

# Raport - TopMusic (B)

Bicu Daniel, grupa A1, anul II

16 Ianuarie 2020

## 1 Introducere

Raportul de fata are rolul de a descrie concepte utilizate in aplicatia Top Music, aplicatie care va fi descrisa mai jos.

Proiectul Top Music presupune realizarea unei aplicatii de tip client/server ce are rolul de a face managementul unui top muzical, pe baza preferintelor, aici acestea fiind reprezentate de voturi, a unor utilizatori(clienti). In cadrul aplicatiei, utilizatorii vor putea fi de tipuri diferite, tipuri ce vin totodata cu plusuri/minusuri din perspectiva functionalitatilor aplicatiei.

Functionalitatile aplicatiei vor fi reprezentate de: crearea unui cont(inregistrarea) de 2 tipuri diferite: administrator/ user normal, partea de log in, cat si de comenzi de tipul: adaugarea unei melodii la un anumit top, votarea unei melodii, afisarea topului general curent, afisarea topului pentru un anumit gen muzical. Dupa cum este deja specificat, functionalitatile aplicatiei sunt modelate si de catre tipul utilizatorului care este logat, astfel:

1. Administratorul va avea in plus, fata de utilizator, dreptul de a sterge melodii, cat si de a putea restriciona optiunea de vot a unui utilizator.
2. Utilizatorul obisnuit va putea sa voteze o anumita melodie, poate sa adauge si comentarii la melodii, sa vizualizeze topuri muzicale si sa obtina informatii generale despre lista administratorilor, cat si despre aplicatie.

## 2 Tehnologii utilizate

Aplicatia Top Music este o aplicatie de tip client/server, prin urmare pentru a realiza conexiunea client - server vom folosi protocolul TCP/IP. Totodata, deoarece dorim ca aplicatia sa permita conectarea mai multor clienti, cat si sa ofere tuturor clientilor raspuns, vom folosi TCP/IP concurent, un motiv in plus pentru aceasta alegere poate fi constiuit si de viteza mai mare pe care aceasta "concurenta" o ofera, cat si de oportunitatea servirii multiplilor clienti in acelasi timp.

**Servirea multiplilor clienti:** Serverul aplicatiei va crea cate un proces copil pentru fiecare client, prin urmare o alta tehnologie utilizata ar fi reprezentata de paralelismul utilizat cu ajutorul primitivelor din limbajul C, precum primitiva `fork()`. Dupa crearea unui proces copil, fiecare proces de acest tip se va ocupa de cate un client.

#### Motivatii ale alegerii protocolului:

- Modelul TCP/IP realizeaza o conexiune intre puncte, fiecare dintre acestea fiind definite de catre un port TCP, cat si de o adresa IP. Totodata TCP se ocupa si de marimea segmentului de date, cat si de rata de transfer, prin urmare acest model ajuta la evitarea conflictelor de retea ce ar putea aparea.
- Modelul TCP/IP ofera totodata siguranta transferului ordonat a unui flux de date(octeti), asigurand astfel si transmiterea ordonata a datelor, lucru ce ajuta la aplicatia Top Music, deoarece fara ca datele sa fie transmise ordonat interogarilor asupra bazei de date vor fi unele eronate, acestea avand nevoie de anumiti parametrii.

#### De ce TCP/IP si nu UDP?

Alegerea protocolului este influentata de tipul utilizarii aplicatiei, astfel: in cadrul aplicatiei Top Music, datele transmise spre exemplu unui nou client trebuie sa fie unele corecte, ordonate, dar si complete, fara alte pierderi, prin urmare, am optat pentru versiunea de TCP/IP, astfel acest model asigura transferul pachetelor cu o rata foarte mica de eroare. Un protocol UDP ar fi mai bun in acest caz doar din perspectiva timpului de asteptarea al unui raspuns pentru un client, insa nu si din perspectiva integritatea datelor.

## 3 Arhitectura aplicatiei

Structura aplicatiei va avea ca si componenta: un server, clientii, cat si o baza de date. Structura prezentata si in schema de mai jos (Figure 1.).

Aplicatia va utiliza o baza de date, iar ca sistem de gestiune pentru acesta se va folosi SQLITE3, iar ca interfata: C/C++. Schema bazei de date va fi consolidata din mai multe tabele: users, melodies, comments. Pentru folosirea acestor tipuri de date a fost utilizata biblioteca sqlite3. In urmatoarea figura se pot observa bibilotecile folosite de catre aplicatie.

Acceptarea multiplilor clienti este posibila prin intermediul folosirii TCP/IP - paralel (concurrent), implementat cu primitiva `fork()` din C sub linux. Definim fiecare proces copil ca reprezinta un client, in momentul in care acesta va dori sa ruleze o comanda, aceasta va fi trimisa server-ului si acesta va lua decizii in functie de un anumit cod trimis din unele din meniurile afisate sau in functie de datele primite de la user. Daca utilizatorul are drepturi, serverul va interoga baza de date in scopul trimiterii raspunsului copilului. In cazul in care user-ul

nu are drepturi, raspunsul va fi unul negativ si se va transmite si argumentul/motivul pentru care nu a fost evaluata cererea, fapt ce usureaza utilizarea aplicatiei, dar si ofera o buna interactiune cu clientul. In cazul in care user-ul este un administrator, acesta va avea si alte drepturi, care vor fi evaluate tot de catre server. O schema mai complexa ar putea fi reprezentata si de modul in care se face partea de log-in/sign up. In acest caz serverul va cere de la client/copil informatii precum: nume, gen, descriere sumara, preferinte. Aceste informatii pot sa ofere client-ului mai apoi niste sugestii in legatura cu melodiile pe care poate ar dori sa le asculte. Acest lucru poate fi realizat cu ajutorul bazei de date implicate, baza de date care poate implica o noua tabela cu id-ul user-ului si cu preferintele acestuia.

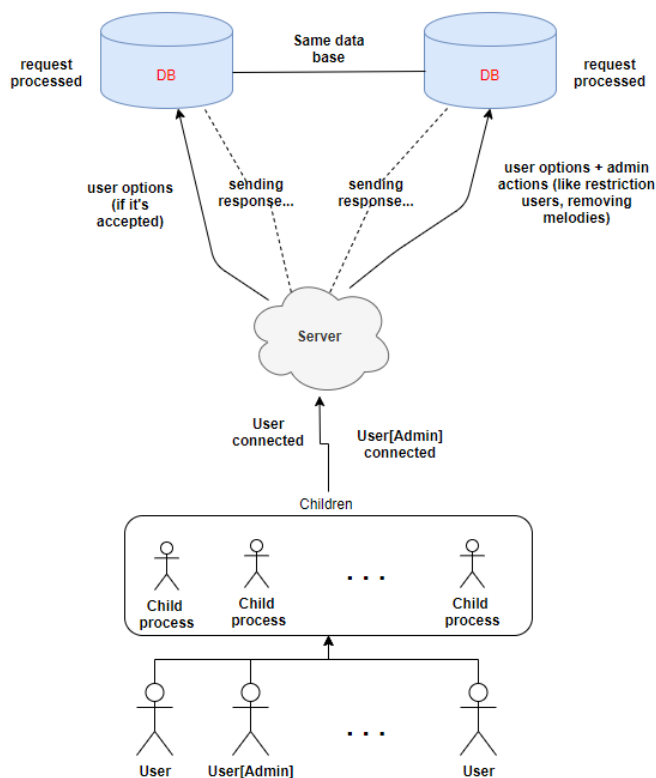


Figure 1: Schema simplificata a aplicatiei

## 4 Detalii de implementare

Aplicatia va avea un server care va prelua datele transmise de catre client si le va procesa, triminand inapoi raspunsurile descrise mai sus. Astfel, clientul poate rula comenzi prevazute in 2 meniuri, unul dintre acestea fiind general despre aplicatie, care contine si comenzile de log-in si register, iar celalalt meniu are ca scop procesarea comenzilor referitoare la functiile propriu-zise ale aplicatiei precum: afisarea top-urilor, inserare melodii, inserare comentarii etc. \*De mentionat ca al doilea meniu apare pentru utilizator doar daca acesta se va loga in aplicatie.

**Scenarii de utilizare:** Exista diferite situatii in care aplicatia ar putea avea un comportament ciudat, acesta ridicand riscul de a crea confuzie utilizatorului, astfel voi prezenta mai jos cateva cazuri plus solutia acestora.

1. In cazul in care serverul preia date de la un utilizator, dar clientul iese fara a se deconecta propriu-zis, serverul va fi responsabil de pierderea clientului si va trebui sa elimine atat copilul care se ocupa de acel client, cat si conexiunile sale, astfel incat o cerere care ar fi trebuit sa fie intoarsa catre clientul care tocmai a iesit, va fi eliminata, fara a mai fi afisata.
2. In cazul in care clientul nu are rol de administrator, acesta nu va avea acces la anumite comenzi, astfel la inceputul conexiunii sale anumite actiuni ii vor fi restrictionate, astfel va fi anuntat in legatura cu anumite erori ce ar putea sa ii ridice niste semne de intrebare pe viitor.
3. In cazul in care conexiunea este realizata cu succes, iar server-ul preia datele corespunzator, flow-ul aplicatiei va fi realizat cu succes, astfel utilizatorul va putea sa intreprinda comenzile dorite, iar raspunsurile vor fi cele asteptate.

## 5 Concluzie

Aplicatia poate sa vina pe viitor cu feature-uri noi precum

1. Adaugarea unei liste de preferinte in momentul in care se inregistreaza(sign-up). Aceasta lista va retine doar cuvintele cheie in cautarea melodiilor din acel top, mai apoi fiindu-i afisate clientului melodiile de care ar putea fii interesat.
2. Un comentariu adaugat de catre un administrator va fi afisat de fiecare data primul, acest comentariu putand reprezenta un advertisement/ o stire de interes mai mare pentru utilizator.
3. La fiecare log-in, clientul ar putea primi ca si News melodia/melodiile la care au fost postate cele mai multe comentarii/ voturi. Acest news ar putea ajuta clientul sa descopere o noua melodie, care poate ii va placea,

astfel isi poate schimba eventual si preferintele in legatura cu genul de muzica pe care il asculta deobicei.

## References

<https://profs.info.uaic.ro/computernetworks/cursulaboratorul.php>  
<https://support.holmsecurity.com/hc/en-us/articles/212963869-What-is-the-difference-between-TCP-and-UDP->  
<https://www.sqlite.org/cintro.html>  
<https://profs.info.uaic.ro/andrei.panu/>  
[https://www.sqlite.org/c3ref/c\\_abort.html](https://www.sqlite.org/c3ref/c_abort.html)  
[https://www.tutorialspoint.com/sqlite/sqlite\\_c\\_cpp.htm](https://www.tutorialspoint.com/sqlite/sqlite_c_cpp.htm)  
<https://man7.org/linux/man-pages/>