



**ŽILINSKÁ UNIVERZITA V ŽILINE**

Fakulta riadenia  
a informatiky

Semestrálna práca z predmetu  
*vývoj aplikácií pre mobilné zariadenia*

**POROVNAJ.TO**

Vypracoval: Marek Štefanča

Študijná skupina: 5ZIF11

Akademický rok: 2024/2025

V Žiline dňa 7.6.2025



## Obsah

Úvod .....	2
Prehľad podobných aplikácií .....	3
Heuréka .....	3
Kimbino .....	3
Analýza navrhovanej aplikácie .....	4
Návrh architektúry aplikácie .....	5
DB entity .....	5
Dao rozhrania .....	6
ViewModel triedy .....	6
FormItemViewModel .....	6
TopBarViewModel .....	6
Navigation .....	7
Obrazovky a komponenty .....	7
Finálna verzia obrazoviek a mapovanie .....	8
Hlavná obrazovka .....	8
Zoznam Kategórií a položiek .....	9
Informácie o položke .....	9
Vytvorenie novej položky .....	10
Zoznam zdrojov .....	11



## Úvod

V rámci predmetu Vývoj aplikácií pre mobilné zariadenia som sa rozhodol vytvoriť aplikáciu ktorá slúži na porovnanie cien vybraných položiek v kamenných obchodoch.

Pre výber tejto aplikácie som sa rozhodol hlavne z dôvodu že takáto aplikácia v obchode Google Play chýbala.

Častokrát som hľadal nejaký vyhľadávač cien v kamenných obchodoch, avšak nepodarilo sa mi nájsť žiadnu alternatívu ktorá by toto uspokojila, nakoľko v obdobných aplikáciách je len akciový tovar nie tovar ktorý sa dlhodobo predáva...

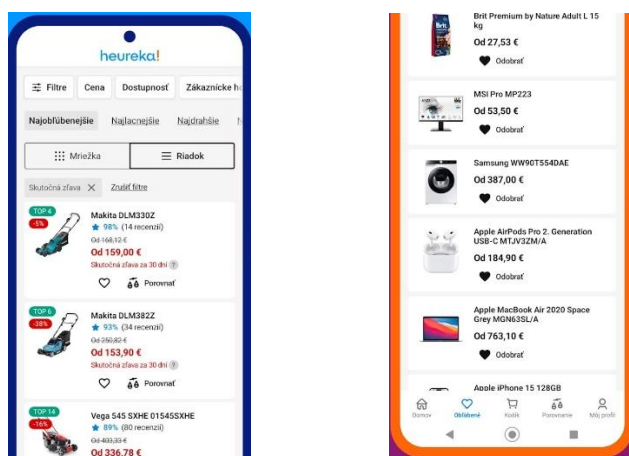
Vzhľadom na to, že si hypermarkety svoje štandardné ceny prísne strážia som sa rozhodol využiť existujúce jazykové modely k tomu aby mi vygenerovali dáta z ktorých sa skladá aplikácia (aby nejaké boli). Tieto dáta boli následne importované do SQLite databázy, kvôli efektívnejšiemu vyhľadávaniu a normalizácii.

## Prehľad podobných aplikácií

Vzhľadom k tomu, že aplikácia bola v čase realizácie unikátna, tak neexistovali žiadne riešenia ktoré by boli s obsahom aplikácie totožné, t.j. sú len podobné aplikácie, či weby.

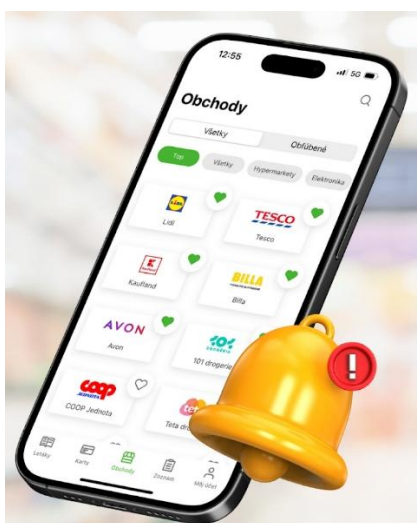
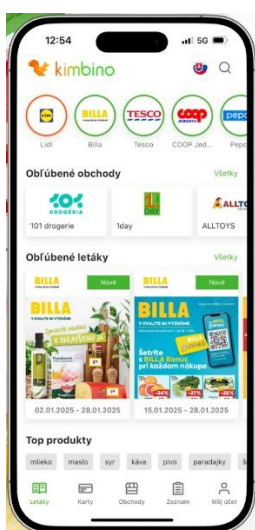
### Heureka

Ako jeden z príkladov môžeme uviesť heurčku, ktorá je online katalógom pre položky a vieme v nej porovnať cenu, bohužiaľ heureka disponuje iba položkami ktoré sa predávajú v internetových obchodoch, nie v kamenných prevádzkach. Teda základné potraviny by sme tam našli len ťažko



### Kimbino

Ďalšou aplikáciou ktorá ma pri výbere oslovila je aplikácia Kimbino, táto aplikácia splňala skoro všetko čo by som od nej požadoval až na to, že sa nedalo vyhľadávať položky podľa kategórii alebo podľa názvu, dalo sa iba po obchodoch, tiež chýbalo porovnanie ceny u „konkurencie“. Ináč ma zaujala aplikácia designom ktorý som zjednodušil pre možnosti intuitívnejšieho ovládanie ľuďmi





Ako bolo už vyššie spomenuté aplikácie sú podobné ale úplne mňou nespĺňajú účel, resp. síce sú možno prepracovanejšie ale chýba im to čo by som ako potenciálny zákazník hľadal. Preto som si teda z nich prevzal to čo by som očakával, zjednodušil som to na použiteľnú formu a samozrejme implementoval to tak aby systém fungoval off-line, nakoľko ak je človek na mieste kde nie je signál/ iný prístup k internetu, tak nemá možnosť si pozrieť v podstate nič...

## Analýza navrhovanej aplikácie

Aplikácia bude nástroj ktorý umožní nám vizualizovať dostupné dáta, ktoré sú implementované v zariadení (sú importované v sqllite databázy).

V rámci aplikácie je možnosť zobrazíť si dáta podľa rôznych možností.

Jednou zo základných možností je využitie filtrovania prostredníctvom základných kategórii ako sú napríklad:

- Mliečne výrobky
- Drogéria
- Nápoje

A mnoho ďalších kategórii..

Ďalším spôsobom ako dáta filtrovať je na základe obchodov. V aktuálnej verzii DBS sú nahraté tieto obchody

- Tesco
- Billa
- Kaufland
- Jednota
- Terno

K dispozícii je databázový návrh vymyslený tak aby sa kedykoľvek dal tento zoznam rozšíriť, hlavne ak by sa aplikácia rozšírila nejakým systémom REST API, to sa mi bohužiaľ nepodarilo implementovať v rozumnej forme tak aby som vedel dáta generovať zo servera pre klienta, nakoľko som nemal kde hostovať služby.

Ďalším systémom ako prehľadávať dáta je zobrazíť ich podľa zoznamu produktov, kde nájdeme všetky produkty ktoré sú distribuované v danej verzii databázy.



## Návrh architektúry aplikácie

Aplikácia je rozdelená na niekoľko častí:

- DB entity (tabuľky s atributami)
- Dao rozhrania
- ViewModel triedy
- Navigation
- Obrazovky
- Komponenty

### DB entity

Zoznam entít a ich popis:

- Category
  - entita ktorá popisuje kategóriu
- Currency
  - entita ktorá popisuje menu
- Item
  - entita ktorá popisuje položku
- ItemStorePrice
  - referenčná tabuľka ktorá prepája predmet s cenou a obchodom
- Price
  - entita ktorá popisuje cenu
- StoreItem
  - entita ktorá popisuje daný obchod

## Dao rozhrania

V rámci rozhraní som sa snažil využiť čo najlepší potenciál systému sqllite, preto sa snažím čo najviac výpočtového výkonu využiť pri využívaní dotazov na databázu. Okrem štandardných funkcií ktoré využívajú skoro všetci študenti v SP som sa rozhodol využiť pokročilejšie vyhľadávania, nakoľko ak máme možnosť dáta rovno obmedziť vo výbere tak vieme zrýchliť beh aplikácie.

Operácie nad DB náročné a pri väčšom množstve dát môžu aj dlhšie trvať, tak je rozumné ich optimalizovať tak aby sme docielili výber čo najmenej relevantných výsledkov

```

@Query("SELECT c.* FROM STORE s " +
        "join ItemStorePrice isp on (isp.refToStore = s.id) " +
        "join items i on i.id = isp.refToItem " +
        "join category c on c.id = i.refToCategory " +
        "where s.id = :storeId " +
        "group by c.id")
fun getCategoriesInStore(storeId: Int): Flow<List<Category>>
```

## ViewModel triedy

V rámci práce využívam niekoľko viewModel tried. Tieto triedy sú tzv. medzičlánkom medzi backendom a frontendom a uchovávajú informácie podľa potreby.

V rámci aplikácie využívam tieto viewModely:

- FormItemViewModel
- ItemScreenView
- TopBarViewModel

### FormItemViewModel

V rámci tohto modelu uchovávam stav formuláru, ktorý je súčasťou triedy InsertItemFormUiState.

V rámci tohto formuláru uchovávam tieto polia:

- Názov predmetu
- Obchod
- Výrobca
- Cena

Každý s uvedených parametrov sa mení v závislosti od vstupu od používateľa. Ak si je používateľ istý tým čo zadal môže kliknúť na tlačidlo vytvoriť, pokiaľ sú hodnoty vyplnené tak ho navhost vráti na zoznam položiek, v prípade že to nie je v poriadku tak sa nestane nič.

### TopBarViewModel

Ako je z názvu zrejmé jedná sa o viewModel k hornej lište aplikácie v rámci tohto viewModelu sa uchovávajú tieto informácie:

- Zoznam všetkých predmetov
- Zoznam všetkých obchodov
- Zoznam všetkých kategórií

Následne sa ešte uchováva stav toho či je menu hlavné alebo vedľajšie. Pre hlavné menu sú k dispozícii 3 možnosti výberu. Kategórie, Obchody a Predmety. Vedľajšie menu slúži pre vnorené operácie, ktoré nie sú na hlavnej obrazovke a má k dispozícii šípku späť, ktorá nás vráti na predchádzajúcu aplikáciu

## Navigation

V rámci aplikácie potrebujeme vedieť kde je čo uložené, preto využívame návrhový vzor navigation, konkrétne navhost, pomocou ktorého definujeme cestu, ktorá je vo formáte URI, táto cesta môže obsahovať aj parametre, ktoré musíme zadať, nakoľko nám ovplyvnia to čo sa nám má zobrazovať

```
composable(  
    route = "item/{id}",  
    arguments = listOf(  
        navArgument(name: "id") { type = NavType.IntType }  
    )  
) { this: AnimatedContentScope, backStackEntry ->  
    val itemId = backStackEntry.arguments?.getInt(key: "id")  
  
    if (itemId != null) {  
        ItemScreen(itemId = itemId)  
    }  
  
    topBarViewModel.setIsMain(false)  
}
```

V rámci aplikácie je to teda rozdelené na statické cesty a dynamické cesty. Statické cesty sú predovšetkým cesty ktoré sa používajú v hlavom menu a teda nie sú ovplyvnené zobrazovaným obsahom. Tieto statické cesty sú definované v enume TopBarSection

## Obrazovky a komponenty

Každá obrazovka sa skladá z komponentov. Pod komponentom v tomto prípade rozumieme sériu androidx komponentov ktoré sú zabalené v rámci nejakej composable metody. Spojením takýchto composable metod vytvoríme súvislo obrazovku.

V rámci aplikácie mám definované tieto obrazovky

- CreateItemScreen
- ItemScreen
- StoreScreen



## Finálna verzia obrazoviek

Ako bolo vyššie spomenuté aplikácia sa skladá z niekoľkých obrazoviek. Nižšie uvedené nájdete fotografie daných obrazoviek. Aplikácia podporuje Svetlý aj tmavý design, tento sa vyberá na základe voľby režimu v operačnom systéme.

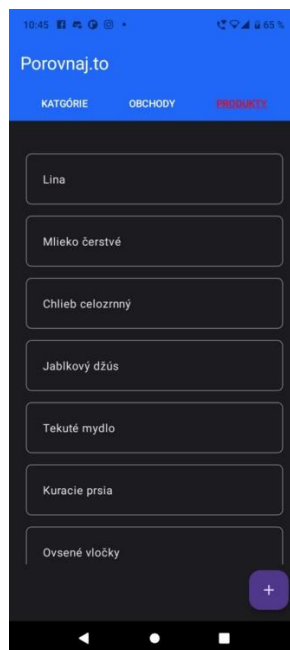
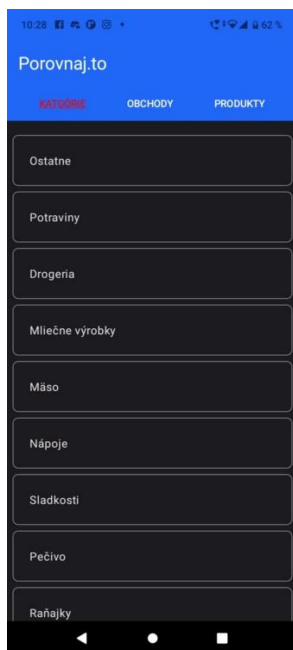
### Hlavná obrazovka

Na tejto obrazovke môžeme vidieť zoznam obchodov. Po kliknutí na logo alebo názov obchodu sa nám zobrazia kategórie položiek v danom obchode.



## Zoznam Kategórii a položiek

Po kliknutí na tlačidlo kategórie v hlavnom menu budeme presmerovaní na zoznam kategórii, v prípade že klikneme na obchod zobrazia sa nám len kategórie produktov ktoré sú v danom obchode. Veľmi podobný je zoznam položiek ktorý je vizuálne totožný zo zoznamom položiek, avšak rozdielom je obsah a interakcia



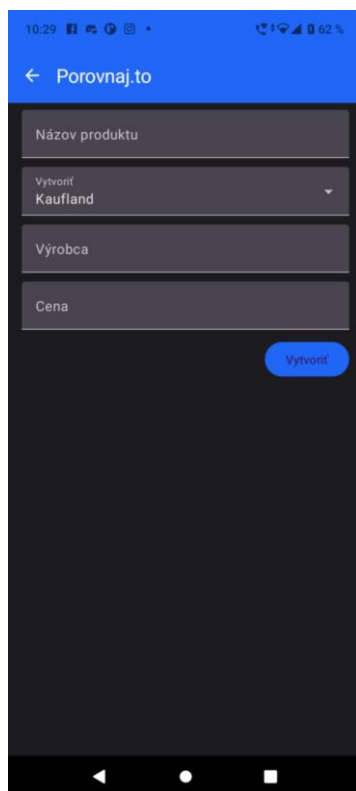
## Informácie o položke

Po vybratí konkrétnej položky zo zoznamu položiek môžeme vidieť samotnú položku resp jej názov, výrobcu, najnižšiu cenu a ostatné ceny v iných reťazcoch. Pokiaľ chýba ikona tak sa zobrazí namiesto ikony krížik



## Vytvorenie novej položky

Pokiaľ nám nejaký tovar chýba môžeme si ho vytvoriť, avšak musíme vedieť o ňom nejaké základné informácie (Názov položky, výrobcu, predajcu (obchod kde sa to predáva) a cenu). Po kliknutí na tlačidlo vytvoriť sa nám vytvorí daný produkt a uloží do lokálnej databázy



## Mapovanie a dostupné cesty

V tejto časti práce popisujem kde sa dá odkiaľ dostať a odkiaľ sa dá dostať kam

Z hlavnej obrazovky sa vieme dostať na tieto obrazovky

- Zoznam Kategórii
- Zoznam produktov
- Zoznam kategórii pre daný obchod

Zo zoznamov kategórii sa vieme dostať naspäť do hlavnej obrazovky, prípadne si vieme vybrať zobrazenie produktov pre danú kategóriu

Zo zoznamu produktov sa vieme dostať naspäť na zoznam kategórii, ale zároveň si vieme aj prezrieť produkt. Pokiaľ sme si vybrali celkový zoznam produktov tak sa vieme jednoducho dostať na hlavnú obrazovku. Na zozname produktov je v ľavom dolnom rohu tlačidlo + ktorým sa nám otvorí ďalšia obrazovka kde si vieme nový produkt vytvoriť, z tejto obrazovky sa vieme dostať naspäť na zoznam produktov



## Zoznam zdrojov

Prednášky a cvičenia predmetu VAMZ

<https://developer.android.com/reference/android/view/View>

<https://stackoverflow.com/questions/71743753/android-viewmodel-by-activityviewmodels-called-before-by-viewmodels>

Podobné aplikácie v obchode Play

- <https://play.google.com/store/apps/details?id=group.heureka.sk>
- <https://play.google.com/store/apps/details?id=sk.kimbinogreen.kimbino>
- <https://play.google.com/store/apps/details?id=com.kupi.kupino>

Inšpirácia dizajnom aplikácie mojeLetaky

- <https://play.google.com/store/apps/details?id=sk.mletaky>