Documentație VirtualSoc

Năstasă Baraș Luca

Universitatea ”Alexandru Ioan Cuza”, Iași [lucanastasa@gmail.com](mailto:ioncelciuc2001@gmail.com) [luca.nastasa@info.uaic.ro](mailto:ionut.ciuc@info.uaic.ro)

**Abstract.** Acest document conține detalii despre implementarea și funcționalitatea proiectului VirtualSoc. În secțiunile următoare se găsesc detalii despre motivația alegerii proiectului, comenzile posibile și thenologiile utilizate.

**Keywords:** Social Media, Comunicare Server-Client, TCP/IP.

# Introducere

Aplicația VirtualSoc aparține cursului de Rețele de Calculatoarei. Proiectul reprezintă un prototip de aplicație Social Media ce permite utilizatorilor să se autentifice pe o platforma social media, să posteze știri și sa permita comunicarea prin mesaje text intre 2 persoane. Am ales acest proiect deoarece sunt foarte interesat de modul în care funcționează retelele de socializare.

# Tehnologii

Ca tehnologii utilizate, aplicația se folosește de protocolul TCP, deoarece e crucial ca mesajele transmise să ajungă intacte și în ordine. Astfel, se prioretizează corectitudinea pachetelor trimise în detrimentul memoriei utilizate și al timpului de execuție (cele din urmă două aspecte fiind mult mai eficiente la protocolul UDP).

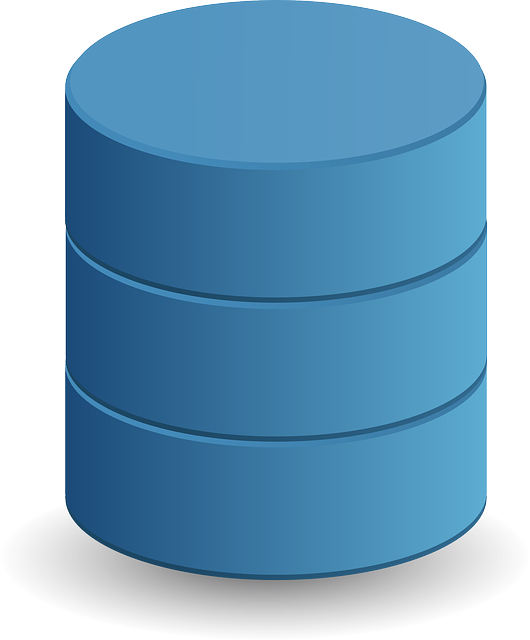
În alte cuvinte, se prefera tehnologia TCP față de UDP deoarece tehnologia UDP nu dispune de verificări de transmitere corectă și integrală a pachetelor, ceea ce ar putea duce la mesaje incorect transmise sau pierdute. Tehnologia TCP vine cu avantaje de a trimite integral și în ordinea dorită pachetele, făcându-se o verificare de primire a acestora între cel care trimite și cel care primește un pachet. În ciuda faptului că aceste verificări pot avea un impact asupra timpului de execuție, este de preferat așteptarea unui mesaj corect și complet.

Pentru stocarea datelor se folosește SQLiteii, o baza de date care intruneste proprietatile ACID (Atomizare,Consistenta,Izolare,Durabilitate) și este foarte ușor de utilizat pentru programatori.

# Arhitectura

## Diagrama de utilizare

Aplicația e formată dintr-o componentă client, și mai multe componente server ce reprezintă microservicii.



Client

Server

## 

## Legendă:

Request server:

Response server:

Request DB:

Response DB:

**Fig. 1.** Modul în care funcționează aplicația

## Structuri de date

Pentru utilizarea aplicatiei se folosesc urmatoarele modele:

* Client

Acestea sunt poprietatiile care vor fi folosite pentru client:

1. id în care retinem id-ul fiecarui utilizator(este unic și va fi folosit pentru identificare).

2. isAdmin în care retinem o valore ce ne spune dacă utilizatorul nostru are statusul de admin.

3. userName în care retinem numele de utilizator.

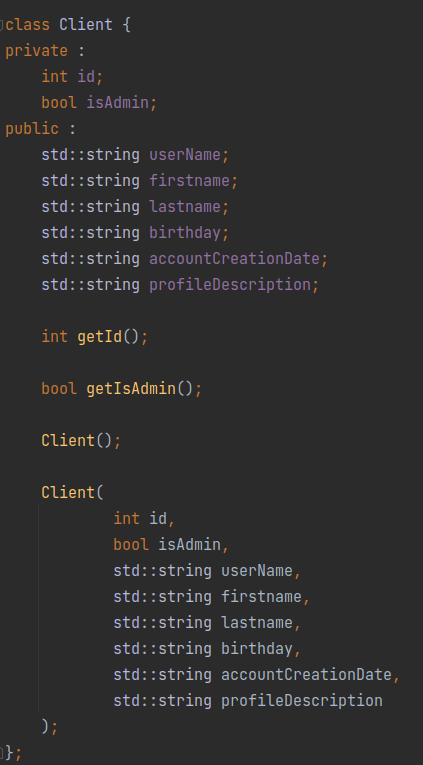
4. firstname în care retinem numele de familie.

5. lastname în care retinem prenumele.

6. bithday în care retinem data nasterii.

7.accountCreationDate în care retinem data în care a fost creeat profilul.

8.profilDescription în care retinem o scurta descriere .



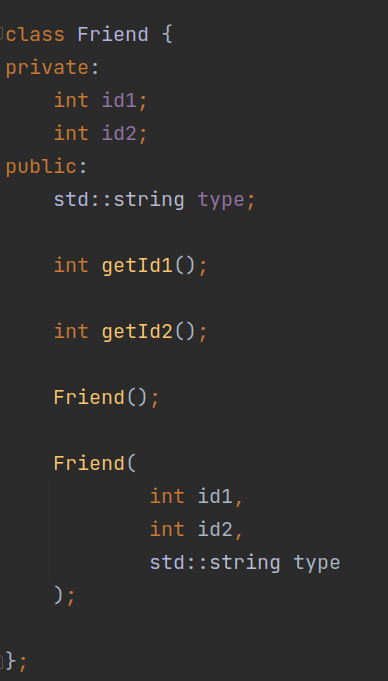
* Friend

Acestea sunt proprietatiile pentru Friend :

1. id1 în care retinem id-ul utilizatorului curent.

2. id2 în care retinem id-ul utilizatorului cu care utilizatorul cu id1 este prieten.

3. type în care retinem tipul de prieten.



* FriendRequest

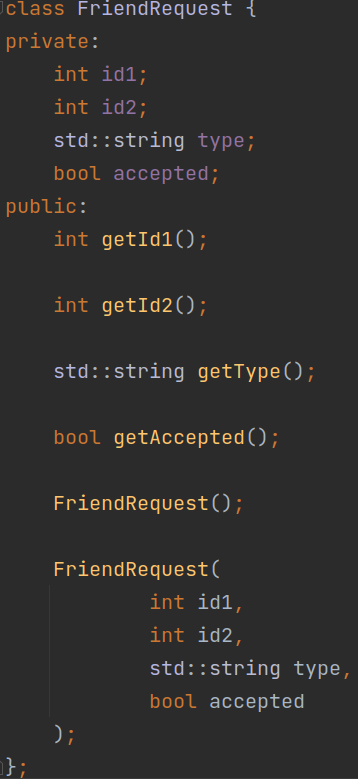
Acestea sunt proprietatiile pentru FriendRequest :

1. id1 în care retinem id-ul utilizatorului curent.

2. id2 în care retinem id-ul utilizatorului pe care utilizatorul cu id1 dorește să-l adauge ca prieten.

3. type în care retinem tipul de prieten.

4. accepted în care retinem dacă cererea este acceptata de utilizatorul cu id2.



* Message

Acestea sunt proprietatiile pentru Message :

1. id în care retinem id-ul mesajului.

2. chatId în care retinem id-ul pentru care distribui mesajul.

3. date în care retinem data în care s-a efectuat mesajul.

4. message în care retinem mesajul.



* Chat

Acestea sunt proprietatiile pentru Chat :

1. messageList o lista în care memoram mesajele.

2. idList în care retinem id-urile pentru care a fost trimis mesajul meu.

3. title în care retinem titlul mesajului.

## 

* News

Acestea sunt proprietatiile pentru News :

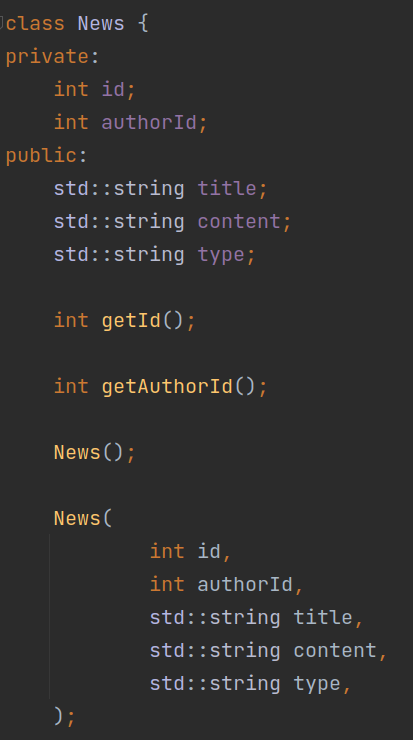
1. id în care rețin id-ul stirii.

2. authorId în care rețin id-ul autorului.

3. title în care titlul stirii.

4. id în care rețin descrierea stirii.

5. authorId în care rețin tipul stirii/postarii.



## Mediul de stocare

Se va folosi SQLite pentru stocarea datelor. Doar serverul va face request-uri către baza de date.

# Detalii de implementare

Pentru partea de server vom simula un sistem de microservicii. Practic, pentru fiecare functionalitate a serverului se va creea un server dedicat care sa manevreze functionalitatea respectiva.

Microserviciile vor fi:

* AuthServer, ce va manevra partea de autentificare : login, creeate account.
* MessageServer, ce va manevra partea de chat și de trimitere de mesaje, precum și operatiile asociate acestora.
* NewsServer, ce va manevra partea de știri: postare, editare, stergere, primire.
* FriendsServer, ce va manevra cererile de prietenie, precum și lista de prieteni.
* ProfileServer, ce va manevra profilul utilizatorilor, oferind posibilitatea editarii profilului și a cautarii de noi prieteni

Pentru partea de mesagerie se va folosi o criptare de tip end-to-end folosiind AES-128.

Comunicarea dintre client și server se va face sub forma de răspunsuri de tip JSON

# Concluzii

Aplicația funcționează corect, dar se pot îmbunătăți anumite aspecte.

O imbunatatire majora a aplicației ar fi creearea unei interfete grafice atractive. Spre exemplu, s-ar putea integra framework-ul Qtiii.

O metoda eficienta de a reduce consumul de date ar fi cache-uirea datelor permitand astfel aplicației să fie folosită și de persoane care se afla în zone cu internet mai slab. De asemenea se reduce și costul pentru transferul datelor.

Poate fi folosită o baza de date mult mai avansata precum PostgreSQLiv.

Implementarea autentificarii folosind coduri unice trimise prin SMS este un alt mod prin care se poate spori securitatea platformei

# References

i <https://profs.info.uaic.ro/~computernetworks/>

ii [https://www.sqlite.org](https://www.sqlite.org/)

iii [https://www.qt.io](https://www.qt.io/)

iv [https://www.postgresql.org](https://www.postgresql.org/)