

# Documentul de proiectare arhitecturala

## IM Chat

### 1. Introducere

#### 1.1 Scopul sistemului

Acest document de proiectare arhitecturala furnizeaza o descriere completa a aplicatiei mobile IM Chat prin intermediul careia oamenii sa poata comunica de la distanta. Utilizatorii vor putea comunica point to point fara a se anunta inainte. Mesajele sunt transmise bidirectional intre utilizatori si pot fi vizualizate aproape instantaneu facand posibila realizarea unui dialog interactiv.

Datorita incarcaturii de date relativ mici din timpul schimbului de mesaje, IM Chat este o aplicatie ce poate fi rulata pe dispozitive mobile aflate la distante mari, functionand la viteze acceptabile.

#### 1.2 Definitii, acronime

Peer-to-Peer - comunicarea a doi useri care sunt online;

Server - conectează utilizatorii din rețea la o sursă centralizată;

BD - reprezintă o modalitate de stocare a unor informații și date pe un suport extern, cu posibilitatea extinderii ușoare și a regăsirii rapide a acestora;

IM Client = Un punct final al serviciului IM. IM Server = O entitate în rețea care oferă funcționalități de mesagerie în timp real.

IM session = Schimb de mesaje aproape în timp real în care expeditorii și receptorii se unesc pentru o perioadă de timp (sesiune). Sesiunea este stabilită la un moment dat, continuă pe o durată finită și apoi este dizolvată. Mesajele schimbate sunt asociate împreună în contextul acestei sesiuni.

IM User = Un om care folosește caracteristicile descrise printr-un dispozitiv terminal. În acest document, termenii "User" și „IM User” sunt echivalenți.

Instant Messaging Service = O aplicație de sistem prin care un client este capabil să furnizeze (aproape) în timp real mesaje.

Status = utilizator Online sau Offlinere

User Menu = O bază de date locală în terminal conținând utilizatorii.

### **1.3 Documente referinte**

Documentul prezent, de proiectare arhitecturala este realizat cu scopul de a prezenta utilizatorului detaliile de implementare a aplicației conform cerintelor acestuia. Cerintele utilizatorului sunt evidentiuate și descrise pe larg în documentul intitulat “Specificarea și analiza cerintelor”.

## **2. Obiective de proiectare**

Aplicația IM Chat are ca scop facilitarea procesului de comunicare textuală între doi utilizatori de dispozitive mobile Android prin intermediul unei conexiuni la internet. Pe parcursul dezvoltării aplicației vom urmări utilizarea unor tehnici specifice, astfel încât aplicația finală să poată fi utilizată în condiții reale și consumul de resurse să fie redus.

Programul va fi capabil să satisfacă principalele cerințe funcționale: adăugare cont, autentificare, chat, sau modificare profil. În vederea îndeplinirii cerințelor nefuncționale specificate în “Documentul de specificare a cerințelor”, acest document de proiectare a arhitecturii sistemului prezintă în continuare criteriile de performanță pe baza cărora va fi creată aplicația finală.

### **2.1.Criterii de performanță**

Aplicația IM Chat va fi un client care va comunica cu serverul la crearea contului, autentificare sau comunicare cu contacte offline iar viteza schimbului de mesaje client-server sau client-client depinde în cea mai mare măsură de internet. Performanțele aplicației vor depinde de asemenea și de baza de date la care este legat serverul de mesagerie întrucât serverul se va menține în permanență o conexiune cu aceasta. În

ceea se priveste softwareul aplicatiei server, acesta va fi proiectat utilizand fire de executie pentru fiecare cerere din partea clientilor, paralelizand executia. Serverul va fi capabil astfel sa primeasca si sa raspunda simultan la aproximativ o mie de astfel de cereri ce pot fi cereri de autentificare, creare cont, sau cereri de comunicare cu un alt client. Pentru un sistem distribuit de servere numarul de cereri la care se va putea raspunde simultan va creste proportional cu numarul de statii din acest sistem. Timpul de raspuns estimat pentru o cerere trimisa de client serverului va fi intre 300 ms si 2 s in functie de retea si timpul de raspuns al bazei de date, si intre 200 ms 300 ms pentru comunicarea peer-to-peer intre utilizatori. Memoria RAM utilizata de program va fi intre 100 KB 500 KB.

## **2.2.Criterii de incredere**

*Robustetea.* Programul va fi prevazut cu mecanisme de atentionare a utilizatorului in cazul in care datele introduse de acesta sunt invalide, iar de cele mai multe ori aceste mecanisme presupun afisarea de mesaje si blocarea accesului la ferestrele aplicatiei. Cele mai frecvente cazuri prevazute sunt: introducere email inexistent, parola incorecta sau detalii invalide in campurile pentru completarea profilului.

*Fiabilitatea.* Atat timp cat conexiunea la internet este asigurata programul va garanta buna functionare intrucat serverul de mesagerie va folosi o baza de date sigura si comunicatia dintre server si client.

*Toleranta la defecte.* Erorile cu care s-ar putea confrunta programul ar putea fi generate de retea sau de baza de date. In cazul unei erori de conexiune, mesajele aflate in curs de trimitere catre un alt client vor fi salvate intrun bufer local pe statia clientului, iar aplicatia va deconecta automat clientul. In cazul unei erori generate de baza de date, informatiile vor fi salvate intrun buffer local pe server, iar clientii ce se autentifica, isi creeaza cont sau initiaza o noua conversatie vor fi notificati. Nu sunt prevazute erori cauzate de softwareul aplicatiei.

*Securitatea.* Aplicatia va garanta securitatea mesajelor trimise apeland la o baza de date sigura in care vor fi stocate datele referitoare la conturile utilizatorilor.

*Siguranta.* Impedimentele intampinate ar putea fi cauzate de retea sau de intreruperi hardware atat de partea serverului cat si de partea clientului (intreruperea conexiunii la internet, intreruperea alimentatiei cu energie electrica). In timpul trimiterii in internet a pachetelor intre client si server sau intre clienti pot fi omise pachete sau poate fi nerespectata ordinea. Acest aspect este rezolvat de protocolul TCP din nivelul transport al stivei OSI utilizat de aplicatie, acesta reluand trimiterea pachetelor pana cand acestea respecta ordinea si isi pastreaza integritatea.

### **3.1 Prezentare generala a arhitecturii sistemului**

Arhitectura sistemului este reprezentata de modelul Client-Server.

Subsistemul numit server furnizeaza servicii instantelor altor subsisteme, numite clienti. Clientii cunosc interfata serverului dar serverul nu trebuie sa cunoasca interfetele clientilor.

Fluxul controlului clientului este independent de fluxul controlului server-ului, exceptand sincronizarea pentru a raspunde la cereri si a primi rezultate.

O alta arhitectura a sistemului in vederea comunicarii a doi useri care sunt online este dupa modelul UNU la UNU. Arhitectura UNU la UNU este o generalizare a arhitecturii client--server, in care subsistemele pot juca atat rolul de server cat si rolul de client. Fluxul controlului in fiecare subsistem este independent, exceptand sincronizarile necesare pentru tratarea cererilor.

Firebase este platforma mobilă Google pentru dezvoltarea rapidă a aplicațiilor.

Firebase oferă o suită de instrumente necesare aplicatiei de tip chat cum ar fi:

- Cloud Firestore

Este cea mai nouă bază de date Firebase pentru dezvoltarea aplicațiilor mobile. Se bazează pe succesele bazei de date în timp real cu un nou model de date mai

intuitiv. Cloud Firestore oferă, de asemenea, interogări și scale mai bogate, mai rapide decât baza de date în timp real.

#### -Realtime Database

Este o soluție eficientă, cu latență redusă pentru aplicațiile mobile care necesită stări sincronizate între clienți în timp real.

Firebase este dotat cu funcții care îi ajută pe dezvoltatori ca noi să se concentreze pe crearea de aplicații, în timp ce gestionează toată logica de pe server. Lucruri ca:

Authentication: Aceasta include suport pentru autentificarea prin e-mail și parolă, precum și capabilități de conectare simplă (prin Facebook, Twitter și Google).

Cloud functions: Acestea rulează o logică suplimentară de la server.

Realtime database: Aceasta este o bază de date „NoSQL” care se actualizează în timp real.

Static hosting: Acesta este un mijloc de servire a activelor pre-construite în loc de redare în timpul rulării.

Cloud storage: Ne oferă un loc pentru stocarea materialelor media.

Firebase oferă gratuit autentificare și acces la baza lor de date în timp real. Baza de date în timp real permite până la 100 de conexiuni simultane și 1 gigabyte de stocare pe lună.

## 3.2 Decompozitia in subsisteme si responsabilitatile fiecarui subsistem

Subsisteme care alcatuiesc aplicatia:

->Client;

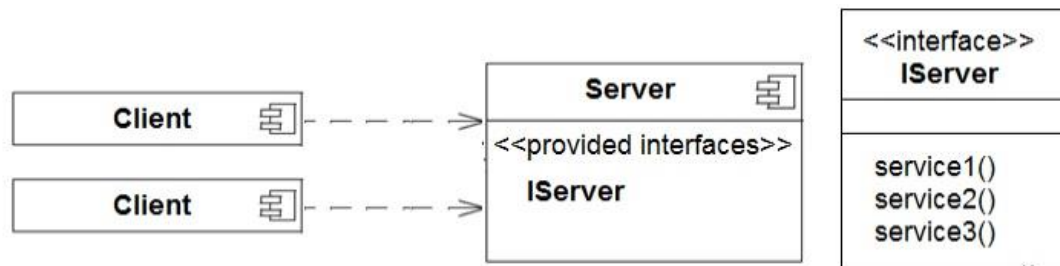
->Server;

->Baza de date;

Relatia de dependență dintre sisteme:

Clientii apeleaza servicii ale serverului. Deasemnea, clientii au si rol de server in contextul comunicarii intre doi utilizatori care sunt online.

Serverul raspunde la cererile clientului efectuand tranzactii cu baza de date. Acesta autentifica utilizatorii, adauga utilizatorii noi in baza de date, conecteaza doi utilizatori care sunt online pentru a comunica direct sau stocheaza mesajele care sunt adresate unui utilizator offline pentru a i le livra in momentul in care acesta este online.



Baza de date stocheaza datele si asigura si securitatea lor.

Diagrama de componente:

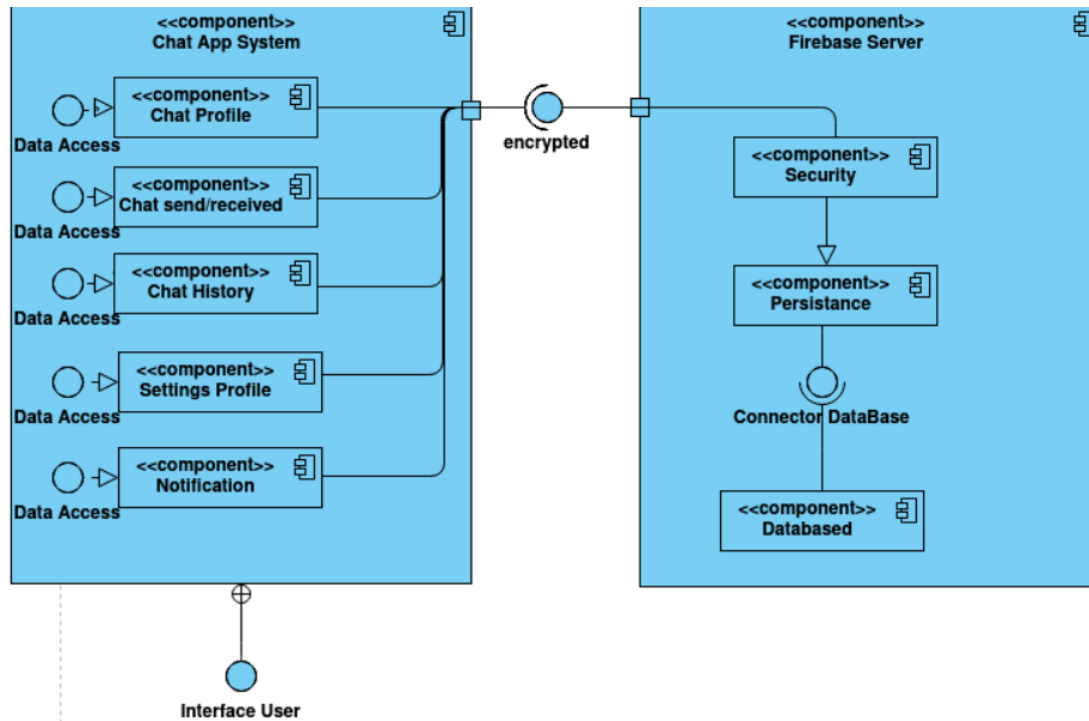
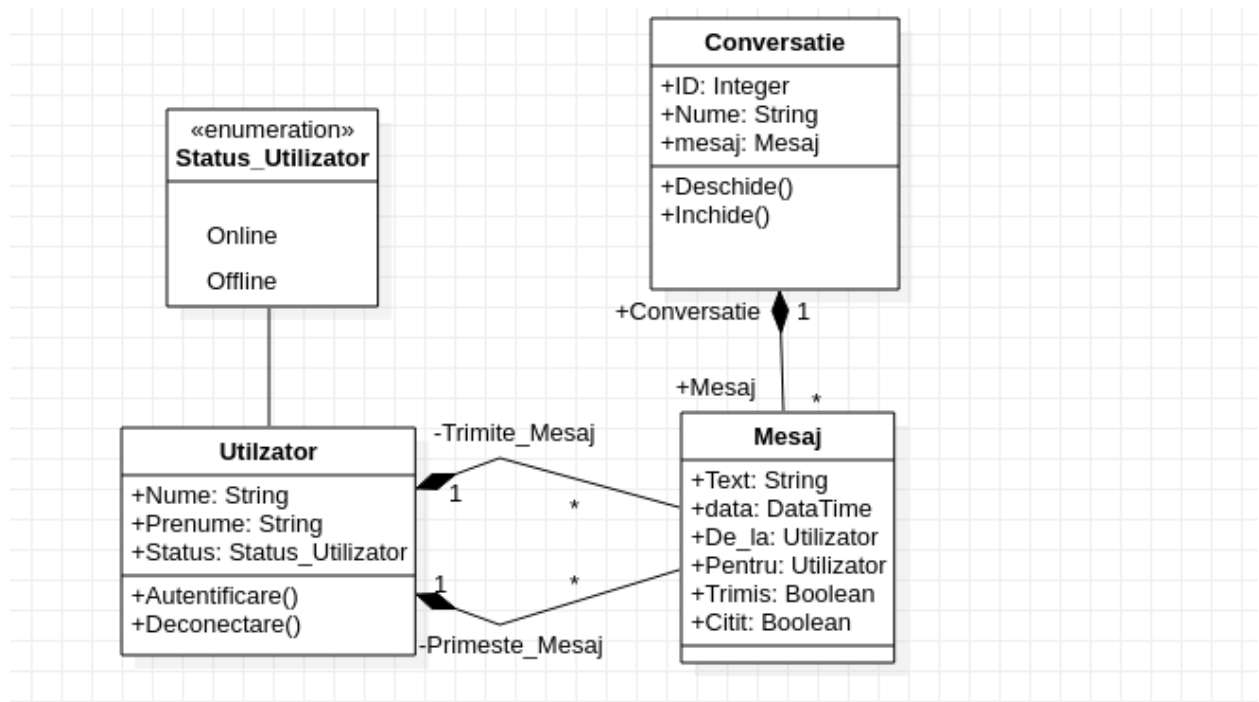


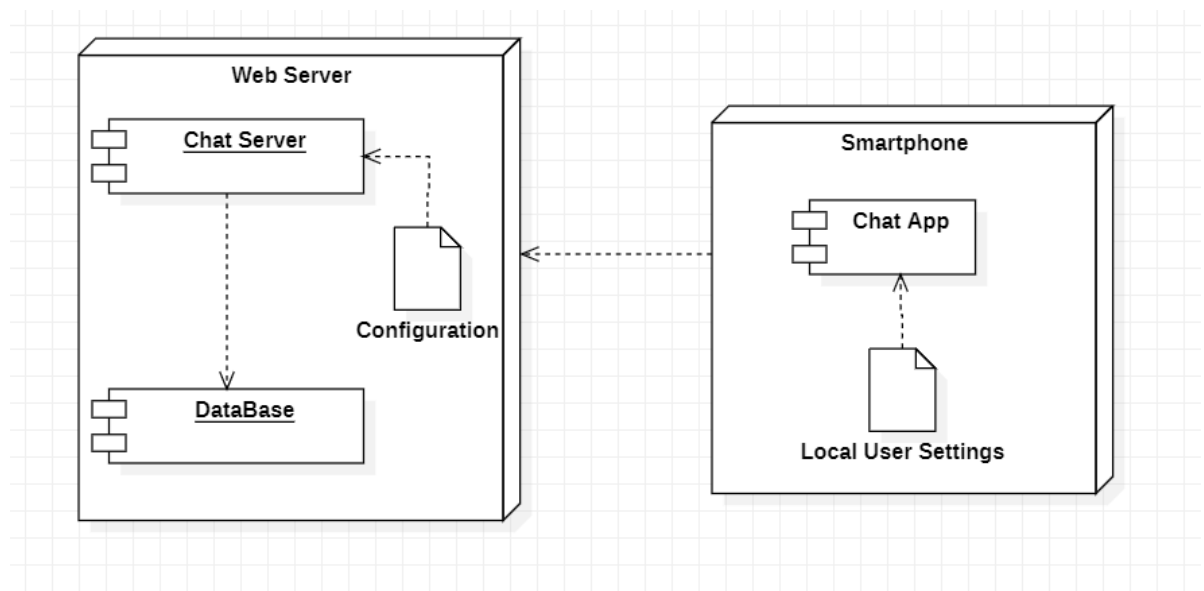
Diagrama de clase



### 3.3 Distributia subsistemelor pe platforme hardware/software

Acest sistem este conceput cu scopul de a oferi niste functionalitati minimale cu cerinte de sistem minimale. In consecinta distribuirea sistemului pe platforme nu necesita un hardware foarte performant, subsistemele solicitand o configuratie minima atat hardware cat si software ce presupun si o conexiune la internet ca urmare a faptului ca trebuie sa comunice intre ele. In vederea obtinerii unei performante cat mai ridicate si a unei complexitati a sistemului cat mai scazute, distribuirea pe platforme se face la inceputul proiectarii propriu-zise. Selectia hardware-ului se face in functie de sistemul de operare pe care va rula aplicatia, sistemul de gestiune a bazei de date si in functie de restrictionarile clientului cu privire la costuri.

#### Diagrama de distributie



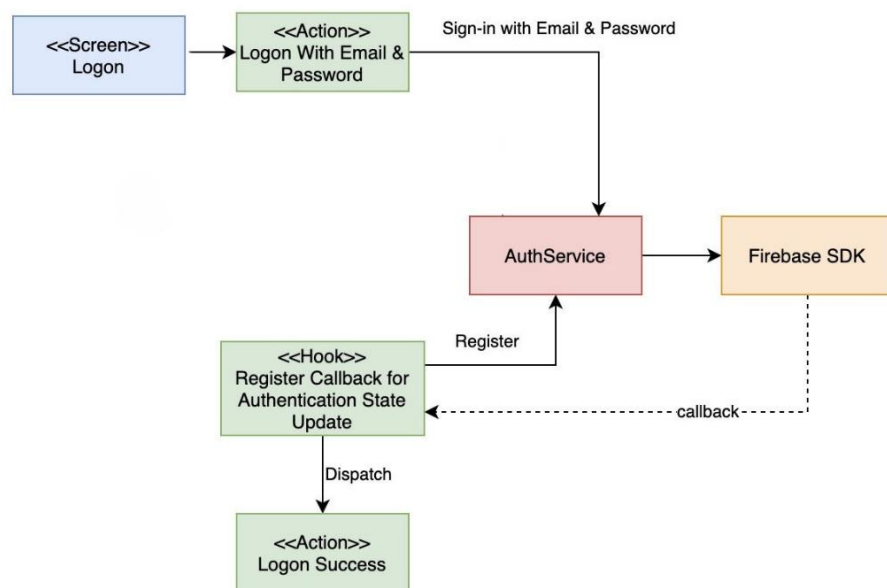


### 3.4 Managementul datelor persistente (fișiere, sistemul de baze de date folosit, schema conceptuala a bazei de date)

Datele persistente ale sistemului sunt reprezentate de datele despre utilizatori și ale mașinii fizice de pe care se conectează și mesajele trimise către un utilizator offline. Aceste date sunt salvate într-o bază de date și pot fi regasite și actualizate în cursul mai multor execuții.

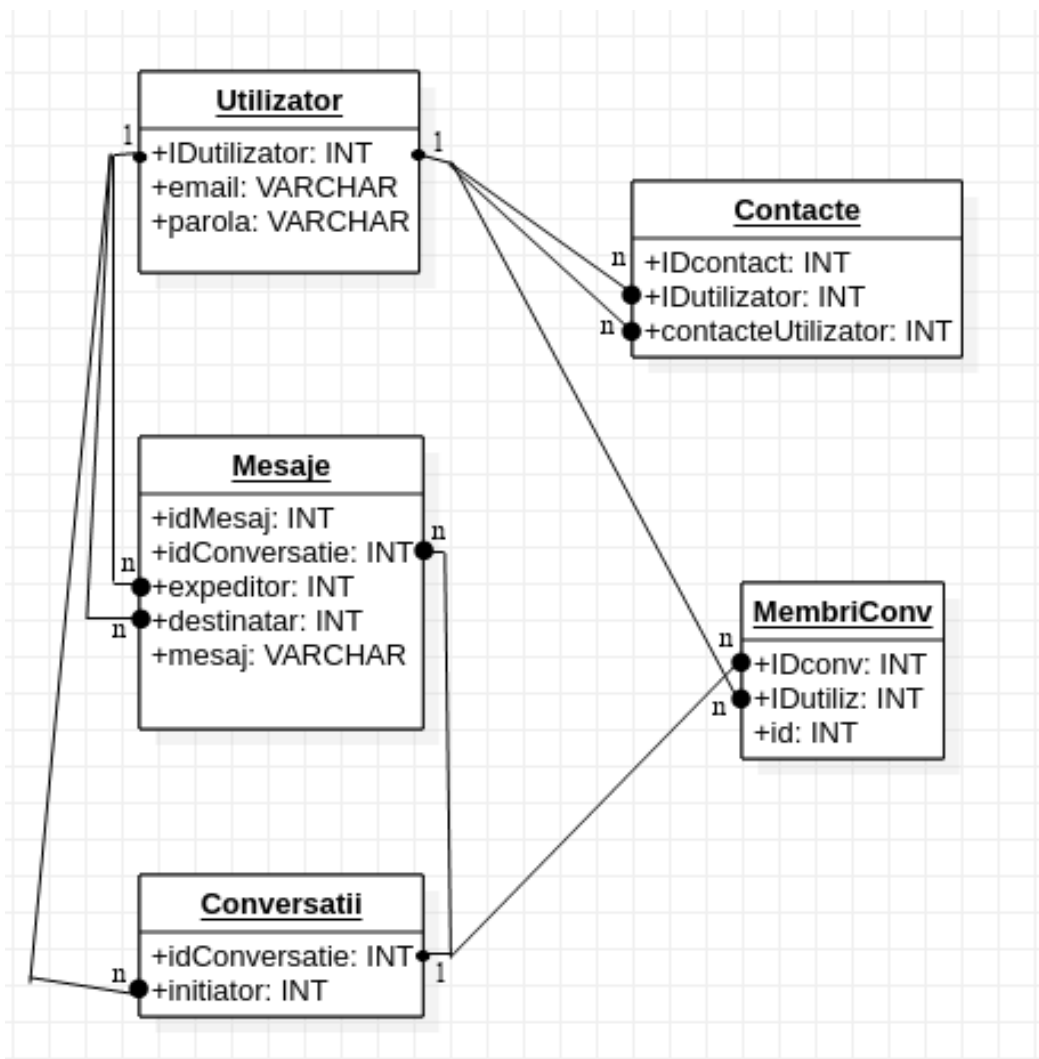
Pentru a crea baza de date necesară aplicației vom folosi platforma Firebase. Firebase este o platformă creată pentru a ajuta programatorii să dezvolte aplicații de înaltă calitate. Produsul inițial al companiei Firebase a fost o bază de date în timp real. Firebase Auth este un serviciu care permite autentificarea utilizatorilor scriind cod doar pe partea de client.

Firebase Authentication este o bibliotecă utilizată pentru a crea cont unui utilizator în aplicația IM Chat. Clasa FirebaseAuth oferită de această bibliotecă oferă metode pentru a crea un cont nou, folosind o adresă de email și o parolă, posibilitatea de a ne autentifica și de asemenea de a ne deloga.



Un obiect `FirebaseUser` reprezintă contul unui utilizator care s-a înregistrat în aplicație. Instanța obiectului `FirebaseUser` este independentă de instanța obiectului `FirebaseAuth`. Un utilizator `Firebase` are un set fix de proprietăți de bază: un identificator unic, o adresă de email, un nume, o fotografie URL stocate în baza de date a utilizatorilor `Firebase`.

## Diagrama bazei de date



### **3.5 Controlul accesului utilizatorilor la sistem**

Fiecare instanta din sistem are anumite drepturi de acces la resursele sistemului. Actorii sistemului sunt utilizatorul si administratorul.

Utilizatorii au dreptul de a crea un cont, de a se loga cu un nume de utilizator si o parola valide, de a cauta si adauga noi utilizatori in lista de prieteni, de a transmite mesaje utilizatorilor din lista de prieteni si de a modifica datele referitoare la profilul propriu.

Credentialele despre fiecare utilizator sunt criptate si stocate in baza de date fiind accesibile doar administratorului de sistem.

Administratorul are toate drepturile unui utilizator plus posibilitatea de a accesa, modifica sau sterge datele despre oricare utilizator din baza de date.

### **3.6 Fluxul global al controlului**

Server-ul contine o bucla principala in care se asteapta cereri de la utilizatori in vederea autentificarii sau conectarii. Atunci cand se produce un astfel de eveniment, el este prelucrat de server si pe baza informatiei asociate evenimentului, serverul inregistreaza un nou client in baza de date, il conecteaza cu un alt client pentru o comunicare One to One sau salveaza mesajele in baza de date daca acestea sunt adresate unui utilizator offline pana in momentul in care se primeste o cerere de autentificare de la utilizatorul care a fost offline si i se livreaza mesajele.

Subsistemul client contine o bucla principala in care asteapta cereri de la alti utilizatori in vederea comunicarii One to One. La inceputul buclei se verifica daca i-au fost adresate mesaje cat timp a fost offline asteptand o cerere de la server.

### **3.7 Condițiile limita (cazurile de utilizare limită)**

Condițiile limita semnifică acordul dintre producător și utilizator referitoare la confidențialitatea datelor personale ale utilizatorului. Acestea vor fi criptate, datele nefiind făcute publice pentru folosirea în alte scopuri.

## Planificarea implementarii proiectului

Nume	Asignat	Sprint	Estimare
Instalare Framework + Setari Android Studio Înregistrare utilizator Funcția Login/Logout Funcția Meniu Functia Display Users	Asimionesei Daniel Nicolaescu Antonia	1	10h
Send Message Display Message Message Status Functia User Profile Functia Add Photo Profile Notificari conversatii	Asimionesei Daniel Nicolaescu Antonia	2	10h
Istorice conversatii Prezență user în IM Chat IM one-to-one Cautare utilizatori Resetare date de conectare Afisarea ultimului mesaj din conversatie Numarul mesajelor necitite	Asimionesei Daniel Nicolaescu Antonia	3	10h