Universitatea Tehnică a Moldovei

Facultatea - Calculatoare Informatică și Microelectronică

**Raport**

**la Baze de Date**

Lucrarea de laborator Nr.2

Tema: Întreținerea bazei de date

A efectuat: st.gr. SI-211 Chihai Adrina

A verificat: Rodica Bulai

Chișinău 2023

# 1 Sarcina lucrării

Metode de întreținerea a bazei de date

2 Obiective

Am implementat un sistem de monitorizare pentru a urmări parametrii critici ai bazei de date, precum consumul de resurse, timpul de răspuns al interogărilor etc. Acest lucru ajută la identificarea și remedierea problemelor înainte ca acestea să afecteze serviciul.

3 Desfășurarea lucrării de laborator

Efectuarea întreținerii în PostgreSQL este importantă pentru a asigura sănătatea și performanța bazei de date. Mai jos sunt prezentate câteva comenzi și sarcini de întreținere folosite frecvent:

VACUUM:

VACUUM este folosit pentru a recupera spațiul de pe disc ocupat de tuplele moarte (înregistrările șterse sau actualizate).

Formă de bază: VACUUM [FULL] [ANALYZE] [VERBOSE] nume\_tabel;

Exemplu: VACUUM FULL ANALYZE my\_table;

ANALYZE:

ANALYZE colectează statistici despre distribuția datelor în tabele și indexuri. Acest lucru ajută optimizerul de interogare să ia decizii mai bune.

Formă de bază: ANALYZE [VERBOSE] [nume\_tabel];

Exemplu: ANALYZE my\_table;

REINDEX:

REINDEX reconstruiește unul sau mai multe indexuri.

Formă de bază: REINDEX [VERBOSE] [nume\_tabel];

Exemplu: REINDEX my\_index;

CLUSTER:

CLUSTER reorganizează fizic datele unei tabele bazate pe indexul specificat. Acest lucru poate îmbunătăți performanța interogărilor.

Formă de bază: CLUSTER nume\_tabel USING nume\_index;

Exemplu: CLUSTER my\_table USING my\_index;

VACUUM FULL:

VACUUM FULL rescrie întreaga tabelă și este o formă mai agresivă a VACUUM. Necesită un blocaj exclusiv pe tabel.

Formă de bază: VACUUM FULL [VERBOSE] nume\_tabel;

Exemplu: VACUUM FULL my\_table;

pg\_dump:

pg\_dump este o utilitate în linie de comandă pentru realizarea unui backup al unei baze de date PostgreSQL.

Exemplu: pg\_dump -U username -d database\_name -f backup\_file.sql

pg\_restore:

pg\_restore este folosit pentru a restaura o bază de date PostgreSQL dintr-un arhivă creată de pg\_dump.

Exemplu: pg\_restore -U username -d database\_name backup\_file.sql

CHECKPOINT:

Declanșează manual un checkpoint. Un checkpoint curăță toate paginile murdare din cache-ul de memorie către disc.

Exemplu: CHECKPOINT;

RESET:

Resetează unul sau mai mulți parametri de configurare la valorile lor implicite.

Exemplu: RESET nume\_parametru;

SET:

Schimbă valoarea unui parametru de configurare pentru sesiunea curentă.

Exemplu: SET nume\_parametru = valoare;

**Concluzie**

În ansamblul său, implementarea și administrarea sistemului de gestionare a bazelor de date în PostgreSQL a fost realizată cu succes. Procesul a implicat instalarea și configurarea inițială a platformei, definirea și configurarea utilizatorilor și a rolurilor acestora, monitorizarea performanțelor și optimizarea interogărilor, precum și stabilirea unui riguros plan de mentenanță.

Prin aplicarea acestor etape și politici, am creat un mediu robust și sigur pentru stocarea și gestionarea datelor. Politicile de securitate și controlul accesului asigură protecția informațiilor sensibile, iar back-up-urile periodice oferă o soluție eficientă pentru recuperarea datelor în caz de incidente. Actualizările și patch-urile de securitate mențin sistemul într-o stare optimă și sigură.

În concluzie, acest plan de gestionare a bazelor de date în PostgreSQL furnizează o fundație solidă pentru buna funcționare a sistemului și protejarea datelor. Este esențial să se continue monitorizarea și revizuirea periodică a planului de mentenanță pentru a răspunde la schimbările tehnologice și la evoluția nevoilor organizației.