**ACADEMIA DE STUDII ECONOMICE DIN BUCUREȘTI, FACULTATEA DE CIBERNETICĂ, STATISTICĂ ȘI**

**INFORMATICĂ ECONOMICĂ**

PROIECT DISPOZITIVE SI APLICATII MOBILE

Aplicație pentru persoanele care doresc să călătorească împărțind mașina (ridesharing)

**PROF. COORDONATOR: STUDENT:**

Asist. univ. **CIUREA CRISTIAN Tudor Ștefan Grupa 1085, Seria D**

CUPRINS

[PROIECT DISPOZITIVE SI APLICATII MOBILE 1](#_Toc29718771)

[CUPRINS 2](#_Toc29718772)

[INTRODUCERE 3](#_Toc29718773)

[DESCRIERE SOLUȚIE 4](#_Toc29718774)

[DESCRIERE TEHNICA A MODULUI DE IMPLEMENTARE 9](#_Toc29718775)

[CONCLUZII 11](#_Toc29718776)

[BIBLIOGRAFIE 12](#_Toc29718777)

[LISTA FIGURI 12](#_Toc29718778)

INTRODUCERE

Conceptul de ridesharing presupune utilizarea la comun a unui autoturism, și este de real folos persoanelor care nu posedă o mașină proprie, dar și șoferilor care călătoresc singuri și vor să socializeze, să îți facă prieteni, și de ce nu să își câștige banii pe combustibil. Folosirea la comun a unei mașini, în contextul supraaglomerării cu automobile și a poluării, care se resimt peste tot în lume. Este foarte important să găsim metode pentru a reduce aceste efecte negative pe care omul le are asupra Pamantului, iar reducerea numarului de autovehicule, împărțirea acestora, dar și adoptarea unor modele mai ecologice, reprezintă câteva variante viabile.

Pentru a încuraja conceptul de ride sharing, aplicația RideApp vine atât în ajutorul celor care caută să călătorească pe o anumită rută și să întălnească persoane cu care să meargă, cât și a șoferilor care doresc să își împartă mașina pe durata unei călătorii. Această platformă pentru dispozitivele mobile cu sistem de operare Android, are ca țintă un public specific, și anume persoanele care călătoresc, care fac naveta sau care vor un altfel de taxi.

DESCRIERE SOLUȚIE

Soluția informatică RideApp, prezintă o interfață simplistă, ce are în vedere o experiență comodă și rapidă de navigare a utilizatorilor.

Putem presupune următoarele scenarii:

- Ești student, locuiți la cămin, și dorești să pleci acasă în vacanță. Ești prea comod să iei trenul și preferi să plătești niște bani în plus pentru o mașină cu șofer care să te ducă direct acasă, scăpând de grija purtării bagajelor. Ce trebuie să faci? Deschizi RideApp și cauți să vezi dacă este cineva care face același traseu ca și tine. Cum? În primul rând este necesară crearea unui cont, din secțiunea de înregistrare. După ce te loghezi, din bara de navigație aflată în partea inferioară, accesând pagina “Gaseste”, vei putea să introduci adresa de plecare și cea de sosire, cu posibilitatea de a salva aceste locații pentru o căutare viitoare pe aceeași rută. După ce datele sunt introduse, se poate observa în lista de rezultate dacă există călătorii pe ruta selectată, iar dacă da, poți lăsa un mesaj prin care să stabilești detaliile cu șoferul și ceilalți participanți. Dacă nu, tot trenul este soluția.

- Ești șofer și vrei să pleci la drum, însă nu ai companie și ai mulți kilometri de parcurs până la destinație. În plus, ți-ar prinde bine niște bani cu care să plătești combustibilul. De ce să nu iei pe cineva cu tine? Pe RideApp, după ce îți creezi un cont, din pagina “Adaugă”, poți face călătoria ta publică și astfel oricine va putea să te găsească și să călătoriți împreună. După ce sunt introduse datele în formularul cu detaliile călatoriei (punct plecare, punct sosire, data, numar locuri, pret, daca este loc pemtru bagaj), aceasta va fi adăugată la “Călătoriile mele” și vei putea verifica periodic daca sunt persoane interesate, vizualizând mesajele atașate.

Descriere funcționalități:

1. Register – Introduce datele utilizatorului si creeaza un cont pe baza caruia vor fi publicate sau cautate calatoriile.

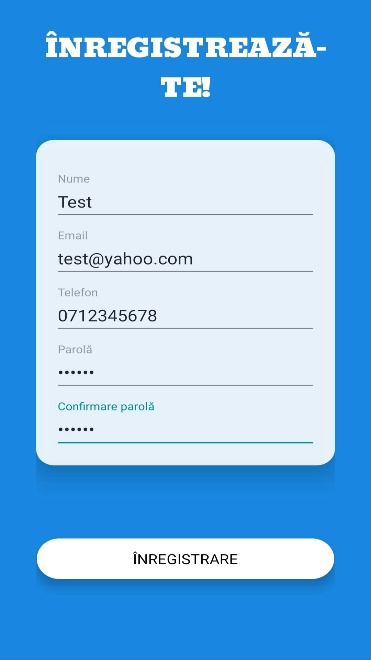


Figure 1 – Register

2. Login – pe baza mail-ului si a parolei, se face autenificarea in cont si se ofera acces la functionalitatile propriu-zise ale aplicatiei.

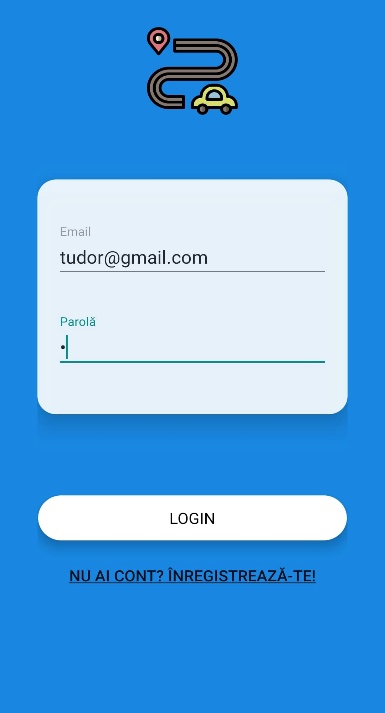


Figure 2 - Login

3. Acasa – pagina care afiseaza calatoriile adaugate de utilizatorul curent. Fiecare calatorie are in dreptul ei butoanele “vezi detalii” si “mesaje”, care ofera mai multe informatii despre calatorie.

In lista, afisanadu-se informatiile de baza, precum punct plecare – punct sosire, si data, daca se apaza butonul referitor la vizualizarea detaliilor, se va deschide o pagina noua unde sunt afisate complet informatiile.

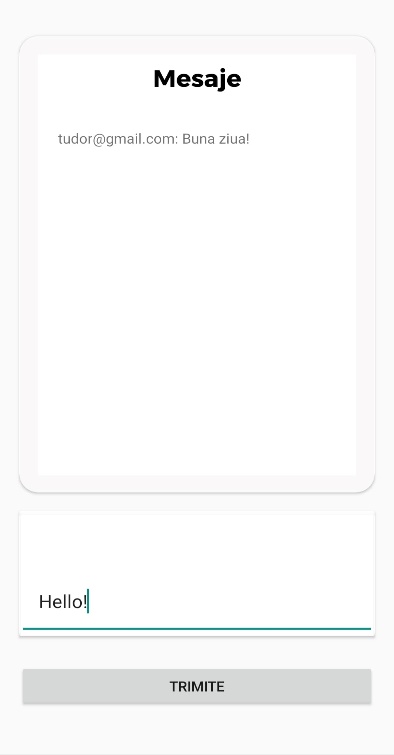
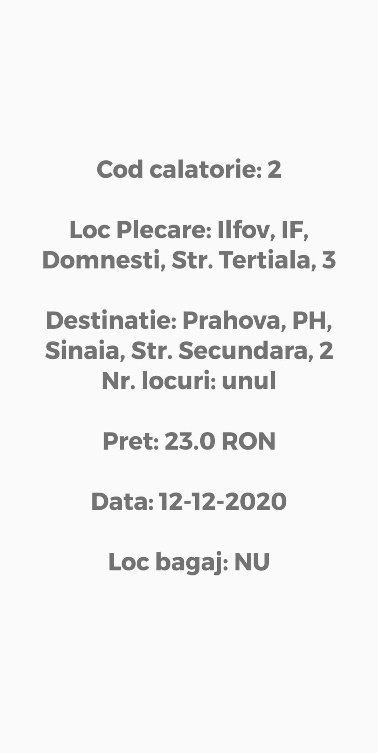


Figure 3 - Acasa

Figure 4 - Ride Details

Figure 5 - Mesaje

La apasarea butonului de mesagerie, se deschide o fereastra ce ofera posibilitatea transmiterii unor comentarii legate de acea calatorile. Comentariile sunt publice si oricine este interesat de acea calatorie poate pune intrebari intr-un mod foarte simplu si intuitiv.

4. Adauga – este pagina pe care utilizatorii o folosesc atunci cand ei sunt soferi si vor sa posteze o calatorie. Pentru asta, este necesar sa se introduca locul de plecare, destinatia, numarul de locuri disponibile, pretul/loc, data plecarii si daca exista posibilitatea depozitarii de bagaje.

Pentru a nu exista confuzii referitoare la formatul datei, s-a implementat un calendar simplu, care printr-o atingere selecteaza data dorita. De asemenea, la completarea campurilor care necesita introducerea unei locatii, de ofera sugestii de autocompletare, pentru a face selectia cat mai usoara pentru utilizator.

Dupa ce se adauga toate aceste detalii, datele vor aparea sub forma unei calatorii noi in lista ta de pe pagina de Acasa.

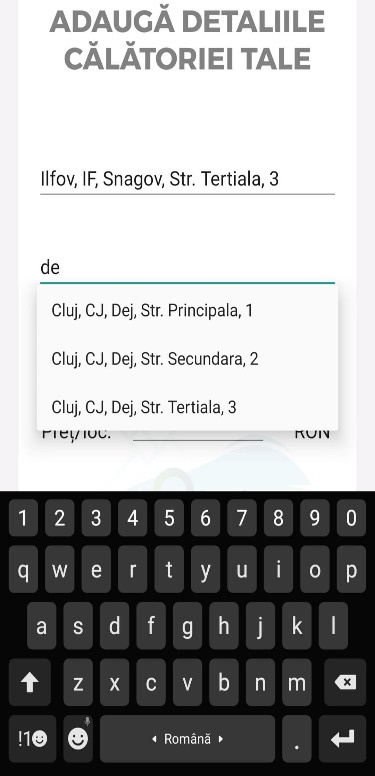


Figure 6 - Adauga (Autocompltare)

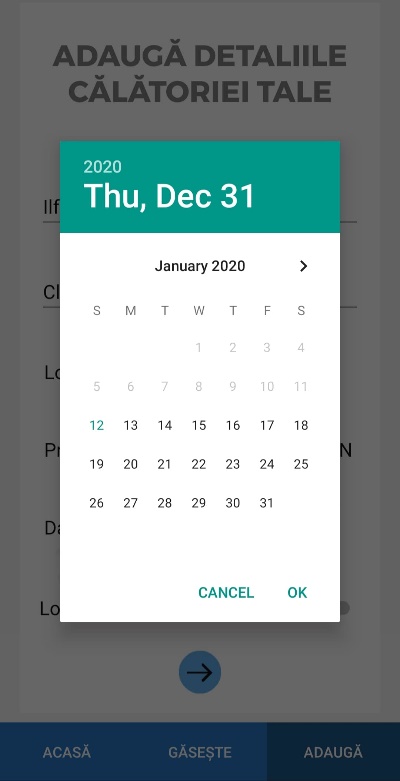
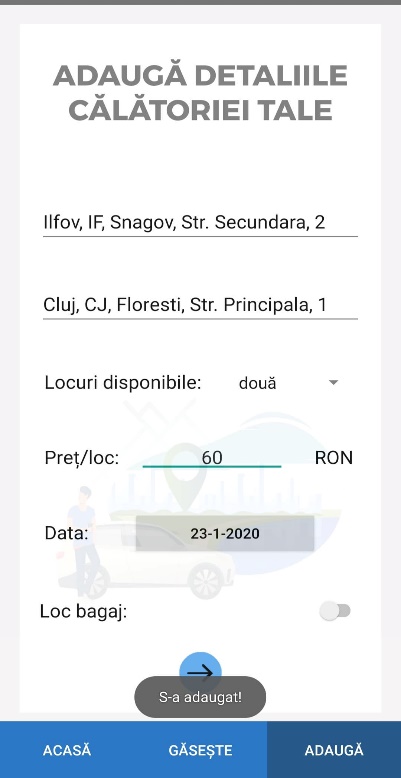


Figure 7 - Adauga

Figure 8 - Adauga (Calendar)

5. Gaseste – permite utilizatorului sa introduca punctul de plecare si punctul de sosire pentru gasirea unei calatorii. Campurile beneficiaza, de asemenea, de autocompletare.



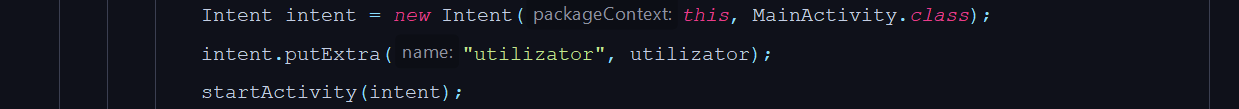
Figure 9 - Gaseste

Daca un utilizator frecventeaza des o ruta, prin bifarea optiunii din aceasta pagina, la fiecare reincarcare a aplicatiei, campurile vor fi autocompletate si astfel utilizatorul nu va mai trebui sa reintroduca de fiecare data adresa ce il intereseaza.

La fel ca si la pagina de acasa, cele doua butoane pentru detalii si mesaje sunt prezente, implementand aceleasi functionalitati.

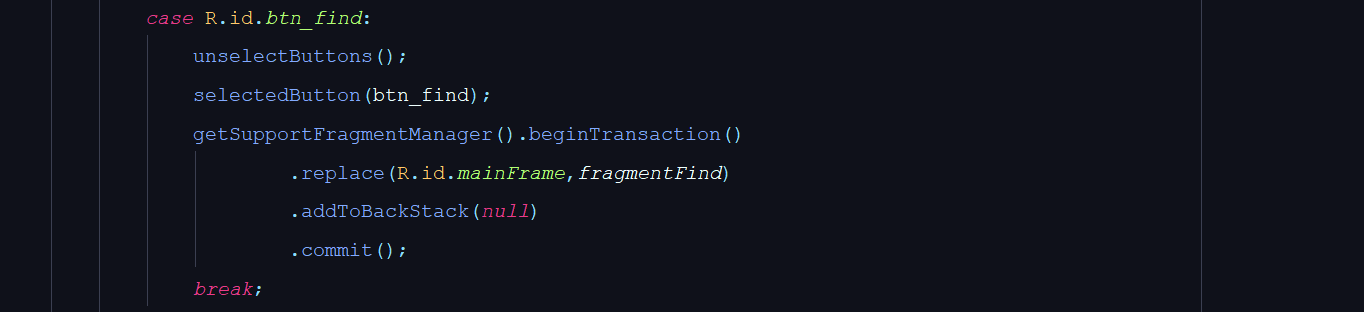
DESCRIERE TEHNICA A MODULUI DE IMPLEMENTARE

Atat pe pagina de Login cat si pe cea de Register, controalele vizuale (inclusiv butoanele) au fost puse pe panel-uri de tip CardView, pentru a stiliza aplicatia. Campurile de introducere a datelor sunt de tip TextInputEditText, iar datele sunt preluate din acestea si introduse in baza de date pentru inserare/verificare. Tabela care stocheaza utilizatorii este numita Users si contine campurile email(primary key), nume, telefon si parola. La logare, se interogheaza baza de date pentru a verifica daca mail-ul si parola introduse se regasesc in tabela Useri. Daca da, se transmite obiectul User prin Intent catre aplicatia principala MainActivity, si se ofera acces la functionalitatile propriu-zise.

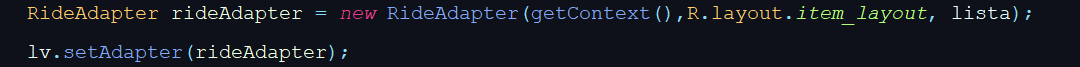


Elementul de baza pentru MainActivity este bottom bar-ul creat prin introducerea a trei butoane intr-un LinearLayout orizontal. Aceasta bara coordoneaza deschiderea si comutarea intre cele trei fragmente care sustin aproape toate functionalitatile aplicatiei (Acasa, Gaseste, Adauga).

Implementarea se face printr-o instructiune switch, care verifica id-ul butonului si in functie de butonul selectat, se afiseaza intr-un FrameLayout fragmentul indicat.



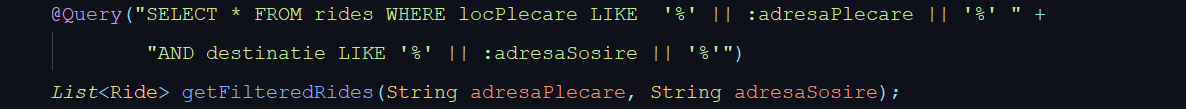
Butonul pentru pagina Acasa, deschide o instanta a FragmentHome, care contine intr-un LinearLyout, doua controale: TextView si CardView. In TextView este afisat mesajul de intampinare, personalizat pentru numele utilizatorului. CardView contine un ListView care vine cu un adaptor personalizat pentru fiecare item (RideAdapor), incadrat intr-un layout simplist: doua TextView-uri (pentru loc pornire – loc sosire, si pentru data) si doua butoane (“Vezi detalii” si “Mesaje”).



Obiectul Ride corespunzator unui item din ListView este trimis prin Intent la apasarea fiecaruia dintre cele doua butoane. Cand se apasa “Vezi detalii”, se deschide o activitate noua care preia obiectul Ride si il afiseaza intr-un TextView, cu toate atributele sale.

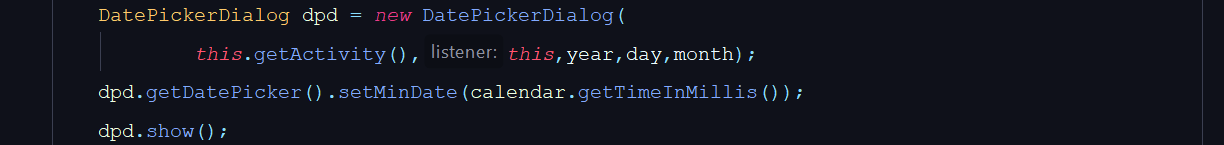
Butonul de “Mesaje”, deschide de asemenea o activitate noua, care implementeaza panoul de mesaje si EditText-ul de lasare a unui mesaj (feedback). La pornirea activitatii, se face request catre baza de date pentru a primi mesajele din tabela Feedbacks, corespunzatoare obiectului de tip Ride transmis prin Intent. Se afiseaza lista de mesaje intr-un ListView, acestea avand concatenat si mail-ul utilizatorului pentru identificarea meajului. In partea de jos, folosind EditTextul si un buton de trimitere, se salveaza mesajul in baza de date, dupa care ListView-ul este actualizat.

Butonul pentru pagina Gaseste, deschide o instanta a FragmentFind, care printre altele, implementeaza o cerere HTTP pentru a prelua continutul unui fisier de tip JSON, responsabil pentru crearea unui adaptor ce ofera sugestii de locatii pentru AutoCompleteTextView-uril folosite. Dupa ce sunt introduse campurile respective, pentru a afisa rezultatele cautarii se interogheaza baza de date si se returneaza obiectele care au orasul si strada ca cele introduse precedent.



Ultimul buton, cel de Adauga, deschide un fragment de timp FragmentAdd, care contine formularul de introducere a datelor, folosind mai multe controale: AutoCompleteTextView, Spinner, DatePickerDialog, Switch si Button. De asemenea, si acest fragment utilizeaza JSON-ul folosit in pagina precedenta, iar destinatia si locul de sosire sunt sugerate de catre control.

Butonul asociat campului “Data”, deschide o ferastra ce contine DatePickerDialog, care nu lasa utilizatorul sa selecteze o data aflata in trecut.



Obiectul creat este introdus in baza de date si astfel este populata tabela Rides si implicit, ListView-ul ce contine calatoriile utilizatorului din sesiunea curenta.

Avantajul oferit de aceasta implementare este dat de usurinta cu care se poate utiliza si obtine informatia de care ai nevoie, platforma putand fi utilizata de persoane de toate varstele.

CONCLUZII

In opinia mea, avand in vedere ca piata telefoanelor mobile include peste 2 milioane de utilizatori ai unui sistem de operare destul de atractiv, iOS, pentru a putea oferi aplicatia unui segment cat mai mare de utilizatori, consider ca implementarea aceste solutii in Xamarin ar fi mai benefica si mai strategica.

BIBLIOGRAFIE

1. <https://android.jlelse.eu/android-architecture-components-room-relationships-bf473510c14a> (documentatie pentru utlizarea Foreign Key in ROOM)
2. <https://android.jlelse.eu/android-architecture-components-room-relationships-bf473510c14a> (documentatie query in ROOM)
3. Seminare Asist. univ. CIUREA CRISTIAN
4. <https://github.com/alexandrudita/dam/wiki/> (documentatie generala)
5. <https://stackoverflow.com/questions/25545449/datepickerdialog-setmaxdate-show-one-month-and-one-day-additional> (documentatie DatePickerDialog)

LISTA FIGURI

[Figure 1 – Register 6](#_Toc29718389)

[Figure 2 - Login 6](#_Toc29718390)

[Figure 3 - Acasa 7](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718391)

[Figure 4 - Ride Details 7](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718392)

[Figure 5 - Mesaje 7](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718393)

[Figure 6 - Adauga (Autocompltare) 8](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718394)

[Figure 7 - Adauga 8](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718395)

[Figure 8 - Adauga (Calendar) 8](file:///C:\Users\tudor\Desktop\Documentatie_Rideapp.docx#_Toc29718396)

[Figure 9 - Gaseste 8](#_Toc29718397)