

ŽILINSKÁ UNIVERZITA V ŽILINE

Fakulta riadenia a informatiky

Informatika

Vývoj aplikácií pre mobilné zariadenia

Semestrálna práca

2019/2020

Ondrej Jarina,

Zdenko Pečeňa

5ZY031

## Krátky popis aplikácie

Aplikácia je určená na vyhľadávanie športových aktivít v okolí a vytváranie nových športových aktivít. Používateľ môže jednoducho vytvoriť novú udalosť, na ktorú sa ostatní používatelia môžu prihlásiť.

Práca bola vytvorená počas štúdia na Univerzite v Záhrebe, na ktorom sme boli ako študenti počas študijného programu Erasmus+. Bola vytvorená v rámci predmetu Software Analysis and Design, ktorý bol zameraný na tvorbu aplikácií pre mobilné telefóny a jeho výstupom bola funkčná aplikácia pre systém Android.

## Aplikácia TeamMate

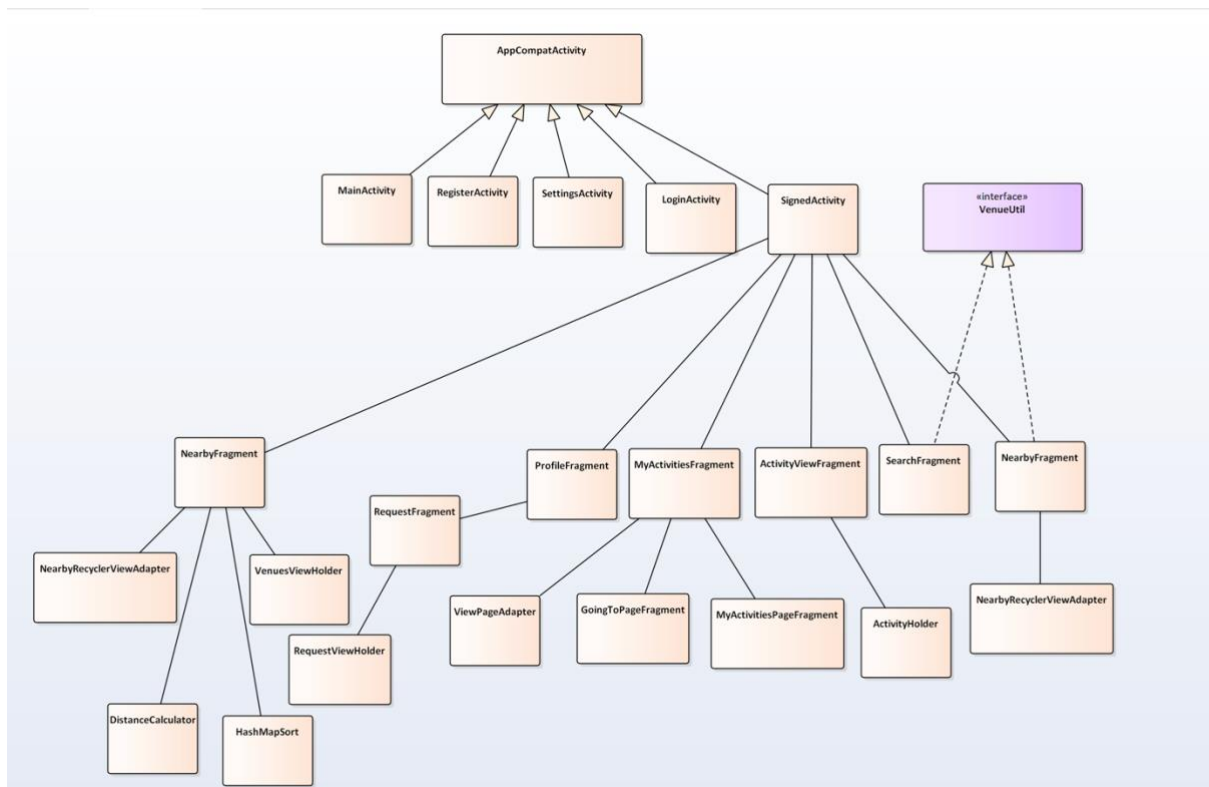
Vytvorili sme aplikáciu pre športovcov. Pri prvom spustení aplikácia ponúkne používateľovi zaregistrovať sa. Môže si vytvoriť nový účet pomocou e-mailu a hesla, alebo sa môže prihlásiť so svojim existujúcim Google účtom cez službu Google Login. Aplikácia vyhľadá pomocou polohy zariadenia najbližšie umiestnené športoviská, ktoré zobrazí užívateľovi a zoradí podľa vzdialenosti. Používateľ si zvolí športovisko zo zoznamu, v ktorom môže vytvoriť novú športovú udalosť, alebo sa pripojiť k inej, už vytvorenej, športovej aktivite. Aplikácia vyhľadáva športoviská z Google Maps a údaje k miestu (ako napríklad fotky) získa zo služby Google Places. Používateľ tiež môže vyhľadať konkrétne športovisko zadaním jeho názvu alebo adresy pomocou full-textového vyhľadávania.

## Prípady použitia

1. Registrácia nového používateľa
  - a. Každý neprihlásený používateľ má možnosť zaregistrovať sa / vytvoriť si účet na používanie aplikácie
  - b. Každý registrovaný používateľ, ktorý nie je aktuálne prihlásený má možnosť sa prihlásiť a získať prístup k svojmu účtu
2. Prehliadanie športovísk
  - a. Každý prihlásený používateľ vidí zoznam najbližších športovísk, zoradený podľa vzdialenosti
  - b. Používateľ môže zvoliť ľubovoľné športovisko zo zoznamu
  - c. Každé športovisko zobrazuje svoju vzdialenosť, názov a fotografiu
3. Prehliadanie aktivít
  - a. Každý prihlásený používateľ po zvolení športoviska má k dispozícii pohľad na všetky aktivity v danom športovisku
  - b. Každý prihlásený používateľ si môže prezrieť zoznam, v ktorom vidí aktivity, ktoré vytvoril a tiež aktivity, ktorých sa zúčastňuje
4. Vyhľadanie aktivít
  - a. Každý používateľ vie vyhľadať konkrétne športovisko zadaním jeho názvu, adresy

5. Vytvorenie aktivity
  - a. Každý prihlásený používateľ má možnosť vytvoriť športovú aktivitu v danej lokalite, s podmienkami a informáciami, ktoré si určí
6. Prihlásenie sa na aktivitu
  - a. Prihlásený používateľ sa môže zaregistrovať (poslať žiadosť o registráciu) na športovú aktivitu, ktorú vytvoril iný používateľ
7. Schválenie žiadostí o zaregistrovanie na aktivitu
  - a. Prihlásený používateľ, ktorý je tvorcom aktivity má možnosť schváliť alebo odmietnuť žiadosti o zúčastnenie sa ním vytvorenej aktivity
8. Zobrazenie profilu
  - a. Každý používateľ si môže zobraziť svoj profil, kde má informácie o svojom účte a profilovú fotografiu.
9. Odhlásenie z aplikácie / zrušenie účtu
  - a. Používateľ má možnosť sa odhlásiť z aplikácie a prihlásiť sa neskôr so svojím menom a heslom
  - b. Používateľ má možnosť odstrániť svoj účet natrvalo, kedy sú odstránené všetky jeho údaje

## Diagram tried



## Popis implementácie

### Aktivity a Fragmenty

#### Aktivity

- RegisterActivity
  - Aktivita, ktorá sa spustí pri prvom spustení aplikácie
  - Pomocou nej sa vie nový používateľ zaregistrovať
  - Ako layout používa AndroidX Constraint layout
  - Metódy
    - onCreate
      - vytvorí polia na vyplnenie údajov (e-mail, heslo)
      - inicializuje Firebase, Google prihlasovanie
    - signIn
      - prihlásenie pomocou Google sign in
    - onLoginClicked
      - ukončí aktivitu a spustí LoginActivity
      - na prihlásenie pomocou existujúceho účtu
    - onRegisterClicked
      - skontroluje, či sú všetky polia vyplnené správne
      - ak používateľ už existuje, tak nepovolí vytvorenie účtu
      - ak neexistuje, vytvorí účet pomocou createNewPlayerWithEmail
    - onActivityResult
      - po použití Google sign in zavolá metódu handleSignInResult
    - handleSignInResult
      - skontroluje, či je Google Sign In úspešné
      - ak áno, overí totožnosť cez FirebaseAuth
    - FirebaseAuth
      - Pripojí sa do databázy používateľov, skontroluje, či používateľ existuje alebo nie
      - Ak neexistuje účet, pridá nový účet do kolekcie players v databáze, použitím metódy createNewPlayer
      - Užívateľa zistí podľa unikátneho identifikátora UID
    - createNewPlayer
      - vytvorí nového používateľa z Google účtu
      - získa Meno, Profilovú fotku, telefónne číslo
      - pridá do kolekcie players
    - createNewPlayerWithEmail
      - vytvorí nového používateľa z e-mailu a hesla
      - pridá do kolekcie players
    - switchActivity
      - po úspešnom prihlásení ukončí aktivitu a spustí SignedActivity
- LoginActivity

- prihlásenie používateľa pomocou existujúceho účtu
- používa AndroidX Constrain Layout
- Metódy
  - onCreate
    - vytvorí polia na vyplnenie údajov (meno, heslo)
    - inicializuje Firebase
  - loginButtonClicked
    - skontroluje, či sú polia správne vyplnené
    - ak nie, vypíše Toast správu s chybou
    - prebehne overenie pomocou Firebase
- SettingsActivity
  - AndroidX Constraint Layout
  - Na zmenu nastavení (telefónne číslo, odstránenie účtu)
  - Metódy
    - onCreate
      - vytvorenie tlačidiel na uloženie nastavení, neuloženie nastavení, vymazanie účtu
    - onSaveClicked
      - aktualizuje atribút phone v kolekcii players u konkrétneho používateľa
    - onDeleteClicked
      - vymaže používateľa z databázy firebase
    - onDiscardClicked
      - ukončí aktivitu, neurobí žiadne zmeny
- MainActivity
  - Metódy
    - onCreate
      - spustí Register Activity (tá kontroluje, či používateľ je už prihlásený)
- SignedActivity
  - „hlavná“ aktivita aplikácie
  - funguje len po prihlásení používateľa
  - Metódy
    - onCreate
      - vytvorí navigačné tlačidlá na spodu, inštanciu triedy LocationProvider
    - onRequestPermissionsResult
      - skontroluje, či aplikácia má povolený prístup k polohe
    - createNavigationListener
      - vytvorí listener na fragmenty
    - onResume
      - aktualizuje aktuálnu polohu

## Fragmenty

- ActivityViewFragment
  - zobrazenie zvolenej športovej aktivity zo zoznamu
  - používa Android Relative Layout

- Metódy
  - createPictureUrl
    - získa aktuálnu fotografiu miesta z Google Places
  - setData
    - získa dáta k tejto aktivite
- MyActivitiesFragment
  - GoingToPageFragment
    - requestActivities
      - získa aktivity, ktorých sa používateľ zúčastňuje z databázy
      - vloží ich do Firestore Recycler Adapteru
    - createQuery
      - vytvorí požiadavku na všetky aktivity používateľa pomocou userId
      - vráti všetky aktivity používateľa
  - MyActivitiesPageFragment
    - requestActivities
      - získa aktivity, ktoré používateľ vytvoril
    - createQuery
      - získa všetky aktivity, kde creatorId je rovnaké ako ID používateľa userId
- NearbyFragment
  - zobrazí zoznam všetkých športovísk v okolí
  - Android Relative Layout
  - Metódy
    - onClickListener
      - animácia pri klikaní
    - getAllVenues
      - získa zoznam všetkých športovísk z databázy
    - createObjects
      - vytvorí jednotlivé objekty currentVenue – okrem ostatných atribútov obsahujú aj aktuálnu vzdialenosť od zariadenia
    - setAdapter
      - nastaví Recycler View Adapter
- NewActivityFragment
  - aktivita na vytvorenie novej športovej aktivity v danom športovisku
  - Android Relative Layout
  - Metódy
    - onCreateView
      - vytvorí aktivitu a uloží do databázy
    - checkCorrectData
      - skontroluje, či zadané dáta sú správne
- ProfileFragment
  - Metódy
    - onCreateView
      - získa z Firebase dáta o používateľovi
      - zobrazí profilovú fotografiu a ostatné údaje

- tu tiež sa dajú potvrdzovať požiadavky o pridanie do športových aktivít
  - createQuery
    - vytvorí požiadavku na získanie dát o používateľovi
  - logout
    - odhlásenie aktuálneho používateľa
- SearchFragment
  - full-textové vyhľadávanie
  - pomocou služby Algolia
  - Metódy
    - algoliaSearch
      - vyhľadá indexované dáta v databáze
      - AfterTextChanged
        - vyhľadáva a vracia výsledky po každej zmene textu
    - getAllVenues
      - získa všetky športoviská z databázy
    - setAdapter
      - vytvorí Firestore Recycler Adapter
- VenueViewFragment
  - Zobrazenie zvoleného športoviska zo zoznamu
  - Relative Layout, RecyclerView
  - Zobrazuje športovisko a všetky aktivity v ňom
  - Metódy
    - onCreateView
      - animácie pri prepínaní
      - zobrazí všetky aktivity
    - initialiseDB
      - inicializuje firebase databázu
    - picReferenceCreate
      - ak nemá obrázok (nemá PictureReference), získa základný obrázok
    - createPictureUrl
      - podľa PictureReference vytvorí URL
    - setViewAttributes
      - atribúty, ktoré sa zobrazia
        - názov, zemepisná šírka a dĺžka, obrázok
    - configureVenueRecyclerView
      - zobrazuje všetky aktivity v danom športovisku
    - hasExpired
      - kontroluje, či aktivity sú ešte platné, alebo už vypršali (podľa dátumu)

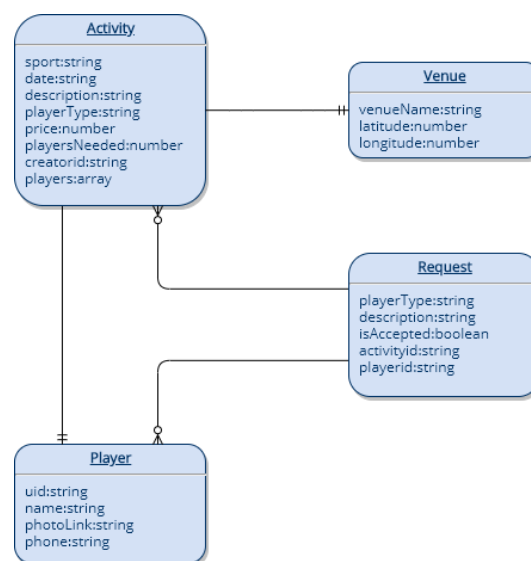


## Databáza

- Activity
  - Športová aktivita, vytvorená používateľmi
- Player
  - používateľ (používateľské účty)
- Request
  - Požiadavka na pridanie do aktivity (vytvorená používateľom)
- Venue
  - Športovisko, športová hala
  - Získané z Google Maps

Objekty v programe kopírujú objekty v databáze.

Doménový model



## Geolokačné dáta

- Geometry
- Location
- Photo
- PlacesResponse
- Result
- LocationProvider
  - získava aktuálnu polohu zariadenia
  - posíla požiadavku na získanie dát z Google Places API
  - používa Google Location knižnicu na získanie polohy

## Použité služby

### Firestore Cloud Firestore

Ako databázu sme zvolili Firestore Cloud Firestore, kvôli prepojeniu s Androidom, rýchlemu výkonu pri spracovávaní požiadaviek. Firestore tiež používa cache, takže aplikácia vie fungovať aj bez pripojenia na internet.

### Google Places SDK

Google Places SDK sme použili na vyhľadávanie športovísk v okolí. Na posielanie požiadavky používame knižnicu Volley (vieme posilať HTTP požiadavky) a na konvertovanie JSON objektu, ktorý z Google Places získame používame Gson.

### Algolia Search

Algolia Search je použitá na full-textové vyhľadávanie. Zvolil som ju preto, lebo ponúka veľmi dobré prepojenie s Firestore pomocou Firestore Cloud Functions. Bezplatný plán ponúka 50 tisíc operácií mesačne a 10 tisíc indexovaných záznamov.

## API kľúče

K prístupu k API k službám Google, Firestore a Algolia je potrebné vlastniť API kľúče. Tie z dôvodu bezpečnosti nie sú zverejnené cez verejný repozitár na Githube.