

Zadatak - Android Developer

Izraditi nativnu Android aplikaciju prema uputama koje slijede.

Aplikacija se sastoji od **tri ekrana**: liste git repozitorija, prikaza detalja odabranog repozitorija i lista odabranih favorita repozitorija i suradnika.

Prvi ekran je prikaz liste git repozitorija uz osnovne detalje navedene u nastavku. API endpoint se nalazi na sljedećem linku:

https://api.github.com/search/repositories?q=language:kotlin&order=desc&sort=stars&per_page=20&page=1

Informacije potrebne za prikazati su: owner login i avatar, repo name, description, language, stargazers_count, forks_count, open_issues i watchers_count. Također je potrebno prikazati ikonu za dodavanje i uklanjanje repozitorija s popisa favorita. Ikona mora imati dva različita stanja koja ovise o tome je li repozitorij već dodan na popis favorita. Pomicanjem liste do kraja radi se paginacija i učitavaju se novi repozitoriji. Klikom na repozitorij iz liste se prelazi na drugi ekran.

Potrebno je prikazati i traku za pretraživanje koja će na unos teksta pretraživati već prikazane repozitorije.

Drugi ekran treba prikazati detalje odabranog repozitorija koji se mogu preuzeti na sljedećem endpoint-u:

<https://api.github.com/repos/{owner}/{name}>

Informacije potrebne za prikazati su sve iste kao na listi repozitorija, uz dodatno repo default_branch, created_at, updated_at, listu suradnika (contributors_url), te button koji će na klik otvarati html_url u browseru. Također je potrebno prikazati ikonu za dodavanje ili uklanjanje repozitorija s popisa favorita, kao na početnom ekranu.

U listi suradnika potrebno je prikazati avatar i ime samog suradnika, kao i ikonu za dodavanje i uklanjanje suradnika s liste favorita.



Treći ekran služi za prikaz obje liste favorita, za što je potrebno koristiti dva taba - jedan za listu repozitorija, a drugi za listu suradnika. Na ovom ekranu je moguće samo ukloniti repozitorij ili suradnika iz liste favorita.

Za izradu ovog projekta je potrebno koristiti programski jezik **Kotlin**. Dopušteno je koristiti library-e koji vam mogu pomoći u izradi bilo kojeg dijela aplikacije.

Dodatne napomene:

- Potrebno je koristiti **MVVM arhitekturu**
- Za UI komponiranje koristiti **Jetpack Compose**
- Za dependency injection koristiti **Dagger Hilt**
- Za rad s lokalnom bazom podataka koristiti **Room**

Rješenje je potrebno poslati u obliku linka na git repozitorij (GitHub, GitLab, Bitbucket) na team.android@nsoft.com

Za sva eventualna pitanja i nejasnoće stojimo na raspolaganju putem gore navedene e-mail adrese.