

Kod izrade mobilne aplikacije na temu OpenStreetMap, poslužili su nam neki od izvora koje smo koristili i prilikom izrade web aplikacije na ovom linku:

<https://wiki.openstreetmap.org/wiki/Android>

Pod odjeljkom Libraries for developers/Maps odabrali smo Mapbox Android SDK.

Na ovom linku <https://docs.mapbox.com/android/maps/overview/> smo pronašli dobar uvod, start i ideju kako da u naš projekat implementiramo kod koji se tu nalazi.

Na stranici: <https://docs.mapbox.com/playground/marker> smo pronašli različite prikaze mape što čini jedan dio ovog projekta. Kada generišemo koordinate na navedenom linku, dobijemo starter code i access token. Taj kod smo iskoristili u Main Activity u kombinaciji sa dijelom koda vezanim za izradu markera iz anotacija sa linka navedenog prije. Za izradu ove aplikacije bilo je potrebno da napravimo svoj server za šta su nam koristile upute sa slijedećih stranica: <https://medium.com/@adnanrahic/hello-world-app-with-node-js-and-express-c1eb7cfa8a30>

<https://www.freecodecamp.org/news/how-to-deploy-a-nodejs-app-to-heroku-from-github-without-installing-heroku-on-your-machine-433bec770efe/>

Kada smo povezali GitHub i Heroku dobili smo odgovarajući link našeg servera koji smo potom uključili u OpenStreetMap.java. Nismo koristili cloud baze nego smo napravili listu JSON objekata čime smo olakšali pozivanje. Dakle, na serveru se nalaze odabrane atrakcije koje smo odlučili prikazati markerima na našoj mapi. Te atrakcije predstavljaju elemente od kojih svaki ima svoje ime, langitude i longitude. Pozivanje smo olakšali kreiranjem jednog niza (options) unutar kojeg smo napravili request i definisali da nam vraća JSONArray. Da bi dodali ikonicu za svaki element, unutar tog requesta kreiramo for petlju, uzimamo njegov objekat i kupimo potrebne vrijednosti. Sam prikaz se omogućuje kada prikupljeno prosljedimo symbol manageru.

Za različite prikaze mape i Login smo koristili LinearLayout(vertical) te za buttons (back, show markers, logout) horizontal LinearLayout, dok smo za sam prikaz mape(mapView) koristili FrameLayout.

Zanimljivo je možda pomenuti, kako u AndroidStudiu svaki string ima svoje life cycle metode tako i MapView (MapBox) ima svoje vlastite. Iz tog razloga override-amo istu metodu dva puta.

U funkciji authentication imamo varijablu savedata u kojoj čuvamo podatke koje vraćamo kao JSON pa pretvaramo u string jer smo u klasi ResponseListener definisali da prima string. Kada zahtjev dođe na server izvršava se funkcija authentication sa svoja dva parametra (req, res, next) i server daje odgovor na osnovu toga da li je user pronađen ili nije u skladu s čime se također ispisuju toast poruke.

Kao i kod web aplikacije, želimo naglasiti da je i u ovom slučaju, po nama, najbitniji faktor za uspješnu izradu bio dobro istraživanje, što je uveliko olakšalo cjelokupan proces.

*Na projektu radili:*

*Komić Mersiha, Muhović Belmin, Šehić Emira, Topalović Ezudina*