

# Feedback Klusjesapp

## SWOT- analyse

### Strengths

- Grote potentiële userbase, mensen hebben altijd klusjes om te doen, en iedereen is wel op zoek naar een manier om een centje bij te verdienen.
- Uniek idee, er zijn weinig apps die dit concept gebruiken.
- Kaart: voor veel studenten (die meestal geen eigen auto en rijbewijs hebben) is afstand van groot belang, ze willen klusjes vinden die niet te ver weg zijn.
- Klusjes waarbij je normaal lang moet wachten tot ze gedaan worden en waar je vaak veel moet betalen aan bedrijven kunnen gedaan worden door studenten die meer tijd hebben en sneller kunnen helpen, minder betaald moeten worden en die er zelf ook eruit leren.

### Weakneses

- Een probleem is dat de mensen die de klusjes zullen doen waarschijnlijk geen bewijs hebben dat ze die klus correct kunnen uitvoeren, (bv. bij het op een hond passen, als je deze bij een hondenhotel zou achterlaten ben je zeker dat er goed voor gezorgd wordt). Indien de users kunnen reviews plaatsen bij iemand die een klusjes heeft gedaan bij hen dan kunnen users meer vertrouwen krijgen door een aantal goede reviews, en bij een aantal slechte reviews kunnen mensen wegblijven van iemand die hun werk niet goed doet.
- Wanneer er geen permissies worden gegeven tot locatie, crasht de app wanneer naar de kaart wordt genavigeerd, maar er wordt nooit gevraagd aan de gebruiker om toegang te geven tot locatie. De kaart die wordt weergegeven wanneer locatiepermissies aan staan is leeg. In de Pixel 6 API wordt helemaal geen kaart gerenderd, enkel het zoekbalkje staat er.

### Opportunities

- Reviews kunnen plaatsen bij users die klusjes doen. Op de homepage staat er wel een aanbeveling van “Steven Vos Tuinen” met 4.5 sterren, maar het is niet duidelijk reviews door iedereen geplaatst kunnen worden, of enkel voor mensen nadat deze hun klus geklaard hebben voor jou.
- Een verschillende UI voor users die klusjes doen en users die klusjes plaatsen zou handig zijn. Er zal waarschijnlijk een minimum aan users zijn die beiden klusjes plaatsen en zelf klusjes doen, maar de app geeft momenteel weinig verschil voor deze 2 soorten gebruikers. Bijvoorbeeld op de home page staan er categorieën (op moment van feedback zonder functionaliteit).

Zijn deze bedoeld om een klusje te plaatsen of te zoeken? De gebruikers die zelf klusjes plaatsen zullen niet veel nut hebben van de kaart functie.

- Socio-economisch aspect: in België hebben we een trend van vergrijzing waarbij er veel oudere mensen zijn die niet kunnen werken en hulp nodig hebben. Een app die zowel meer studenten aan het werk zet als de oudere bevolking kan helpen met klusjes die ze zelf niet meer kunnen doen kan hier op anticiperen.

### Threats

- De app zal 2 soorten gebruikers hebben: degenen die klusjes posten en degenen die klusjes doen. Voor een optimale werking zou je ongeveer evenveel van beide soorten gebruikers nodig hebben. Anders heb je gebruikers die geen klusjes opgelost krijgen of users die geen werk vinden.
- De meeste klusjes kunnen opgelost worden door bedrijven (bv. tuiniersbedrijven, hondenhotels, ...). Deze app zou een alternatief bieden tegenover de diensten van die bedrijven. Wanneer de app veel succes heeft, zal deze concurrent worden voor deze bedrijven. Je kan dus in veel verschillende markten veel mogelijke concurrenten hebben.
- Wat gebeurt er wanneer iemand zijn klusje niet goed uitvoert of schade aanbrengt aan eigendom van zijn opdrachtgever? Wat gebeurt er wanneer een opdrachtgever de persoon die zijn klusje gedaan heeft niet betaalt?
- Wordt het verrichtte werk aangegeven bij de belastingen? Is dit in het zwart? Wat met het maximum van 450 uren dat een student mag werken in België? Misschien helpt een disclaimer of een Terms of Service hierbij.

### Code Review

Positieve puntjes:

- Navigatie is volgens jetpack compose opgebouwd, met behulp van routes en een **AppState**
- Duidelijke opdeling van bestanden
- Communicatie met databank via services
- Gebruik van viewmodels

Negatieve puntjes:

- Hardcoded strings (bv. bij HomeScreen.kt 51-58: Alle categorieën zijn hardcoded)
- Errors juist afhandelen (bv. HomeViewModel.kt 21: Er wordt gewoon iets geprint naar stdout)

- Er is een package met common composables maar er deze worden niet gebruikt.
- Op veel plekken staat `@ExperimentalMaterialAPI`. Dit is volgens ons niet de beste oplossing voor waarschijnlijk een vaag probleem. We denken dat het beter is om dit bij zijn oorsprong op te lossen, zodat deze annotation niet door heel het project doorloopt.