

Projekat - Spirala 3

Koristeći web servise ([link1](#), [link2](#)) 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.rma23.projekat.data.repositories**. Početna aktivnost treba imati naziv **MainActivity** i nalaziti se u paketu **ba.etf.rma23.projekat**. Data klasu Game proširite sa potrebnim atributima. UserImpressions ćemo povezati sa web servisom na sljedećoj spirali, za sada igre koje dobijete sa web servisa neka imaju user impressions listu praznu.

Klase koje je potrebno implementirati:

IGDBApiConfig - klasa koja čuva base url na web servis IGDB i njegovu retrofit instancu.

GamesRepository -

- **getGamesByName**(name:String):List<Game> - vraća listu igara koje sadrže string u nazivu sa IGDB api-a
- **getGamesSafe**(name:String):List<Game> - vraća listu igara koje sadržavaju string u nazivu ali koje zadovoljavaju age rating za postavljene godine, ukoliko godine korisnika nisu postavljene vratite praznu listu
- **sortGames**():List<Game> - sortira igre koje su trenutno prikazane u aplikaciji po nazivu u poretku A-Z stavljajući prvo omiljene igre a ona ostale igre koje nisu dodane da su omiljene. Npr. Ako imamo igre sa nazivima A,B,C,D i igre B i D su omiljene, nakon sortiranja biće prikazane igre u redosljedu: B,D,A,C

AccountApiConfig - - klasa koja čuva base url na web servis rma23ws.onrender.com i njegovu retrofit instancu.

AccountGamesRepository -

- **setHash**(acHash:String):Boolean - postavlja (lokalno, ne putem web servisa) hash korisnika koji će biti korišten za identifikovanje accounta, vraća true ukoliko je hash postavljen, false inače.
- **getHash**():String - vraća hash korisnika koji je postavljen
- **getSavedGames**():List<Game> - vraća listu omiljenih igara
- **saveGame**(game:Game):Game - dodaje igru u listu omiljenih
- **removeGame**(id:Int):Boolean - briše igru iz liste omiljenih igara
- **removeNonSafe**():Boolean - briše sve igre koje ne zadovoljavaju age rating
- **getGamesContainingString**(query:String):List<Game> - vraća listu omiljenih igara koje sadrže query string u svom nazivu

- **setAge**(age: Int): Boolean - lokalno postavlja godine korisnika, vraća true ako su godine postavljene false ako nisu u rasponu od 3 do 100

Rok za rad na spirali 3 je **04.06.2023** 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 će biti dodani naknadno.

Nakon implementacije zadatka početni testovi moraju prolaziti, morate iskoristiti metode repozitorija za povezivanje sa web servisom

Spirala se pregleda korištenjem testova. Ispoštujte konvenciju imenovanja datu u postavci. Pitanja vezana za postavku spirale postavljajte kroz Piazzu u folderu spirala3