**ŽILINSKÁ UNIVERZITA V ŽILINE**

FAKULTA RIADENIA A INFORMATIKY

Semestrálna práca

Vývoj aplikácií pre mobilné zariadenia

**Habit Tracker**

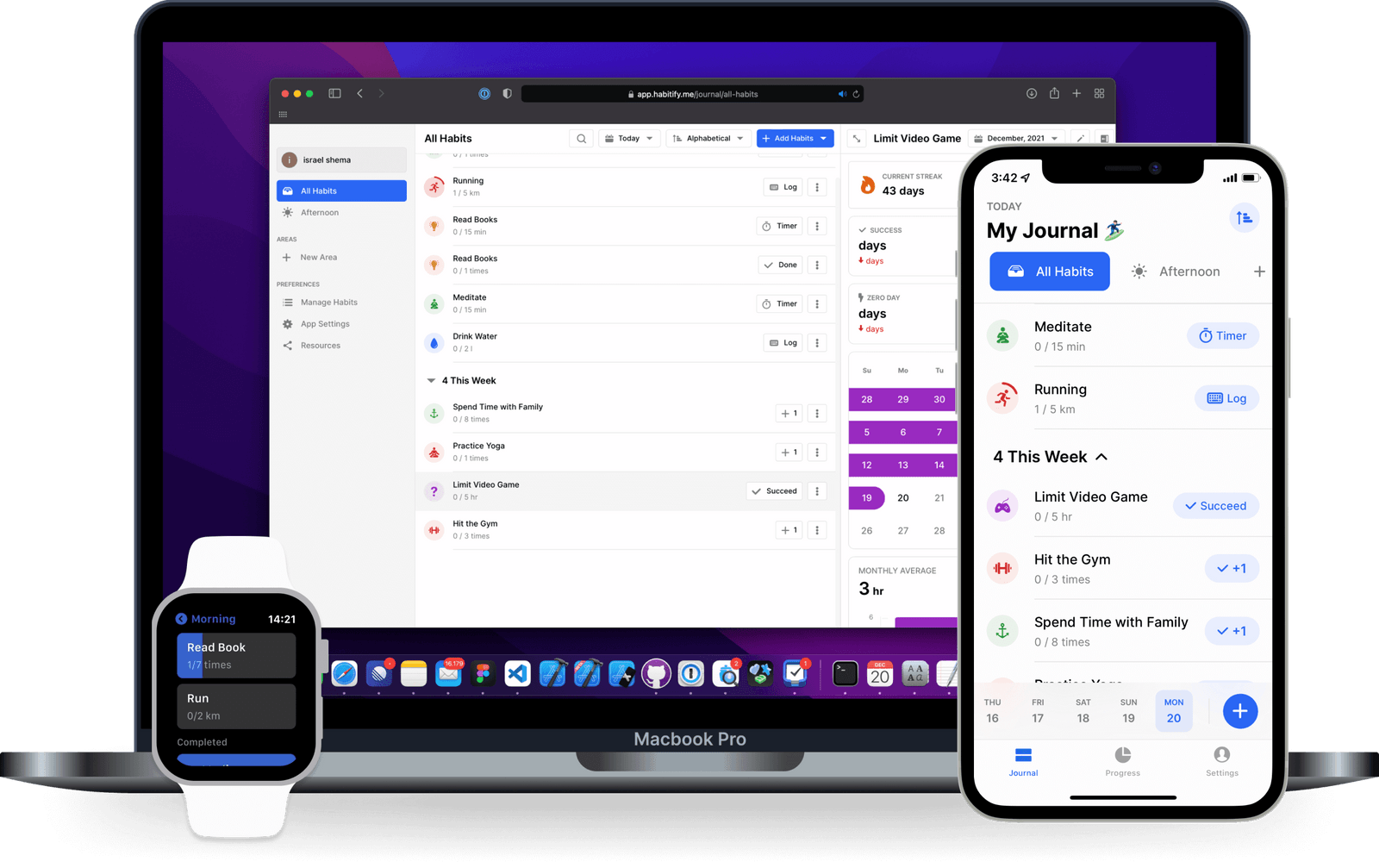
# Popis hry

HabitTracker, je aplikácia na sledovanie zvykov. Umožňuje používateľovi pridávať, mazať a upravovať zvyky. Pri vytváraní je potrebné zadať meno, informácie, obtiažnosť a dátum začatia zvyku. Aplikácia umožňuje hráčovi bojovať až 6 nepriateľov a to vždy po 7, 30, 60, 90, 180, 365 dňoch, čo slúži ako motivácia. Každý deň je potrebné plniť zvyky. Neplnenie denných návykov znižuje počet životov hráča.

# Prehľad podobných aplikácií

## Habitify

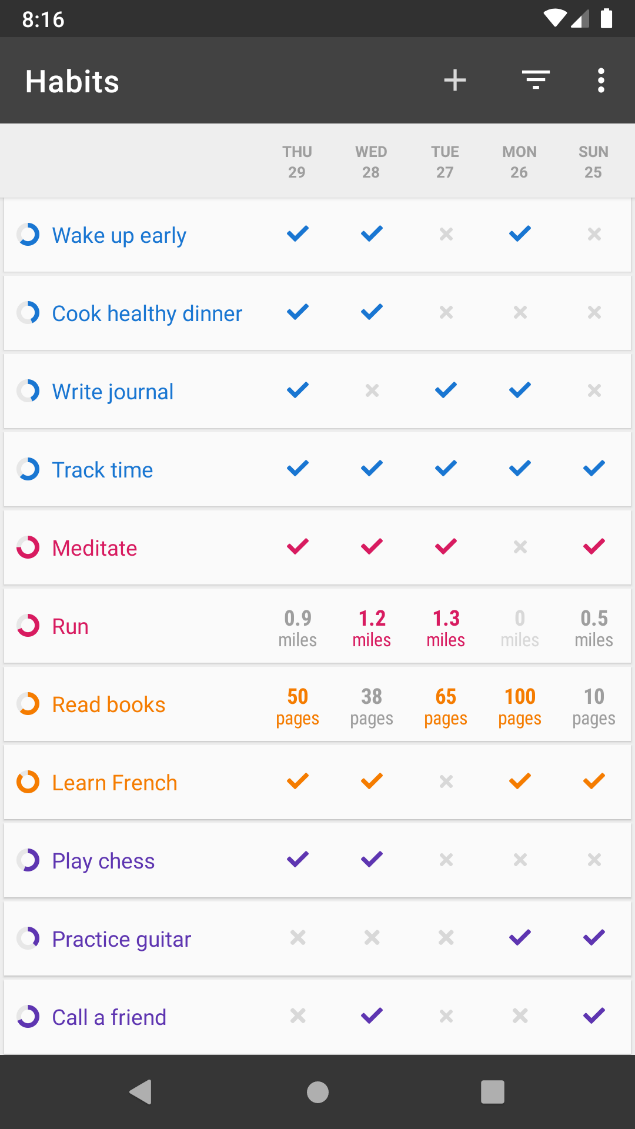
Habitify, umožnuje sledovanie a správu návykov a zvykov, poskytuje prehľadné štatistiky a grafy, ktoré umožňujú sledovať pokrok. Obsahuje kalendár kde sú presne vyznačené dni splnenia a nesplnenie zvykov. Dá sa vybrať deň začiatku.



Obrázok 1: Aplikácia Habitify

## Loop Habit Tracker

Podobné ako Habitify.

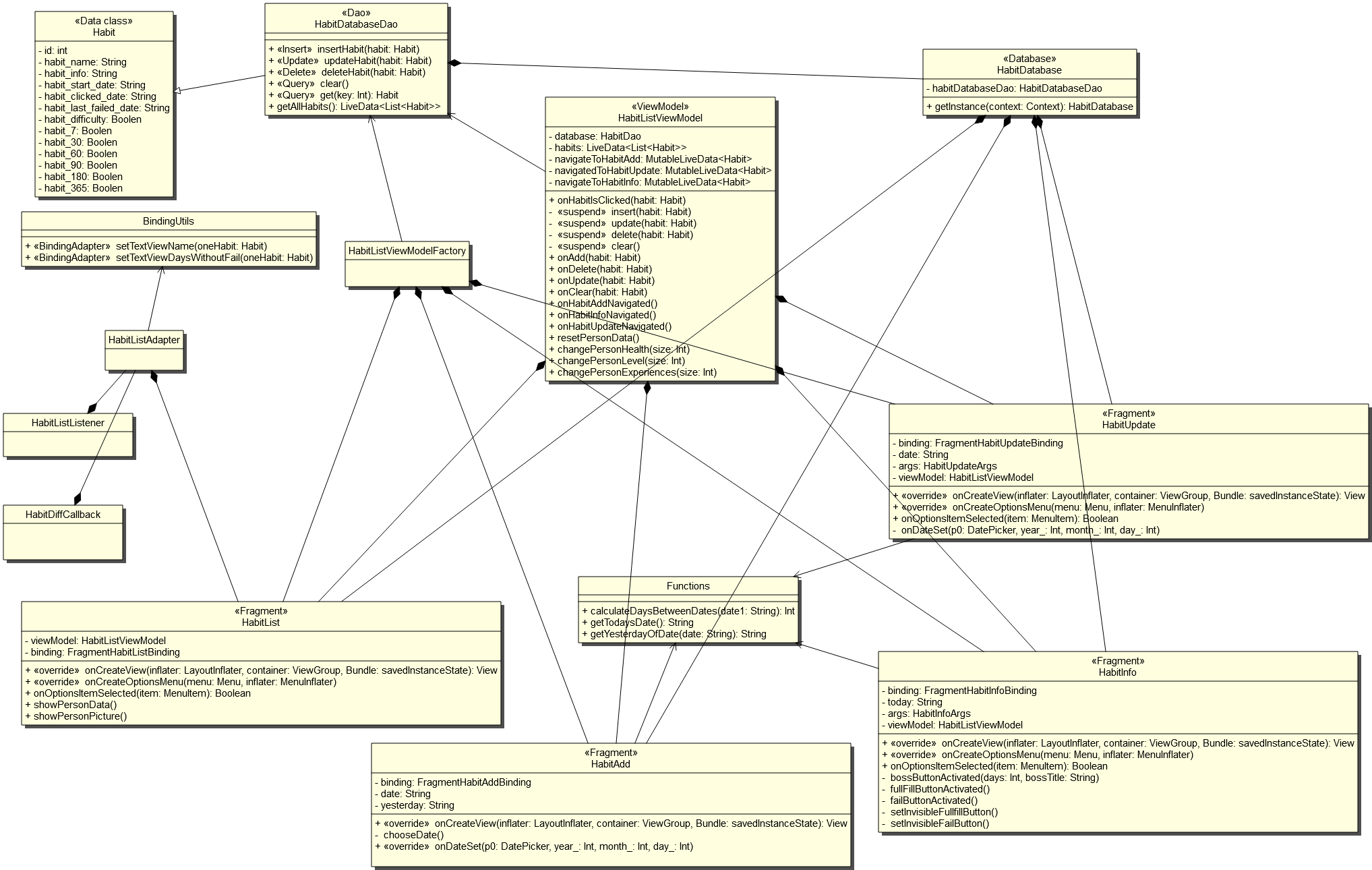


Obrázok 2: Aplikácia Loop Habit Tracker

Moja aplikácia je jednoduchšia na použivanie a líši sa napríklad tým že umožňuje bojovať nepriateľov.

# Návrh aplikácie

## UML



Obrázok 3: UML

# Popis Implementácie

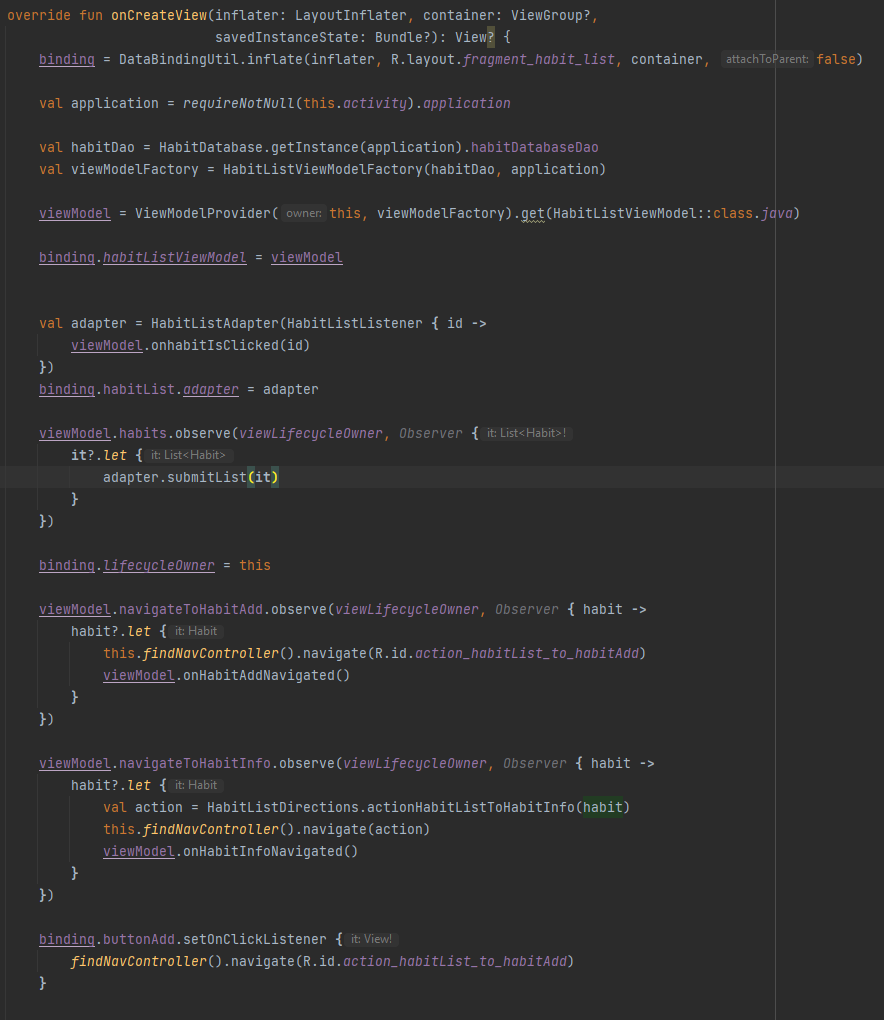
## HabitList (Fragment):

Obsahuje postavu, ktorá je vykreslená na základe údajov uložených v SharedPreferences. Postava môže byť živá alebo mŕtva čo je vyzobrazené pomocou ImageView.

Obsahuje RecyclerView, ktorý zobrazuje zvyky na obrazovku. RecyclerView používa adaptér (HabitListAdapter), aby zobrazil jednotlivé položky (Habit\_item) v zozname.

Zvyky sú uchovávané v databáze pomocou knižnice Room, a prístup k nim sa uskutočňuje pomocou HabitDatabaseDao.

Fragment komunikuje s HabitListViewModel, ktorý poskytuje prístup k údajom a logike spojenej so zvykmi.



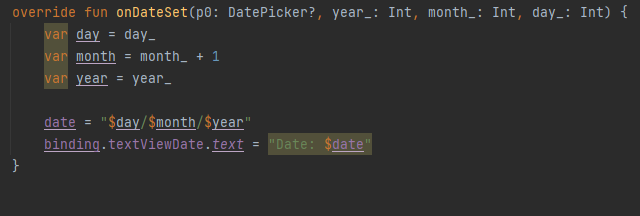
Obrázok 4: Ukážka kódu z triedy HabitList

## HabitAdd (Fragment):

Slúži na pridávanie nového zvyku do databázy.

Obsahuje DatePickerDialog, ktorý umožňuje vybrať dátum začiatku zvyku.

Fragment komunikuje s HabitListViewModel pomocou HabitListViewModelFactory, ktorý poskytuje inštanciu HabitListViewModel a umožňuje prístup k databáze a logike spojenej s pridávaním zvykov.



Obrázok 5: Povinná metóda DatePickerDialogu

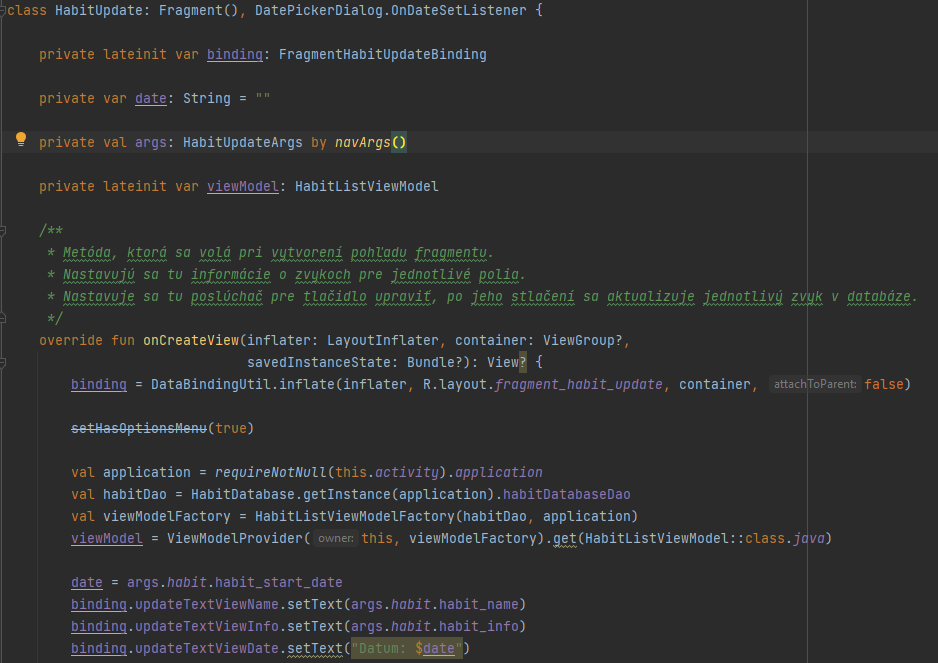
## HabitUpdate (Fragment):

Slúži na úpravu existujúceho zvyku v databáze.

Podobne ako HabitAdd, používa DatePickerDialog na vyberanie dátumu úpravy zvyku.

Údaje o zvyku sa posielajú z HabitList pomocou navArgs (argumenty navigácie).

Taktiež komunikuje s HabitListViewModel pomocou HabitListViewModelFactory na prístup k údajom a logike pre úpravu zvykov.



Obrázok 6: Ukážka kódu Fragmentu FragmentUpdate

## HabitInfo (Fragment):

Zobrazuje podrobné informácie o konkrétnom zvyku.

Taktiež umožňuje používateľovi bojovať s nepriateľom.

Údaje o zvyku, sa posielajú z HabitList pomocou navArgs.

Komunikuje s HabitListViewModel na prístup k údajom a logike spojenej s informáciami o zvyku. V triede sú použité Toast správy a AlertDialog.

A screen shot of a computer code

Description automatically generated with low confidence

Obrázok 7: Ukážka Toast správy a AlertDialogu

## Habit\_item (Layout):

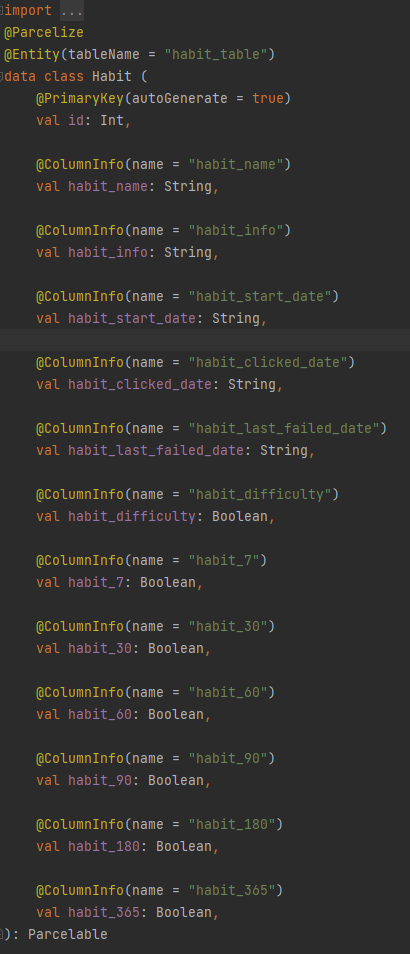
Reprezentuje jednu položku zoznamu zvykov v RecyclerView.

Obsahuje vizuálnu reprezentáciu zvyku a umožňuje interakciu s ním (napr. kliknutie pre úpravu alebo zobrazenie informácií).

## Ďalšie dôležité triedy:

## Habit

Predstavuje jednu entitu (zvyk) v databáze. Obsahuje všetky potrebné atribúty pre reprezentáciu zvyku. Je parcelable, takže je možné ho posielať pomocou navArgs.



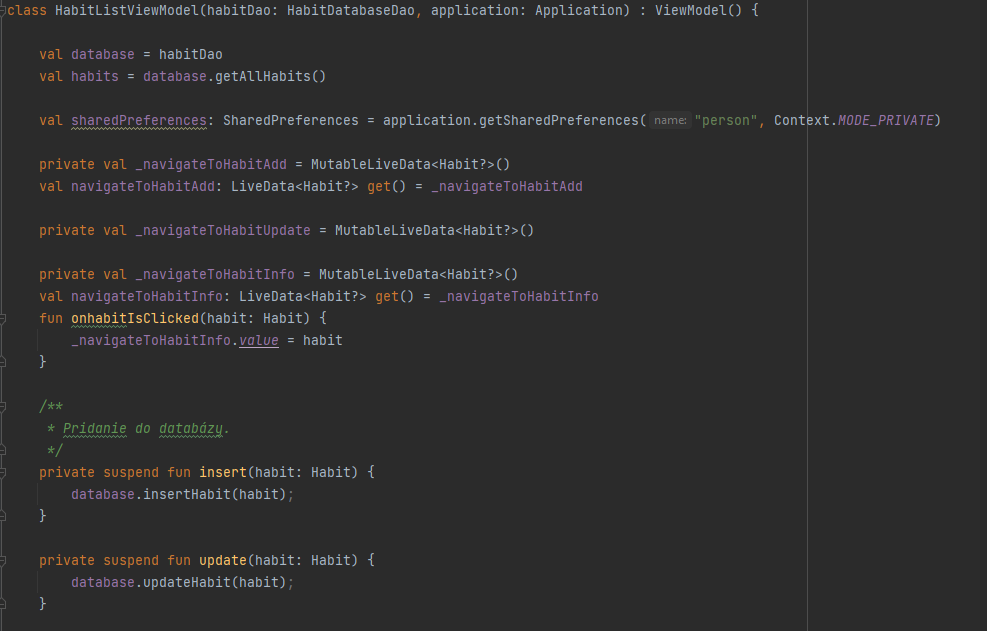
Obrázok 8: Trieda Habit

## HabitDatabaseDao

Obsahuje funkcie na prácu s databázou (pridávanie, aktualizovanie, odstraňovanie zvykov).

## HabitListViewModel

Poskytuje prístup k údajom a logike spojenej so zvykmi pre fragmenty. Komunikuje s HabitDatabaseDao a spravuje dáta pre zobrazenie v fragmentoch. Mení dáta postavy. Zvyky sú ukladané ako LiveData.



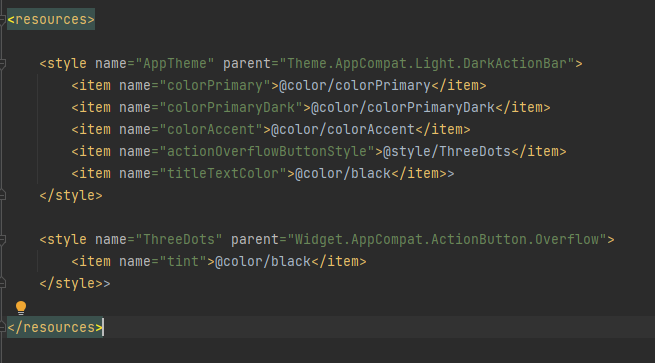
Obrázok 9: HabitListViewModel

## Functions

Obsahuje rôzne pomocné funkcie, napríklad vráti dnešný dátum, včerajší dátum alebo vypočíta počet dní od zadaného dátumu po dnešný dátum.

Tieto komponenty spolupracujú pomocou DataBinding (prístup k komponentom fragmentov) a Navigation (prehod medzi fragmentmi). Týmto sa dosahuje funkčnosť aplikácie na správu zvykov, pridávanie a úpravu údajov a zobrazenie podrobností o zvykoch.

Triedy HabitDatabse, HabitListAdapter a HabitListViewModelFactory sú použité z cvičení s minimálnymi zmenami.



Obrázok 10: Vlastná téma aplikácie

# Zdroje

[Obrázok 1] https://www.habitify.me/

[Obrázok 2] https://loophabits.org/