

# SPRC 2021-2022 - Tema 1 - Aplicație Client/Server pentru baze de date folosind RPC

Responsabil: Bogdan Mocanu

Titular curs: Florin Pop

## 1 Obiectivele temei

În cadrul temei veți dezvolta o aplicație de tip Client/Server folosind conceptul de Remote Procedure Call (RPC) pentru gestiunea unei baze de date cu înregistrări provenite de la o serie de senzori în anumite intervale de timp conform figurii 1.

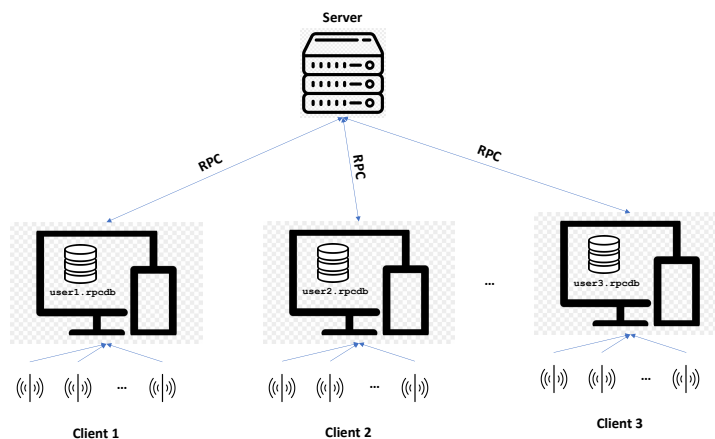


Figura 1: Prezentare generală a soluției propuse.

Obiectivele principale ale acestei teme de laborator sunt:

- realizarea unei aplicații de tip client/server bazată pe RPC cu cereri/răspunsuri multiple prin care se poate gestiona o bază de date stocată la distanță;
- scrierea de specificații pentru o aplicație simplă de administrare a unei baze de date;
- gestionarea eficientă a încărcării unui server de aplicații;
- utilizarea de structuri de date pentru aplicații ce folosesc protocolul RPC.

Cunoștințele necesare rezolvării acestei teme de casă sunt următoarele:

- programare în C/C++, structuri, funcții, lucrul cu fișiere binare/text;
- noțiuni privind folosirea utilitarului RPCGEN și a compilatorului gcc.

## 2 Descrierea soluției cerute

Pentru această temă ne propunem implementarea unei soluții de tip Client/Server pentru gestiunea unei baze localizată pe un server de aplicații. Aplicația Client/Server va fi implementată folosind protocol RPC.

Soluția dezvoltată se numește RPC+DB și include atât componenta de Server cât și cea de Client pentru gestionarea datelor provenite de la diverși senzori în intervale de timp.

### 2.1 Server

Serverul gestionează în memorie o bază de dată. Înregistrările din baza de date reprezintă datele monitorizate provenite de la o serie de  $n$  senzori pe anumite intervale de timp sub forma:

```
1 struct SensorData
2 {
3     int dataId;      //identificator date
4     int noValues;    // numarul de valori
5     float *values    // valorile monitorizate
6 }
```

Listing 1: Structura de gestiune a datelor de la senzori.

Serverul execut o serie de operații de lucru cu baza de date. Un set minimal de operații este:

- **Add/Update** – adăugarea/actualizarea unui înregistrări noi (dacă deja există o înregistrare cu identificatorul specificat, se va întoarce o eroare);
- **Del** - ștergerea unei înregistrări specificate prin identificator (dacă aceasta nu există se va genera un cod de eroare);
- **Update** - înlocuirea unei înregistrări specificate prin identificator cu date noi (dacă nu există înregistrarea se va adăuga ca înregistrare nouă);
- **Read** - citirea unei înregistrări specificate prin identificator (dacă aceasta nu există se va genera un cod de eroare);
- **GetStat** - obținerea de statistici pentru o înregistrare (valoare minimă, valoare maximă, valoare medie și mediană);
- **GetStatAll** - obținerea de statistici generale pe toată baza de date.

În plus serverul oferă și 2 comenzi pentru încărcare (**Load**) și pentru stocare (**Store**).

### 2.2 Client

Clientul este o aplicație CLI care transmite o serie de comenzi către server prin intermediul unor comenzi text. Comenzile sunt transmise către server folosind RPC.

Operațiile pe care aplicația Client le poate face sunt următoarele:

- **Login** – adăugarea/actualizarea unui înregistrări noi (dacă deja există o înregistrare cu identificatorul specificat, se va întoarce o eroare);
- **Logout** - terminarea execuției programului Client;
- **Load** - încărcarea bazei de date locală aferentă fiecărui Client;
- **Store** - salvarea bazei de date aferentă fiecărui Client pe disk;
- **Add** - adăugarea unei înregistrări noi cu tratarea duplicatelor;
- **Del** - ștergerea unei înregistrări specificate prin identificator (dacă aceasta nu există se va genera un cod de eroare);

- **Update** - înlocuirea unei înregistrări specificate prin identificator cu date noi (dacă nu există înregistrarea se va adăuga ca înregistrare nouă);
- **Read** - citirea unei înregistrări specificate prin identificator (dacă aceasta nu există se va genera un cod de eroare);
- **GetStat** - obținerea de statistici pentru o înregistrare (valoare minimă, valoare maximă, valoare medie și mediană);
- **GetStatAll** - obținerea de statistici generale pe toată baza de date.

Opțiunea de **Login** realizează o autentificare a Clientului la Server, prin care acesta primește o cheie de sesiune generată de server (un număr natural de tipul **unsigned long**. Server-ul păstrează o listă cu perechi (**user**, **session\_key**) pentru gestiunea clienților. Comanda **Login** poate eșua (fiind respinsă de server) dacă există deja un utilizator cu același utilizator autentificat în sistem.

Fiecare Client are propria bază de date pe care efectuează operații. După autentificare, Clientul poate încărca de pe disk baza de date existentă (creată în sesiunile anterioare) prin comanda **Load** sau poate crea o bază nouă în memorie dacă nu se execută comanda **load**. Dacă la execuția comenzii **load** nu se găsește baza de date, se va crea una nouă. Dacă comanda **Load** se execută după efectuarea altor comenzi, aceasta va eșua.

Prin comanda **Store** Clientul suprascrie baza de date de pe disk cu cea existentă în memorie (fără a șterge memoria). Dacă după executarea comenzii **Store** se mai execută alte comenzi, efectul acestora se va regăsi doar în memorie, până la o nouă execuție a comenzii **Store**.

Numele bazei de date poate va fi ales de către dezvoltatorul aplicației și ar putea fi de forma: **user.rpcdb**.

Terminarea execuției programului Client se realizează prin execuția comenzii **Logout**. Comanda **Logout** închide sesiunea prin ștergerea perechii **user**, **session\_key** păstrate de server și prin eliberarea memoriei folosite pentru baza de date.

Funcționalitatea minimală privind gestiunea bazei de date cuprinde:

- **Add** - crearea unei înregistrări noi în baza de date. Se va tratat cazul în care în baza de date există deja o înregistrare cu identificatorul specificat. În acest caz comanda **Add** va returna o eroare;
- **Read** - citirea unei înregistrări specificate prin identificator. În cazul în care nu există o înregistrare cu identificatorul specificat comanda **Read** va returna o eroare;
- **ReadAll** - citirea tuturor înregistrărilor din baza de date. În cazul în care în baza de date nu există nicio înregistrare comanda **ReadAll** returnează o eroare;
- **Del** - ștergerea unei înregistrări specificate prin identificator. Comanda **Del** returnează o eroare în cazul în care nu există nici o înregistrare cu identificatorul specificat;
- **Update** - actualizarea unei înregistrări din baza de date pe baza unui identificator. Se va tratat cazul în care în baza de date nu există o înregistrare cu identificatorul specificat. În acest caz comanda **Update** va returna o eroare;
- **GetStat** - obținerea unei statistici pentru o înregistrare (valoare minimă, valoare maximă, valoare medie și mediană). Comanda tratează cazul în care înregistrarea cu identificatorul specificat nu există. În acest caz va returna o eroare;
- **GetStatAll** - obținerea unor statistici generale (valoare minimă, valoare maximă, valoare medie și mediană) pe toată baza de date. Comanda **GetStatAll** returnează o eroare în cazul în care în baza de date nu sunt înregistrări.

### 2.2.1 Model de execuție al programului Client

```
1 login user
2 load
3 add 1110 4 1.0 2.5 0.5 3.5
4 del 2256
5 update 3345 3 7.0 2.3 9.1
6 read 2134
7 get_stat 2134
8 get_stat_all
9 store
10 logout
```

Listing 2: Model de execuție al programului Client.

## 3 Cerințele pentru realizarea aplicației RPC+BD

Pentru realizarea aplicației RPC+BD trebuie să efectuați următoarele sarcini:

- 20 de puncte - definirea interfeței care expune tipurile de date și metodele aplicației;
- 10 de puncte - descrierea specificațiilor de funcționare (prin comentarii în interfață) pentru toate tipurile de date și funcțiile definite în interfață;
- 30 de puncte - implementarea aplicației Server care păstrează baza de date și care este gestionată prin operațiile descrise în interfață;
- 30 puncte - implementarea aplicației Client care va citi comenzile de executat dintr-un fișier text permiț ca parametru al comenzii și le va executa în ordine;
- 10 puncte - testarea soluției complete Client/Server cu cel puțin 3 fișiere pentru baza de date;
- BONUS 10 puncte - identificarea în literatura de specialitate a unui set de date real și utilizarea acestuia pentru testarea soluției RPB+BD.

## 4 Mențiuni suplimentare

- tema se va implementa în C/C++;
- specificațiile de funcționare sunt descrise de cel care realizează implementare aplicației, nu sunt impuse de către cei care vor evalua aplicația RPC+DB;
- evaluarea se va face pe baze fișierelor de test (de bază) puse la dispoziție și a unor alte fișiere de test propuse de evaluator pe baze testelor de bază;
- testele de bază vor conține toate operațiile definite în specificații (cel puțin o dată);
- vă puteți baza pe specificațiile oficiale [ONC+ RPC Developer's Guide](#) și pe aplicația dezvoltată în cadrul laboratorului RPC.