

PNRS 1 – Zadatak 4

HTPP

Implementacija zadatka 4 se nastavlja na zadatak 3, cime se nastavlja razvoj aplikacije pod nazivom **Chat Application**. U narednim poglavljima je dat detaljan opis potrebnih prosirenja.

Potrebno je nadograditi se na postojeći projektni zadatak, dodavanjem komunikacije sa HTTP serverom. HTTP server biće zajednički za sve klijente.

Komunikaciju sa serverom je potrebno ostvariti po uzoru na primere rađene na vežbama. Sa serverom u okviru aplikacije potrebno je ostvariti komunikaciju na sledećim mestima u Android aplikaciji:

- Prilikom **registrowanja novog korisnika** potrebno je poslati njegoo korisničko ime, šifru i email adresu na server, koristeći HTTP metodu POST, pozivom rute **POST /register**. Uspešnost registrovanja novog korisnika proveravati na osnovu odgovora server (Statusni kod za uspešno registrovanje je 200, a bezuspešno 4xx). Voditi računa da korisnika nekom vrstom teksta na ekranu obavestite o uspešnosti njegovog registrovanja (ukoliko nije uspešno predočiti razlog, npr. *"Email address should not be empty"*, ukoliko korisnik nije popunio to polje).
- Prilikom **login-a postojećeg korisnika** – potrebno je poslati njegovo korsiničko ime i šifru koristeći HTTP metodu POST, pozivom rute **POST /login**. Uspešnost akcije proveravati opet analizom odgovora od servera, posmatrajući statusni kod. Ukoliko korisnik ne uspe da obavi login, odgovarajućim tekstom ga obavestiti o razlogu (npr. *"Entered password is invalid"*). Prilikom uspešnog login-a, pored statusnog koda, potrebno je proveriti i sačuvati i primljeni sessionId, koji će za server predstavljati jedinstveni identifikator tog korisnika, koji će biti kasnije potrebno slati prilikom poziva ostalih HTTP ruta.
- Prilikom **pribavljanja liste kontakata** tog korisnika, pozivom HTTP rute **GET /contacts**, koristeći HTTP metodu GET. Od parametara je potrebno poslati sessionId. Uspešan odgovor korisnika rezultuje listom kontakata, a bezuspešan odgovarajućim kodom i razlogom greške. Potrebno je napraviti i **dugme za refresh**, na čiji klik će se pozivati ova ruta.

- Prilikom **slanja poruke nekom korisniku** , korišćenjem HTTP metode POST, pozivom rute **POST /message**. Od parametara je potrebno poslati kome šaljemo poruku i tekst poruke, kao i prethodno dobijeni sessionId. Uspešnost slanja proveravati u odgovoru od server i odgovarajuće reagovati na njegovu uspešnost, obaveštavajući korisnika o eventualnom razlogu bezuspešnog slanja.
- Prilikom **prihvatanja poruke od korisnika**, koristeći HTTP metodu GET, pozivom rute **GET /message/:id**. U ovoj ruti, :id je parameter koji se šalje kao deo rute i predstavlja id kontakta od kog želimo da prihvatimo poruku, a ne u okviru body-ja ili headera. Takođe je potrebno poslati i sessionId. Uspešan poziv rute rezultuje u nizu poruka, složenih po datumu slanja. Odgovarajuće obavestiti korisnika o eventualnoj grešci prilikom prihvatanja poruka. Potrebno je napraviti i **dugme za refresh u okviru ovog layout-a**, na čiji klik će se pozivati ova ruta i osvežiti lista poruka.
- Prilikom logout-a korisnika iz aplikacije, koristeći HTTP metodu POST, pozivom rute **POST /logout**. Od parametara je potrebno poslati sessionId. Odgovarajuće obavestiti korisnika o uspešnosti akcije.

U prilogu zadatka se nalazi README.txt datoteka koja sadrži pozive HTTP ruta i koji podaci se šalju na njih i na koji način.

Srećno!