus on tehty DBFlow (<https://github.com/Raizlabs/DBFlow/>) kirjaston avulla. Tietokanta luetaan FlowCursorList:iin, joka on Cursor pienillä lisäominaisuuksilla. Tiedot näytetään käyttäjälle ArrayAdapterin

avulla listana(Kuva 4). Listaa päivitetään aina kun siihen tulee muutoksia. Lisäksi listassa on ”pull to refresh” –toiminto.

Välilehtien vaihto ja välilehti-palkki on toteutettu google:n I/O Schedule

seminaarin(<https://github.com/google/iosched/tree/master/android/src/main/java/com/google/samples/apps/iosched/ui/widget>) koodien avulla. Itse välilehdet ovat tavallisia Fragmentteja (Kuva 3),(Kuva 4).

Kuva 3: ajojen kirjaus näkymä

Tyyppi	Kuvaus	Matka
	Päivämäärä	
Työ	KHL to d Olof Monday, 23 May 2016 05:39	10 km
Työ	Kuvaus ilta Saturday, 21 May 2016 23:05	8 km
Oma	kävin kaupassa Friday, 20 May 2016 13:00	14 km
Oma	hyvää tt Friday, 20 May 2016 13:00	1 km
Oma	joo Friday, 20 May 2016 07:00	1 km
Oma	testiää joo joo Monday, 2 May 2016 07:00	20 km
Oma	Kujuu juuvaus Sunday, 1 May 2016 13:00	4 km
Oma	uu Saturday, 30 April 2016 19:00	10 km
Työ	Kuvausten Saturday, 30 April 2016 13:22	2 km
Työ	joka Saturday, 30 April 2016 12:22	1 km
Oma	gg Friday, 29 April 2016 12:15	76 km
Työ	kävin jaksoon ns	1 km

Kuva 4: ajojen selaus näkymä

Ajonkirjaus näkymän(LogFragment) ei ole juurikaan mitään erikoista. Syötteet luetaan kuten tavallista. Ajan ja päivämäärän lokalisatio ongelmien estämiseksi aikaa näytetään paikallisena käyttäjälle, mutta se tallennetaan ja käsitellään GMT muodossa.

Käytettävyys

Pyrin arvioimaan sovellusta kokoajan Jakob Nilsenin heursitisten periaatteiden pohjalta

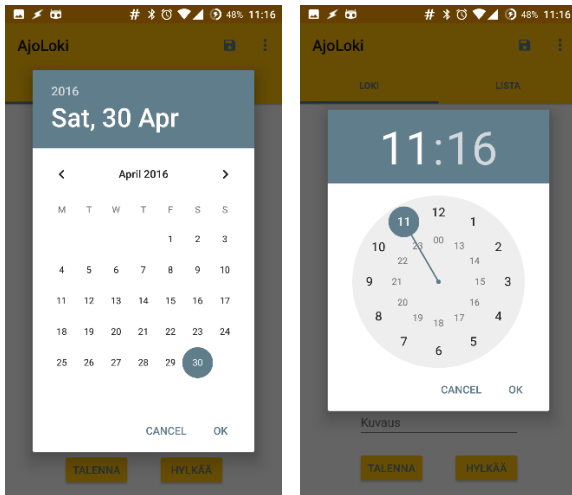
(<http://www.uxness.in/2015/02/10-heuristic-principles-jakob-nielsens.html>).

Visibility of system status

Käyttäjä tietää kokoajan missä hän on sovelluksessa välilehti palkin ansiosta. Toimintojen onnistumisesta ja epäonnistumisesta ilmoitetaan Toastilla.

Match between system and the real world

Pyrin käyttämään mahdollisimman luonnollista kieltä kaikkialla. Lisäksi CSV-tallennus nappi on perinteinen disketti-ikoni. Päivämäärän(Kuva 6) ja ajan(Kuva 5) valitseminen on niin lähellä oikeaa maailmaa kun vain mahdollista.



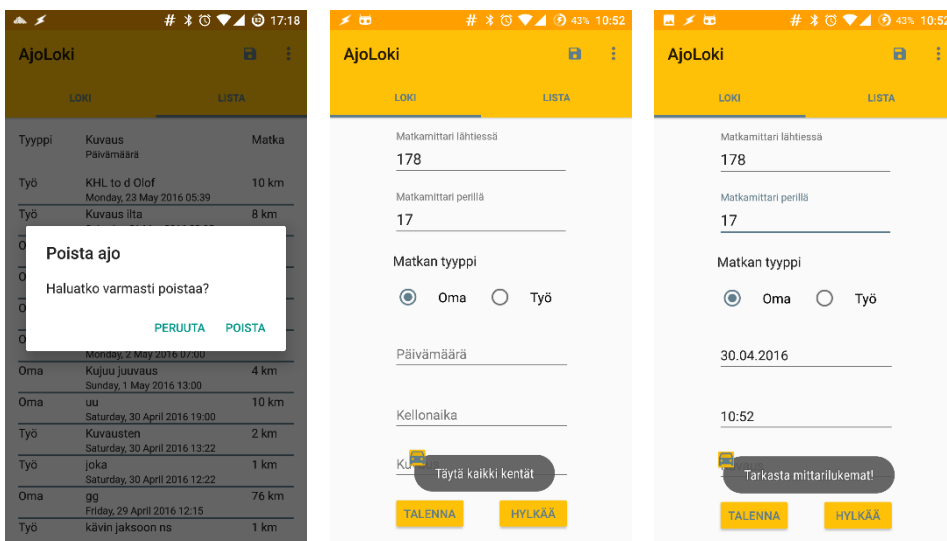
Kuva 6: datepicker fragmnet Kuva 5: timepicker fragmnet

Consistency and standards

Sovellus seuraa material desing ohjeita ja kaikki toiminnot ovat hyvin saman kaltaisia kuin muissakin Android sovelluksissa.

Error prevention

Käyttäjä ei voi tallentaa tyhjää, puoliksi täytettyä(Kuva 8) tai järjetöntä ajoa(Kuva 9). Järjetön ajossa matkamittarin lukema on perillä pienempi kuin lähtiessä. Käyttäjältä kysytään varmistus ennen ajon poistamista(Kuva 7).



Kuva 7: poiston varmistus

Kuva 8: tyhjät kentät

Kuva 9: järjetön ajo

Flexibility and efficiency of use

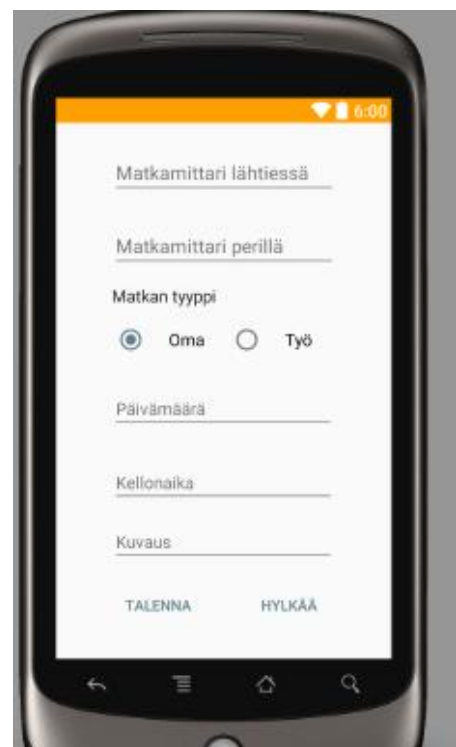
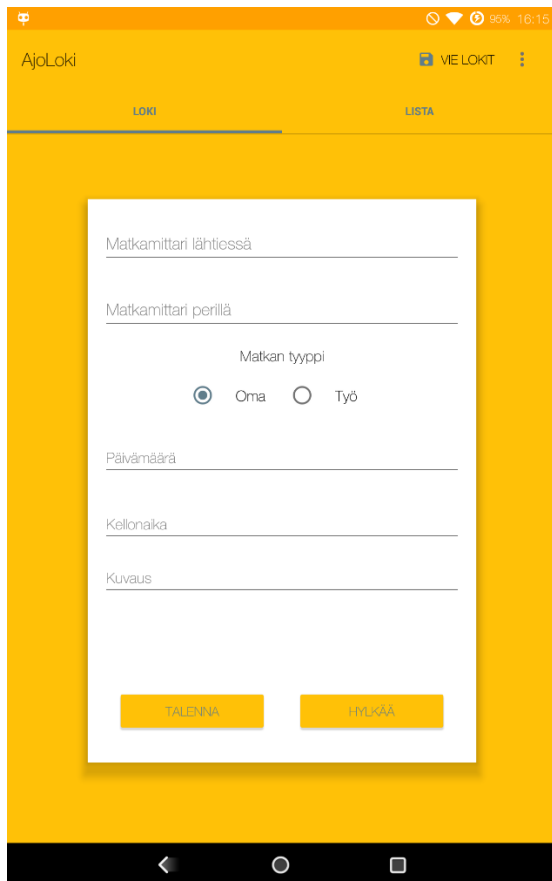
Matkamittari kentät esitäytetään käyttäjälle valmiiksi edellisten ajojen mukaan. Matkamittarin lähtölukema täytetään kokonaan ja perillä lukema osittain, siten että käyttäjän tarvitsee syöttää yleensä vain 1-3

numeroa. Ajonkirjaus näkymässä puhelin pysyy aina pysty asennossa(portrait mode),vaikka puhelinta kuinka kääntelisi. Tämä helpottaa sovelluksen yhdellä kädellä käyttöä ja estää vahinko kallistukset autossa.

Aesthetic and minimalist design

Material Desing. Kaikki on yksinkertaista ja näittiä, mikään ei ole ylimääräistä.

UI skaalautuu pieneltä Nexus Onen näytöltä(Kuva 10) tabletille(Kuva 11) ja kaikelle siltä väliltä. Nexus Onen nappien värit on väärin Android Studion takia.



Kuva 10: Nexus One

Kuva 11: tablet UI