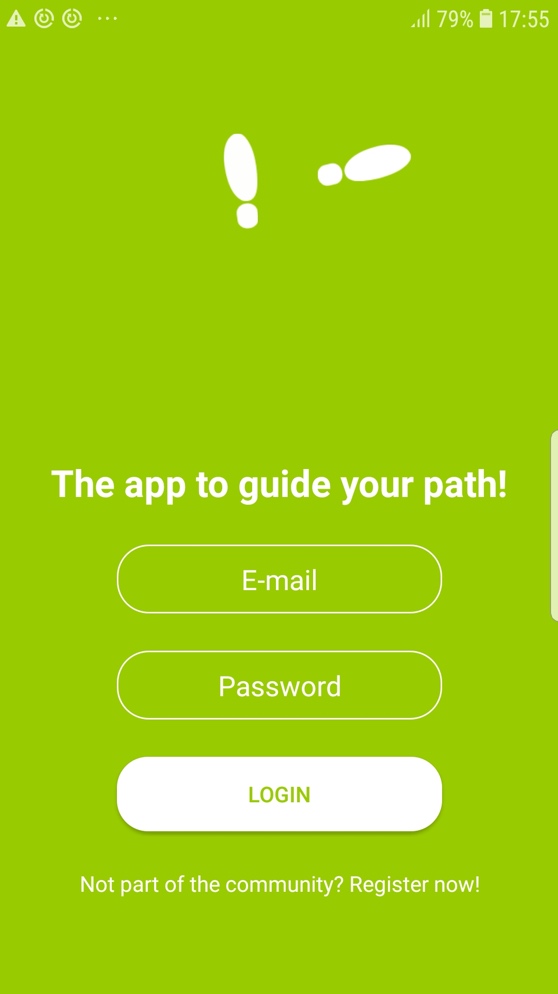
Aplicatie mobile pentru rute de transport in comun

Aplicatia UrbanMobility permite utilizatorilor vizualizarea rutelor de transport in comun.

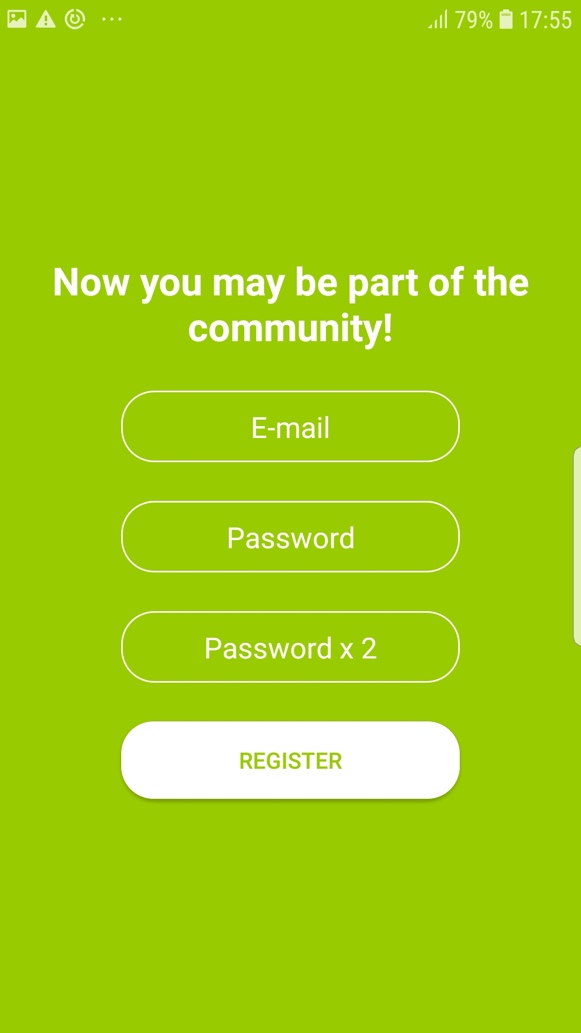


*Fig. 1 Ecran login*

Autentificarea in aplicatie este realizata prin intermediul unui ecran de login (Fig. 1).

Autentificarea este realizata prin intermediul Firebase Authentication.

Inregistrearea in aplicatie este realizata prin intermediul ecranului de inregistrare (Fig. 2). Datele introduse sunt validate si trimise catre Firebase Auth pentru inregistrare.

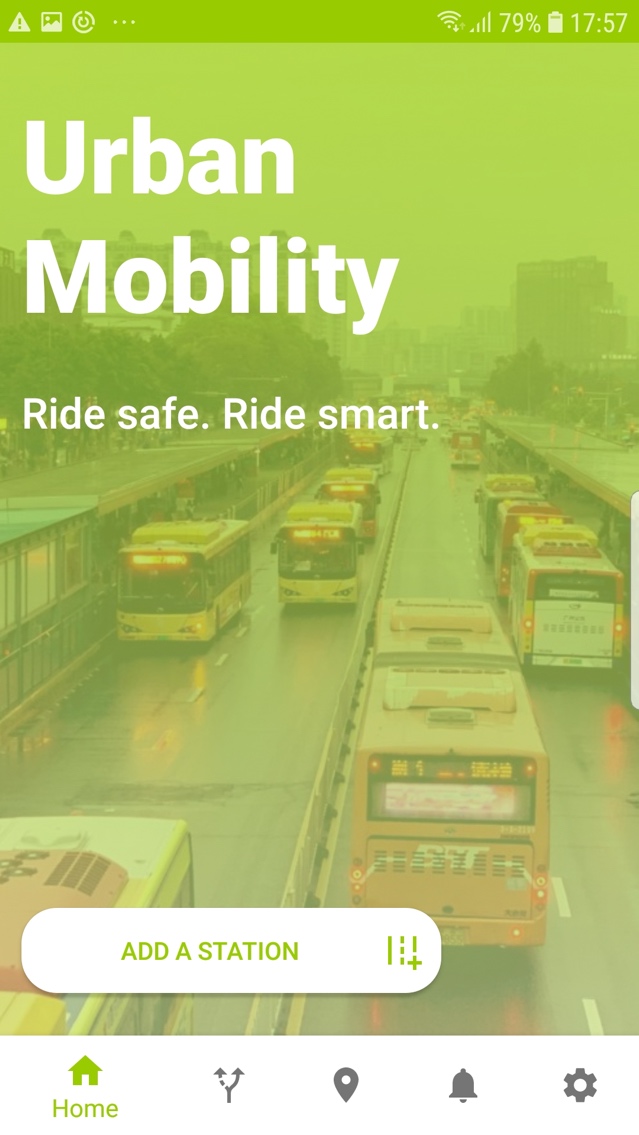


*Fig. 2 Ecran register*

Navigarea intre fragmente este realizata prin intermediul unui bottom navigation menu.

Bottom navigation menu-ul este compus din 5 fragmente:

1. Fragmentul home
2. Fragmentul rute
3. Fragmentul locatie
4. Fragmentul notificari
5. Fragmentul setari



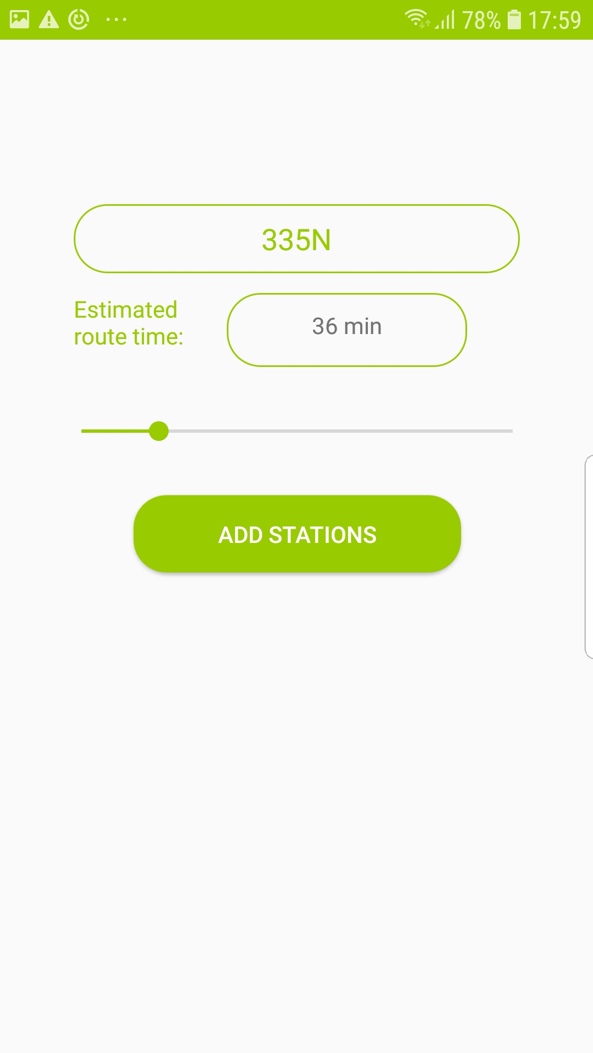
*Fig. 3 Ecran home*

Ecranul de home (Fig. 3) contine butonul de adaugare care permite utilizatorilor adaugarea unei noi rute. La apasarea butonului de add a route va fi afisat ecranul de adaugare a unei noi rute (Fig. 4)

Ecranul de adaugare a unei noi rute (Fig. 4) este compus dintr-un EditText si un SeekBar.

EditText-ul permite utilizatorului introducerea numelui unei rute (ex. 335N).

Seekbar-ul permite utilizatorului introducerea unui timp estimat al calatoriei pe aceasta ruta.



*Fig. 4 Ecran adaugare ruta*

Butonul add stations va deschide activitatea de adaugare statii (Fig. 5).

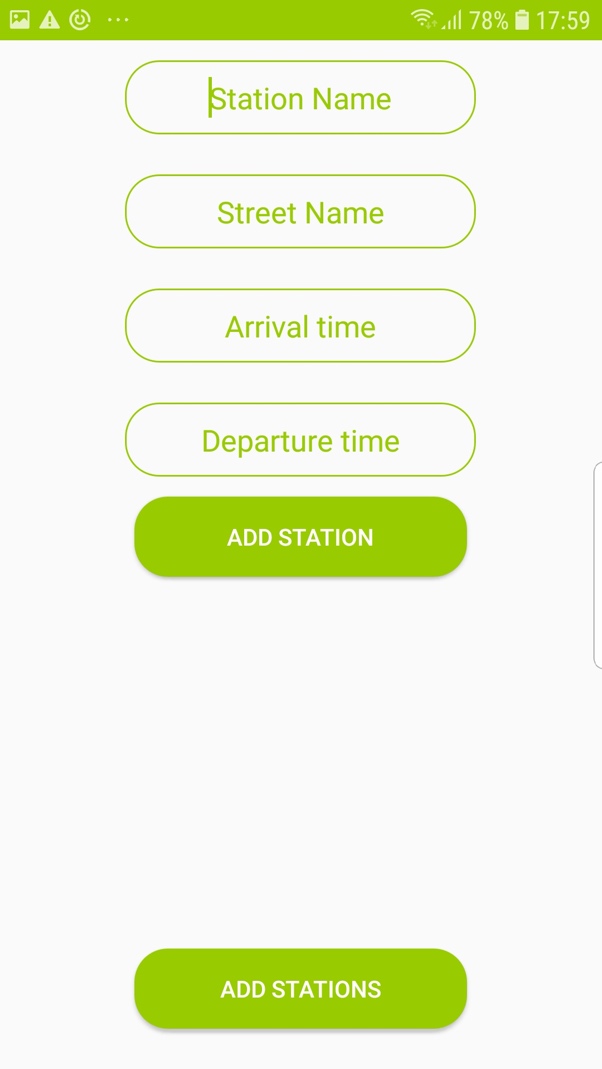
Ecranul de adaugare statii este compus din 4 editText-uri si doua butoane. EditText-urile permit utilizatorului introducerea urmatoarelor detalii despre o statie:  
 1. numele statiei

2. numele strazii

3. ora de sosire a autobuzului la statia respectiva

4. ora de sosire a autobuzului la statia respectiva

Utilizatorul poate adauga mai multe statii prin apasarea butonului Add Station. Daca doreste sa finalizeze adaugarea poate apasa pe butonul Add Stations care il va intoarce in ecranul de home.

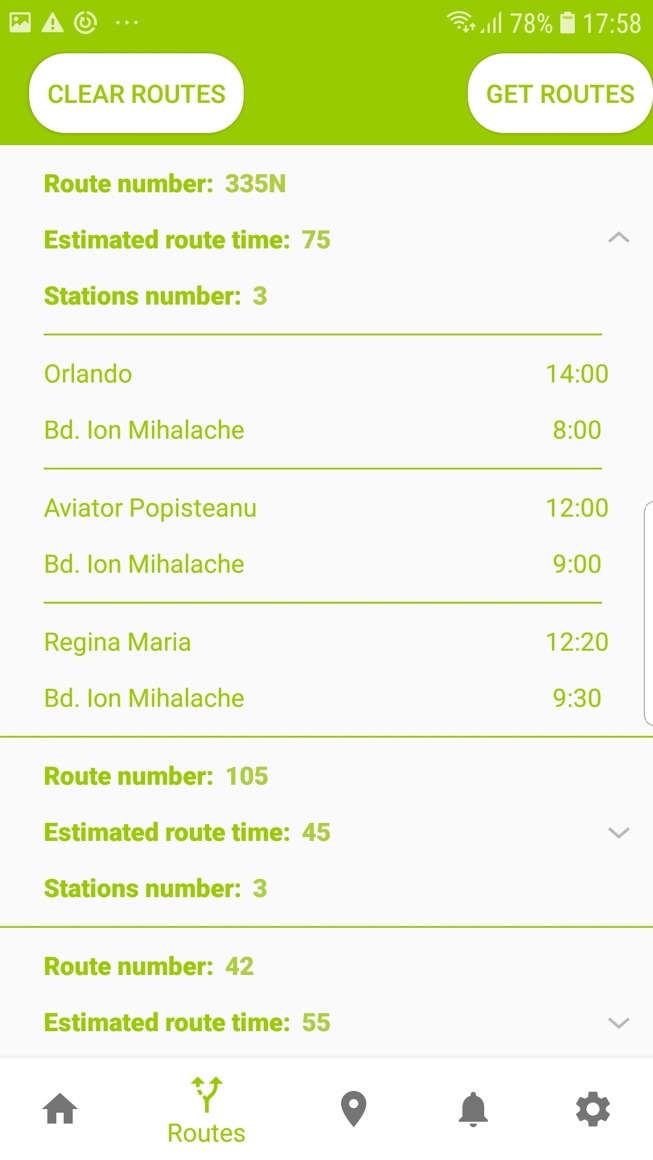


*Fig. 5 Ecran adaugare statii*

Atat rutele cat si statiile sunt salvate persistent folosind libraria Room.

Ecranul de afisare al rutelor (Fig. 6) are un ExpandableListView in care sunt afisate rutele cu posibiltatea de expandare a fiecarei celula din tabel pentru afisarea statiilor ficarei rute.

Butonul de clear routes din partea de sus va sterge toate rutele din baza de date locala Room.



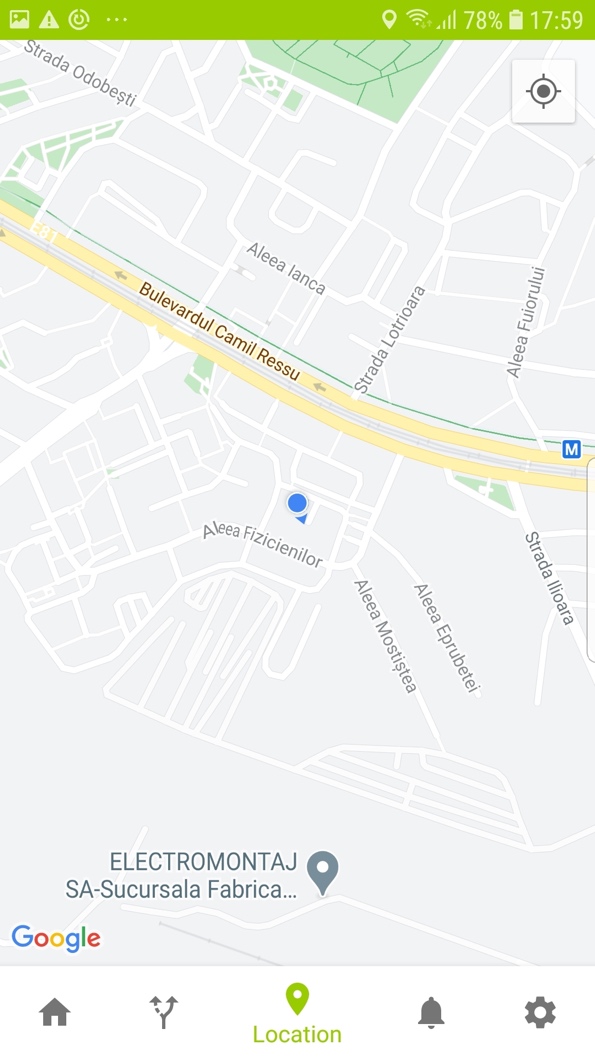
*Fig. 6 Ecran afisare rute*

Butonul Get Routes va prelua un JSON de la un API Mock (Fig. 7) <https://api.mocki.io/v1/c4474388/> , dupa ce este preluat este deserializat iar obiectele sunt salvate local in Room. Operatiile de preluare a datelor si salvare sunt realizate asincron.



*Fig. 7 JSON rute preluat din server*

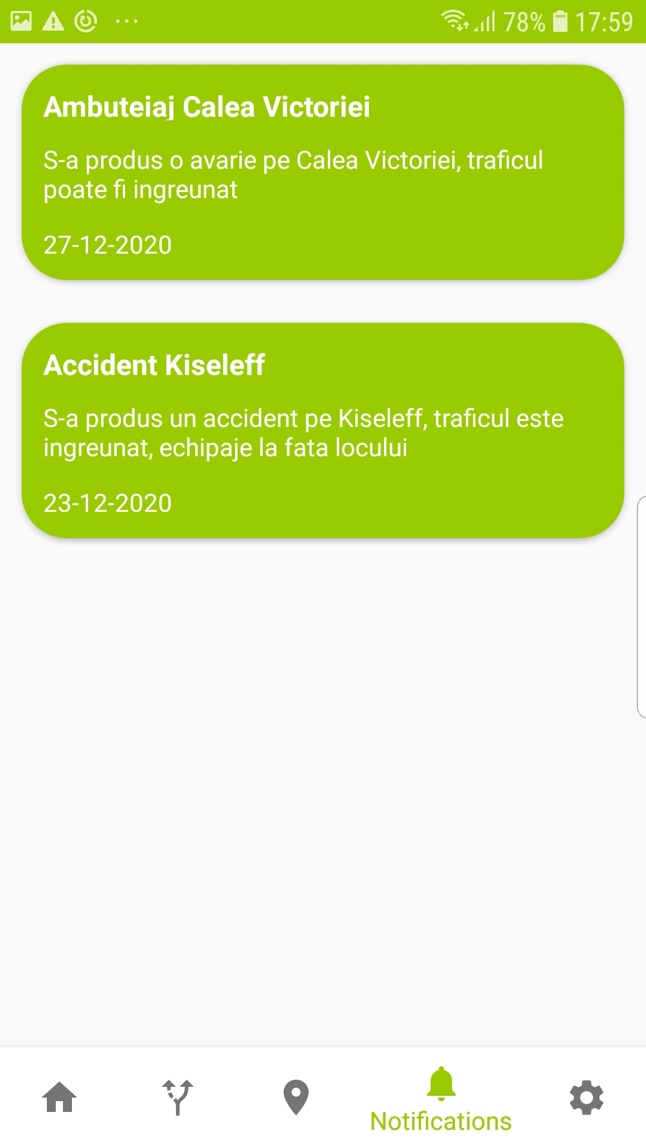
Fragmentul de locatie (Fig .8) ofera utilizatorului posibilitatea de a-si vedea locatia curenta.



*Fig. 8 Ecran locatie*

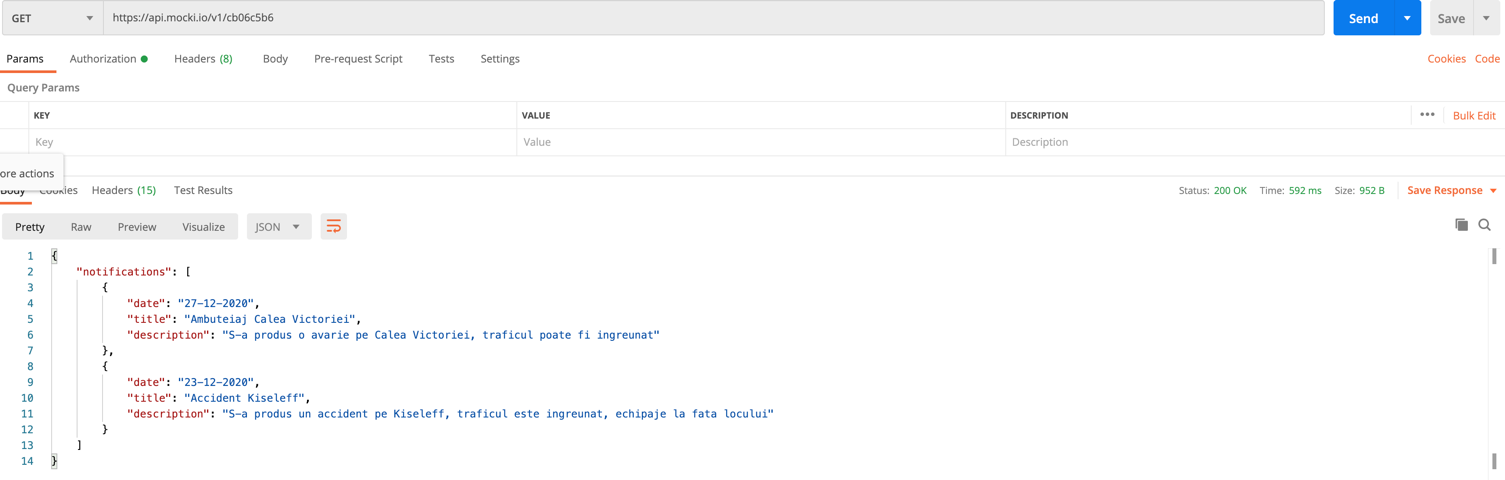
Fragmentul de notificari afiseaza evenimente importante din trafic care ar putea cauza intarzieri ale autobuzelor.

Datele sunt preluate dintr-un API Mock : <https://api.mocki.io/v1/cb06c5b6>.



*Fig. 9 Ecran notificari*

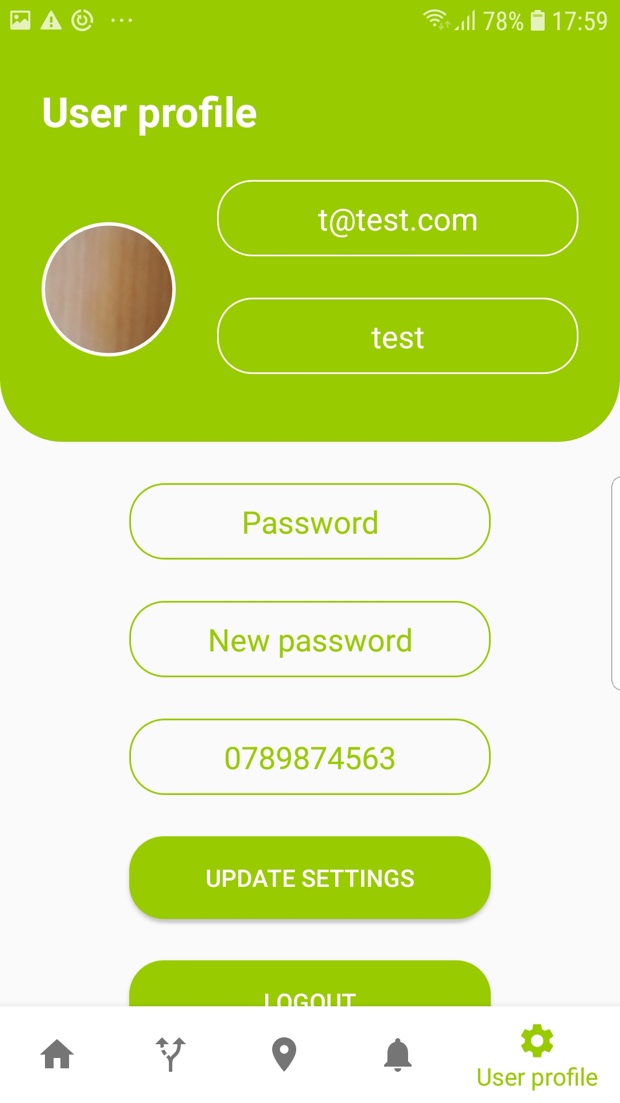
Operatia de preluare a JSON-ului (Fig. 10) din API este realizata asincron.



*Fig. 10 Model JSON notificari*

Ecranul de setari contine urmatoarele campuri:

1. email
2. username
3. parola
4. numar de telefon



*Fig. 11 Ecran setari*

Utilizatorul isi poate modifica profilul, poate incarca o poza de profil pe care o poate alege din galerie. Datele sunt preluate si salvate in Firebase.