

PiARS projektni zadatak – Shared Shopping List

Zadatak 4 – HTTP

Ovogodišnji projektni zadatak biće izrada Android aplikacije pod nazivom **Shared Shopping List**. Projekat se sastoji iz šest celina koje se nastavljaju jedna na drugu i zajedno čine kompletnu aplikaciju. Razvoj aplikacije se odvija u okruženju Android Studio, a rešenje je potrebno testirati na emulatoru ili Android uređaju.

Pred vama je tekst zadatka koji predstavlja četvrtu celinu. Ukupan broj bodova koji možete ostvariti na ovom zadatku je 10 bodova.

Zadaci

U aplikaciju je potrebno dodati podršku za komunikaciju sa Web serverom. Instalacija i pokretanje web servera su objašnjeni u dokumentu priloženom uz kod servera. Dokumentacija ruta servera je data u router.js datoteci iznad svake rute. Testirati rute korišćenjem [Postman](#) aplikacije. Pokretanjem koda na računaru, server će se nalaziti na adresi: **http://ipadresa računara:3000/**.

Primer dokumentacije

```
/**
 * @api {post} /users Create User
 * @apiName CreateUser
 * @apiGroup User
 *
 * @apiParam {String} username User's unique username.
 * @apiParam {String} password User's password.
 * @apiParam {String} email User's email.
 *
 * @apiSuccess {String} message User created successfully.
 * @apiError {String} message Error message.
 */
```

@api blok definiše rutu na kojoj se funkcionalnost nalazi kao i njenu HTTP metodu

@apiParam blok definiše parametre koje ima ruta (u primeru iznad, ruta očekuje 3 parametra u telu zahteva – username, password i email)

Primer slanja zahteva na ovu rutu korišćenjem Postman aplikacije:

POST ▼ http://192.168.242.100:3000/users

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

☐ none
 ☐ form-data
 ☒ x-www-form-urlencoded
 ☐ raw
 ☐ binary
 ☐ GraphQL

| Key | Value |
|--|---------------|
| <input checked="" type="checkbox"/> username | test |
| <input checked="" type="checkbox"/> password | test |
| <input checked="" type="checkbox"/> email | test@test.com |
| Key | Value |

```
/**
@api {delete} /lists/:username/:name Delete List
@apiName DeleteList
@apiGroup List
@apiParam {String} username User's username.
@apiParam {String} name List's name.
@apiSuccess {String} message List deleted successfully.
@apiError {String} message Error message.
*/
```

@api blok definiše DELETE rutu /lists/:username/:name gde *username* i *name* predstavljaju varijable koje je potrebno proslediti u URL-u rute. Primer pozivanje ove rute korišćenjem Postman aplikacije:

DELETE ▼ http://192.168.242.100:3000/lists/testuser/listname

Primetite da se parametri *username* i *name* nalaze u URL-u te ih nije potrebno dodatno slati.

Home dugme

Dodati Home dugme koje će biti vidljivo na svakoj stranici. Klikom na ovo dugme korisnika vratiti na početnu stranicu (Activity sa dugmadima za prijavu i registraciju).

Registracija korisnika

Prilikom registracije, registrovati korisnika na server slanjem POST zahteva na rutu **/users**. U telu zahteva je potrebno proslediti *username*, *password* i *email* kao što je opisano u dokumentaciji rute.

Na serveru je zabranjeno kreiranje korisnika sa istim *username* poljem. Ukoliko se pokuša registracija postojećeg korisnika, server će vratiti odgovarajuću grešku.

Funkcionalnost dodavanja korisnika u bazu ne treba menjati.

Prijava korisnika

Prilikom prijave korisnika na sistem potrebno je poslati POST zahtev na rutu **/login** sa odgovarajućim parametrima. Ukoliko korisnik koji pokušava prijavu postoji na serveru dozvoliti mu pristup. U suprotnom zabraniti pristup korisniku i prikazati Toast poruku sa odgovarajućim tekstom.

Dodavanje nove liste

Prilikom dodavanja nove liste obezbediti da se svaka **deljena** lista čuva na serveru slanjem POST zahteva na rutu **/lists**.

Prikaz svih lista

Prelaskom na WelcomeActivity prikazati sve liste prijavljenog korisnika čitanjem iz baze podataka (bilo da su one deljene ili ne).

U Activity je potrebno dodati *See Shared Lists* dugme. Pritiskom na ovo dugme prikazati spisak svih deljenih lista sa servera, slanjem GET zahteva na rutu **/lists**.

Brisanje deljene liste zahteva i brisanje liste sa servera. Listu obrisati slanjem DELETE zahteva na rutu **/lists/:username/:name** sa odgovarajućim parametrima.

Logiku prikazivanja i brisanja lista koje nisu deljene ne treba menjati.

Prikaz zadataka u listi

Ukoliko je lista **deljena**, potrebno je proširiti funkcionalnost dodavanja novog zadatka u listu čuvanjem zadatka na server. Čuvanje zadatka podrazumeva slanje POST zahteva na rutu **/tasks** sa definisanim parametrima.

Ukoliko lista nije deljena, funkcionalnost dodavanja zadatka se ne menja, odnosno svodi se na čuvanje zadatka u bazi podataka.

Ukoliko se prikazuje lista zadataka iz liste koja ne pripada korisniku (odnosno u ShowList Activity se došlo klikom na listu iz *See Shared Lists* liste) potrebno je prikazati dugme *Refresh*. Klikom na ovo dugme poslati GET zahtev sa odgovarajućim parametrima na rutu **/tasks/:list** radi dobavljanja svih zadataka povezanih sa odgovarajućom listom. Prikazati zadatke dobavljene sa servera.

Prilikom brisanja zadatka iz deljene liste potrebno je isti obrisati i sa servera slanjem DELETE zahteva na **/task/:id** rutu slanjem jedinstvenog identifikatora zadatka.