BeActive

Semestrálna práca pre predmet  
Vývoj aplikácií pre mobilné zariadenia (VAMZ)

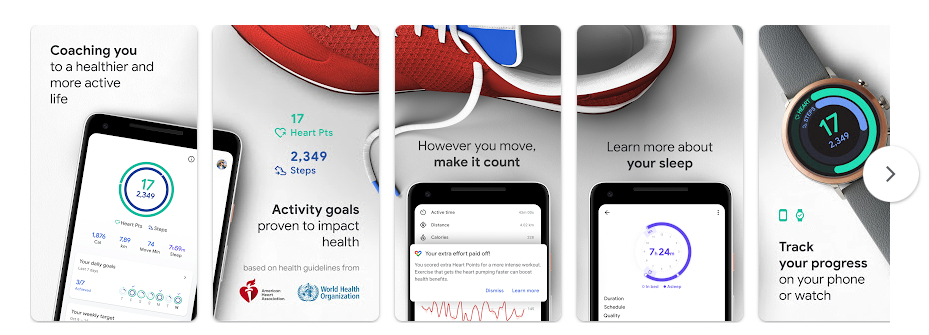
# Popis a analýza riešeného problému

## Špecifické zadania, definovanie problému

Aplikácia BeActive má byť intuitívnym riešením pre zaznamenávanie každodenných športových alebo oddychových aktivít. Aplikácia obsahuje údaje o aktivite ako napríklad počet krokov vykonaných pri aktivite, počet spálených kalórií, dĺžka času vykonávania aktivity a typ aktivity ktorý reprezentuje text (napr. Beh). Problémom takýchto aplikácií na trhu je veľká neprehľadnosť keďže obsahujú čoraz viacej funkcionalít ktoré znesprehľadňujú samotnú aplikáciu.

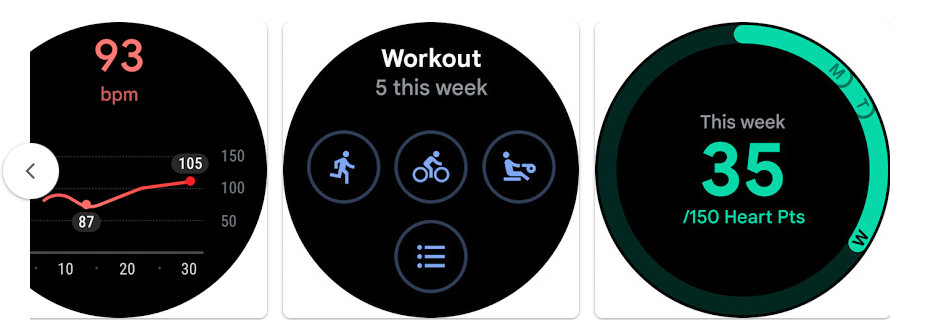
## Podobné aplikácie na trhu

### Google Fit: Sledovanie aktivity

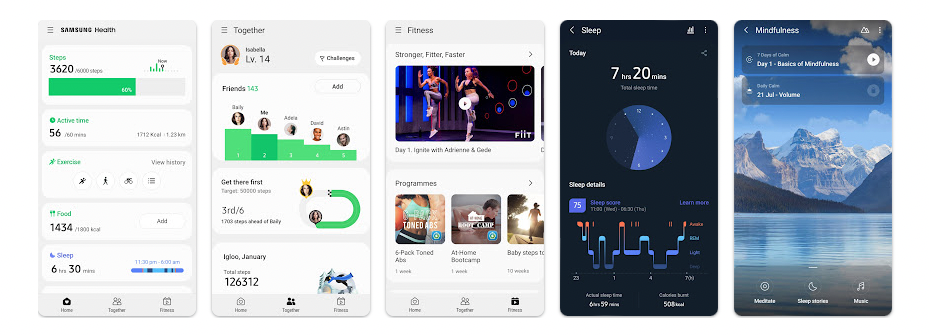


Aplikácia Google Fit od spoločnosti google je asi najlepšou „activity tracking“ aplikáciou na trhu. Obsahuje veľmi veľa „features“, ktoré lákajú veľmi veľa nových užívateľov. Je známa kompatibilitov s veľa typmi hodiniek a teda dokáže zaznamenávať aj tep srdca alebo krvý tlak.

Aplikácia ma teda veľkú výhodu v zdieľaní dát z externých zariadený ktorú moja aplikácia nepodporuje. Naopak mi príde ale trošku neprehľadná pokiaľ ju nepoužívate na hodinkách, na ktorých vyzerá jednoducho a prehľadne aj vďaka malému displeju hodiniek.



### Samsung Health



Aplikácia Samsung Health je ďalší konkurent mojej aplikácii. Samsung vytvoril z cvičenia a zaznamenávania aktivít zábavu s použitím relatívne jednoduchého UI a pridaním možnosti zdieľania a súťaženia v aktivitách spolu s vašimi priateľmi. Aplikácia ma nespočetné množstvo „features“. A však veľký problém tejto aplikácie je, že nepodporuje iné zariadenia ako Samsung. Je to škoda pretože aplikácia ma veľký potenciál ale musíte byť užívateľ Samsung zariadení.

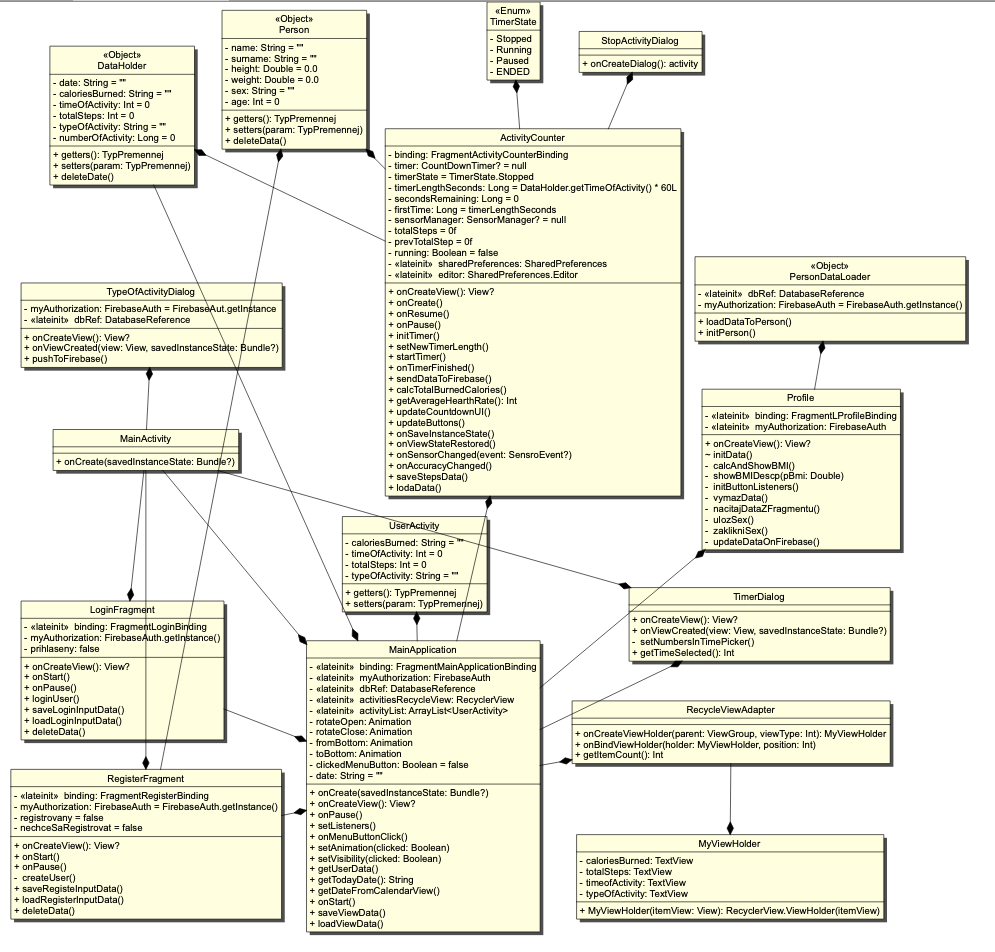
Moja aplikácia je teda kompatibilná so všetkými zariadeniami bezohľadu o akú značku zariadenia sa jedná.

# Návrh riešenia problému

## Krátka analýza

Predstavme si situáciu, že nie sme až taký športovo založený človek ale chceli by sme začal behať. Jednoduchým prihlásením pomocou vášho mailu ste v aplikácii. Spustíte si teda časomieru a tak si určíte cieľ koľko minút chcete bežať. Po uplynutí časovača vás aplikácia vyzve aby ste zadali názov (Typ) vašej aktivity. Po zadaní sa vám v aktuálnom dni zobrazí karta kde vidíte koľko ste spravili krokov, koľko ste spálili kalórií a koľko vám vaša aktivita trvala. Aj toto by mohla byť motivácia pre váš ďalší beh alebo inú aktivitu keď vidíte ako napredujete či už zvyšovaním dĺžky časomiery alebo narastajúcimi kalóriami.

## Návrh aplikácie



# Popis implementácia

## Aktivity a fragmenty

Obrázok, na ktorom je text, monitor, elektronika, iPod

Automaticky generovaný popis

Obrázok, na ktorom je text, monitor, iPod, obrazovka

Automaticky generovaný popis

Aplikácia sa skladá z 5 fragmentov a 1 aktivity. Majú koretkne implementovaný cyklus a ukľadajú prebežný stav buď cez SharedPrefferences alebo metódy onSaveInstace. Najzložitejší fragment bol na zaznamenávanie samotnej aktivity, keďže tam rátam kroky a zároveň aj odpočítavam čas vopred zvolený v Dialogovom okne ktoré som si tiež navrhol.

Obrázok, na ktorom je text, monitor, elektronika, iPod

Automaticky generovaný popis

Pod odmeraní aktivity sa zobrazí okno do ktorého zadáme typ aktivity vo forme textu ktorého dialogove okno som si tiež navrhol.

Obrázok, na ktorom je text, monitor, elektronika, iPod

Automaticky generovaný popis

Na prvej strane tejto sekcie je možné vidieť aj **MainApplicationFragment**, ktorý obsahuje calendarView a RecyclerView s mnou navrhnutým layoutom Card, ktorý zobrazuje všetky aktivity v danom zakliknutom dni.

Obrázok, na ktorom je text, monitor, elektronika, mobilný telefón

Automaticky generovaný popis

Všetky fragmenty taktiež reagujú na otočenie displeja so zachovaným stavom v nich.

## AndroidX komponenty (Navigation)

Hlavný komponent, ktorý som vyžíval je Navigation ktorý mi neskutočne uľahčil prácu s presmerovávaním na iné fragmenty a dialógy. Jeho schéma vyzerá nasledovne.

Obrázok, na ktorom je text, elektronika

Automaticky generovaný popis

Presmerovanie na daný fragment sa dá docieliť jednoduchým riadkom kódu a to...

**Obrázok, na ktorom je text

Automaticky generovaný popis**

Tu vidíme clickListener pre tlačidlo buttonAddActivity, ktorá nas presmeruje na timePickerDialog v ktorom si vyberieme náš časový cieľ aktivity a zároveň do objectu DataHolder uloží dnešný deň pre ďalšie použitie.

## Externé knižnice

### Firebase

Hlavnou súčasťou mojej aplikácie je komunikácia s Firebase Autentifikáciou a Firebase Realtime Database. O samotné spojenie sa stárá knižnica.

Načítavanie z firebase realtime databázy je realizované pomocou ValueListnerea na inštancii FirebaseDatabase triedy.

Obrázok, na ktorom je text

Automaticky generovaný popis

Pomocou tejto metódy načítavam dáta o aktivitách do ArrayListu ktorý sa následne zobrazí v recyclerViewe. RealtimeDatabasa je vpodste json subor. Vyzerá nejako takto...

Obrázok, na ktorom je text

Automaticky generovaný popis

V záložke activityData mám uložené všetky dáta o aktivitách. Pod ňou sa nachádzajú užívatelia vo forme ID, ktoré im priradil sám firebase pri Logine. Ďalej každý user obsahuje podbloky zložené z dátumu, kedy bola aktivita vykonávaná a následne sú poradové čísla jednotlivých aktivít.

Pre nahrávanie do databázy slúži metóda PushToFirebase.

Obrázok, na ktorom je text

Automaticky generovaný popis

Táto metóda načíta do dbRef inštanciu databázy podľa presnej cesty ku ktorej sa dostane cez vopred uložený dátum v objecte DataHolder. Najprv potrebujeme zistiť koľko aktivít sa nachádza v danom dni a ku tomuto číslu následne prirátame 1 a máme index ďalšejaktivity.

Obrázok, na ktorom je text

Automaticky generovaný popis

Metóda UpdateDataOnFirebase slúži na aktualizovanie profilových údajov o osobe, ktoré sú uložené v realtimeDatabase na ceste link-ku-DB/personData/idPerson.

Obrázok, na ktorom je stôl

Automaticky generovaný popis

### StepCounter sensor (krokomer)

Obrázok, na ktorom je text

Automaticky generovaný popis

Pre spustenie merania počtu krokov používam triedu SensorManager implemetovanú priamo v androide. Typ senzoru je Sensor.TYPE\_STEP\_COUNTER. Ak vaše zariadenie nepodporuje tento senzor, tak vyskočí Toast ktorý vás upozorní.

Obrázok, na ktorom je text

Automaticky generovaný popis

Trieda Sensor obsahuje metódu onSensroChanged. Táto metóda priraďuje počet krokov zo senzora do mojej lokálnej premennej totalSteps a zároveň ju zobrazuje v fragmente activityCounter v textViewKrokomer.

# Zoznam použitých zdrojov

Logo aplikácie, vytvorené pomocou -<https://www.clipartmax.com/download/m2i8i8K9Z5m2Z5G6_woman-sports-aerobics-fitness-gymnastics-aerobics-clipart/>

Material progress bar – <https://github.com/zhanghai/MaterialProgressBar>

FirebaseAuthentification – <https://www.youtube.com/watch?v=iSsa9OlQJms>

Prepojenie realtime DB do aplikácie - <https://www.youtube.com/watch?v=rFTJTLdoGDY&list=PLHQRWugvckFry9Q1OT6hLNfyUizT73PwX&index=2>

RecyclerView s custom templatom –<https://www.youtube.com/watch?v=VVXKVFyYQdQ&t=356s>

Floating action button menu – <https://www.youtube.com/watch?v=umCX1-Tq25k&t=49s>

Krokomer - <https://www.youtube.com/watch?v=WSx2a99kPY4>

Výpočet BMI –

<https://www.bmikalkulacka.sk/vysledky/eyJnZW5kZXIiOmZhbHNlLCJ3ZWlnaHQiOjc1LCJoZWlnaHQiOjE4MiwieWVhciI6MjAwMX0%3D?ref=98a514f13de0dd8bd3a4b3a2a1ba7843>

Počítanie kalórií – <https://www.quora.com/What-is-the-formula-to-calculate-calories-burnt-by-running>