**Documentatia proiectului**

*Help with homework*

Studenti

Leca Marian-Rafael (10LF293)

Birtea Alexandru (10LF291)

1. **Prezentarea proiectului**

Proiectul “Help with homework” ofera o platforma unde studentii / elevii pot posta teme cu care nu se descurca, pentru a primi sugestii de rezolvare din partea altor utilizatori.

Acesta este, in esenta, un forum de discutii particularizat.

Prezinta toate functionalitatile ce tin de autentificare (login, sign up si password recover).

Subiectele de discutie si comentariile adaugate de catre utilizatori sunt retinute in Realtime Database, ce tine de API-ul Firebase.

Owner-ul unui topic are drepturi de editare (stergere comentarii, marcarea celui mai bun raspuns, modificare titlu / descriere). Pentru a il face pe utilizator sa simta ca modul de functionare al forumului este unul familiar, am adaugat posibilitatea de a marca un raspuns drept solutie (similar cu StackOverflow).

1. **Tehnologiile folosite**

Proiectul foloseste urmatoarele tehnologii:

* Realtime Database (Firebase) -> permite stocarea datelor in cloud.
* Authentication (Firebase) -> cu ajutorul acestei tehnologii realizam autentificarea, inregistrarea si resetarea parolei utilizatorului.
* Java -> este limbajul de programare in care am scris intreg codul.
* Android Studio IDE -> mediul in care a fost dezvoltata aplicatia

1. **Partea de backend (request-uri)**

In cadrul proiectului, am implementat un singur HTTP request.

Am folosit urmatorul fake API:

[Click pentru API](https://mocki.io/v1/c3fd5bf7-6725-4ec7-a156-e34b00f433a6)

La adresa respectiva se afla 3 comentarii pe care le consideram default, ce se vor incarca mereu inaintea celor din Firebase (adormim thread-ul pentru o secunda si jumatate pentru a realiza acest lucru).

Este de mentionat faptul ca aceste comentarii nu pot fi marcate drept solutie (insa pot fi sterse, ca cele retinute in realtime database).

La stergerea acestor comentarii, este afisat un toast cu un mesaj distinct.

In cazul in care dintr-un motiv anume nu se poate completa cererea, este afisat un mesaj de eroare corespunzator intr-un toast.

1. **Arhitectura**

Aplicatia are patru activitati :

* AuthenticationActivity
* DashboardActivity
* TopicsActivity
* ManagementActivity

Are, de asemenea, si opt fragmente:

* LoginFragment
* SignUpFragment
* ResetPasswordFragment
* DashboardFragment
* SeeTopicsFragment
* ManagementFragment
* AddTopicFragment
* SelectedTopicFragment

Aplicatia incepe cu activitatea AuthenticationActivity, careia i se ataseaza fragmentul LoginFragment. In acest punct, utilizatorul are urmatoarele posibilitati : de a se loga, de a alege sa deschida fragmentul de inregistrare sau pe cel de resetare a parolei (sunt doua textview-uri).

Dupa logarea utilizatorului, se deschide activitatea DashboardActivity si se adauga fragmentul DashboardActivity. Acesta este meniul principal al aplicatiei. Contine urmatoarele:

* 3 card view-uri : unul pentru logout, unul pentru activitatea de topic-uri (toate) si unul pentru topic-urile personale. Acestea au efect ripple la click
* Un textview ce saluta utilizatorul (adresa de e-mail se transmite folosind SharedPreferences)

Daca este ales card-ul pentru toate topic-urile, va fi deschis fragmentul SeeTopicsFragment, unde, intr-un RecyclerView, sunt afisate toate subiectele de discutie din Realtime Database-ul (Firebase) asociat aplicatiei si un buton pentru adaugare de topic, ce deschide fragmentul AddTopicFragment, care contine doua edit text-uri si un buton. Nu se poate adauga un topic cu titlu sau descriere nula sau cu titlu identic cu al unui subiect deja existent (postat de utilizatorul in cauza). In schimb, daca alege un topic, va aparea fragmentul SelectedTopicFragment, ce contine toate informatiile despre topic-ul ales in partea de sus, si mai jos, comentariile (primele default, urmate de cele din baza de date).

Daca este ales cel pentru subiectele de discutie personale, va fi deschisa activitatea ManagementActivity, careia i se va atasa fragmentul ManagementFragment. Aceasta activitate prezinta un RecyclerView ce contine doar topic-urile postate de catre utilizator si un DrawerLayout, unde se afla: adresa de e-mail a utilizatorului logat, butoane pentru adaugare / stergere / modificare topic-uri. Pentru stergere trebuie ales cel putin un topic (un subiect de discutie selectat va avea culoarea verde, iar unul neselectat culoarea gri), iar pentru modificare unul singur (utilizatorul va fi redirectionat catre fragmentul SelectedTopicFragment, pe care l-am descris mai sus).

Contine, de asemenea, si o alarma ce suna din 60 in 60 de secunde cate 2 secunde de fiecare data.

1. **Cum poate fi utilizata aplicatia?**

Aplicatia ii permite utilizatorului sa isi creeze cont, sa se autentifice sau sa isi reseteze parola. Aceasta functionalitate prevede si niste validari pentru e-mail si parola atat pentru toate cele 3 (sign up, login si resetare -> doar e-mail). Se incurajeaza folosirea textview-urilor din cadrul fragmentelor ce tin de activitatea de autentificare (login, sign up, reset password) deoarece apasarea butonului de back va duce la inchiderea aplicatiei in acest caz.

Utilizatorul se poate deloga fie apasand butonul back din activitatea de dashboard, fie apasand butonul card numit ce are label-ul Logout.

Utilizatorul poate adauga subiecte de discutie atat din fragmentul destinat tuturor topic-urilor (exista un ImageView in partea de jos a fragmentului in acest sens), cat si din cel de management, insa pentru a sterge / modifica topic-urile personale, este nevoit sa acceseze partea de management a aplicatiei.

Aplicatia suporta atat portrait mode cat si landscape mode pentru toate activitatile.

Nu exista mai multe tipuri de utilizatori, insa am implementat conceptul de ownership in felul urmator:

Daca utilizatorul logat deschide un subiect de discutie pornit de catre el insusi, capata drepturi de editare, cum ar fi:

* Poate actualiza titlul si descrierea topic-ului facand click pe acestea (i se semnaleaza faptul ca are acest drept prin prezenta butonului de update in dreapta item-ului de topic din fragmentul SelectedTopicFragment)
* Poate sterge orice comentariu (default / prezent in realtime database).
* Poate marca un comentariu ca fiind solutie (doar unul poate fi marcat la un moment dat). In acest caz, va aparea o bifa in partea stanga a acestuia. Daca subiectul de discutie are deja o solutie si se marcheaza alt comentariu, cel de dinainte nu va mai fi solutie.

1. **Concluzii**

In concluzie, proiectul “Help with homework” reprezinta un loc unde cei ce intampina dificultati in rezolvarea temelor gasesc ajutor. Interfata aplicatiei este una user-friendly, amplasarea elementelor vizuale este intuitiva, lucruri ce fac ca lucrul cu aceasta sa fie usor.

Functionalitatile pe care le prezinta acopera toate situatiile pe care le pot intampina elevii / studenti ce aleg sa foloseasca aplicatia.

Scopul proiectului este de a facilita parcursul elevilor si al studentilor prin critici constructive si ajutor acordat in mod reciproc.