

Documentatie finala-Monitorizarea traficului

Craciun Daria

Facultatea de informatica "Al. I. Cuza" Iasi

1 Introducere

Proiectul *Monitorizarea Traficului* constă în dezvoltarea unei aplicații de tip server-client care permite participanților la trafic accesul la informații utile pentru evitarea incidentelor și optimizarea deplasării. Scopul aplicației este de a oferi informații în timp real, precum rapoarte despre incidente, restricții de viteză, prețuri la carburanți și condiții meteo, prin intermediul unui server centralizat care primește și procesează datele trimise de clienți.

Aplicația include funcționalități pentru raportarea și distribuirea automată a evenimentelor din trafic, precum și abonarea utilizatorilor la servicii suplimentare, cum ar fi notificările personalizate. Acest sistem este destinat să ruleze într-un mediu distribuit, cu suport pentru multiple conexiuni simultane.

Totodată, programul implementează principiul următor: Dacă cineva raportează un accident la anumite coordonate, clienții care ajung ulterior în acea locație vor fi solicitați să confirme existența accidentului.

2 Tehnologii Aplicate

Prin utilizarea metodei de concurență TCP, aplicația va permite conectarea simultană a mai multor clienți. Serverul TCP concurent va crea câte un fir de execuție pentru fiecare client. Se va crea un socket pentru server, în care vor fi stocate informațiile acestuia. După aceasta, se efectuează *bind* la port, iar serverul începe să asculte mesajele primite de la clienți. În client, vor fi create mai multe fire de execuție pentru citirea datelor de la server, trimiterea comenzilor și raportarea vitezelor.

În cadrul aplicației, biblioteca cURL este utilizată pentru a obține condițiile meteo din Iași. Clientul realizează periodic o cerere GET către un API extern de prognoză meteo, iar răspunsul primit conține detalii precum starea vremii actuale, temperatura, precipitațiile și alte informații relevante. Aceste date sunt afișate utilizatorilor, oferind un context suplimentar despre condițiile exterioare care ar putea influența traficul.

3 Structura aplicației

Datele primite de la clienți vor fi procesate pe server, iar răspunsurile corespunzătoare vor fi trimise înapoi. Serverul comunică cu două baze de date:

1. Baza de date pentru autentificarea utilizatorilor:

- Stochează informații despre conturile utilizatorilor, inclusiv datele necesare pentru autentificare.
- Conține detalii despre abonamentele utilizatorilor la știri.

2. Baza de date pentru harta orașelor:

- Conține un tabel cu aproximativ 900 de străzi, în care sunt stocate informații precum:
 - accidente;
 - limitele de viteză;
 - calitatea drumurilor.
- Include un tabel `"benzinării"`, care stochează informații despre stațiile PECO și prețurile actualizate ale combustibililor.
- Include un tabel `"mașini"`, care este actualizat automat în funcție de clienții conectați la server. În acest tabel sunt reținute următoarele informații:
 - strada pe care se află clientul logat;
 - viteza maximă permisă pe strada respectivă;
 - viteza cu care circulă clientul;
 - statusul contului, indicând dacă utilizatorul are acces la opțiunile suplimentare (de exemplu, știrile și informațiile despre benzinării).
- Include tabelul `"connect"` care reprezintă conexiunile dintre străzi.

Această structură a bazelor de date permite serverului să gestioneze eficient atât autentificarea utilizatorilor, cât și informațiile legate de trafic, contribuind la o experiență optimă pentru toți clienții conectați.

4 Aspecte de implementare

Conexiunea dintre server și client se va realiza printr-un socket. În client vor exista patru fire de execuție (`*threads*`) care vor funcționa în paralel, fiecare având un rol specific.

Aplicația va include următoarele funcționalități:

- Menu: un meniu care se deschide automat la deschidere, unde descrie comenzile disponibile.
- Login: utilizatorul se conectează folosind un nume de utilizator și o parolă.
- Register: utilizatorul își creează un cont nou.
- Benzinarie: utilizatorul solicită informații despre prețurile combustibililor și 3 cele mai apropiate benzinării în funcție de coordonatele lui.
- Vreme: utilizatorul solicită informații despre condițiile meteorologice.
- Sport: utilizatorul solicită informații despre evenimentele sportive.
- Accident: utilizatorul poate raporta accidente în trafic pe coordonatele care se află.
- Logout: utilizatorul se deconectează.
- Quit: aplicația este închisă.

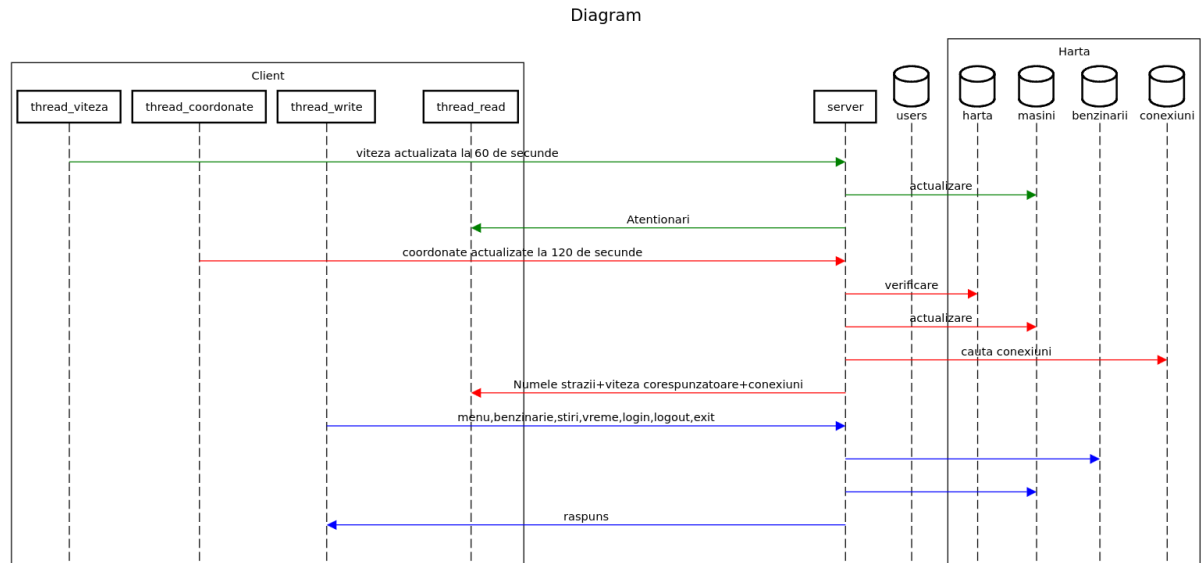


Fig. 1. Enter Caption

În cadrul unor alte funcții, aplicația va trimite automat viteza și coordonate. Viteza este calculată astfel încât să nu fie variații prea mari, aceasta poate crește sau scădea cu 10 km pe minut. După login este trimis un coordonat random, apoi se alege aleator din niște coordonate trimise de la server, care reprezintă conexiunile străzii inițiale.

5 Concluzii

În concluzie, aplicația poate fi îmbunătățită prin adăugarea unei opțiuni de a putea alege traseele cele mai scurte către o anumită destinație. Aceste rute ar putea fi calculate în funcție de viteza cu care se circulă.

References

1. <https://edu.info.uaic.ro/computer-networks/files/7rcProgramareaInReteaIIIEn.pdf>
2. https://www.tutorialspoint.com/sqlite/sqlite_select_query.htm
3. <https://www.geeksforgeeks.org/curl-command-in-linux-with-examples/>
4. <https://curl.se/libcurl/>
5. <https://overpass-turbo.eu/>