

Projekat - Spirala 3

Zadatak 1 [10B]

Koristeći web servis ([link](#)) implementirajte metode i repozitorije koji su navedeni u nastavku i povežite ih sa postojećim korisničkim interfejsom vaše aplikacije. Sve metode ovih repozitorija trebaju se izvršavati u okviru korutina koristeći IO dispatcher. Sve klase u nastavku trebaju se nalaziti u paketu **ba.etf.rma22.projekat.data.repositories**. Početna aktivnost treba imati naziv **MainActivity** i nalaziti se u paketu **ba.etf.rma22.projekat**.

Klase koje je potrebno implementirati:

ApiConfig - klasa koja čuva base url na web servis, defaultni je:

<https://rma22ws.herokuapp.com>

- `postaviBaseURL(baseUrl:String):Unit` - postavlja base url za web servise, ukoliko se ova metoda ne pozove koristi se defaultni url

AccountRepository

- `postaviHash(acHash:String):Boolean` - postavlja (lokalno, ne putem web servisa) hash studenta koji će biti korišten u drugim repozitorijima, vraća true ukoliko je hash postavljen, false inače.
- `getHash():String` - vraća hash studenta koji je postavljen

PitanjeAnketaRepository

- `getPitanja(idAnkete:Int):List<Pitanje>` - vraća sva pitanja na anketi sa zadanim id-em

TakeAnketaRepository

- `zapocniAnketu(idAnkete:Int):AnketaTaken` - kreira pokušaj za anketu, vraća kreirani pokušaj ili null u slučaju greške
- `getPoceteAnkete():List<AnketaTaken>` - vraća listu pokušaja ili null ukoliko student nema niti jednu započetu anketu

OdgovorRepository

- `getOdgovoriAnketa(idAnkete:Int):List<Odgovor>` - vraća listu odgovora za anketu, praznu listu ako student nije odgovarao ili nije upisan na zadanu anketu

- `postaviOdgovorAnketa(idAnketaTaken: Int, idPitanje: Int, odgovor: Int): Int` - funkcija postavlja odgovor sa indeksom odgovor na pitanje sa id-em idPitanje u okviru pokušaja sa id-em idAnketaTaken. Funkcija vraća progres (0-100) na anketi nakon odgovora ili -1 ukoliko ima neka greška u zahtjevu

AnketaRepository

- `getAll(offset: Int): List<Anketa>` - vraća listu svih anketa ili ako je zadan offset odgovarajući page u rezultatima (npr ako je pozvana metoda bez parametra vraćaju se sve ankete, a ako je offset 1 vraća se samo prvih 5)
- `getById(id: Int): Anketa` - vraća jednu anketu koja ima zadani id ili null ako anketa ne postoji
- `getUpisane(): List<Anketa>` - vraća listu svih anketa za grupe na kojima je student upisan

IstrazivanjeIstrazivanjeRepository

- `getIstrazivanja(offset: Int): List<Istrazivanje>` - vraća sva istraživanja ili ako je zadan offset odgovarajući page u rezultatima
- `getGrupe(): List<Grupa>` - vraća sve grupe
- `getGrupeZaIstrazivanje(idIstrazivanja: Int): List<Grupa>` - vraća grupe na istraživanju sa id-em idIstrazivanja
- `upisiUGrupu(idGrupa: Int): Boolean` - upisuje studenta u grupu sa id-em idGrupa i vraća true ili vraća false ako nije moguće upisati studenta
- `getUpisaneGrupe(): List<Grupa>` - vraća grupe u kojima je student upisan

Napomene

- Rok za rad na spirali 3 je **06.06.2021** do **23:59**. Rješenje se treba nalaziti na vašem privatnom repozitoriju (na master/default branchu). Prethodnu spiralu postavite na branch spirala2 (iz master brancha napravite branch spirala2 i onda se vratite na master da radite treću spiralu)
- Testovi se nalaze na repozitoriju
[bice dodani naknadno]
- **Nakon implementacije zadataka početni testovi moraju prolaziti, morate iskoristiti metode repozitorija za povezivanje sa web servisom**
- Implementirajte sve u skladu sa **MVVM** arhitekturom. U slučaju da ne koristite MVVM arhitekturu možete ostvariti maksimalno 80% bodova spirale.
- Spirala se pregleda korištenjem testova. Ispoštujte konvenciju imenovanja datu u postavci. Određeni testovi su sakriveni i izvršit će se tokom pregledanja spirale.
- Pitanja vezana za postavku spirale postavljajte kroz Piazza u folderu spirala3