

Universitatea Tehnică “Gheorghe Asachi” Iași

Facultatea de Automatică și Calculatoare

Specializarea “Calculatoare și Tehnologia Informației”

Proiect Baze de Date Car Dealership

Dumea Emilian, Grupa 1311B

Cadru Didactic Coordonator: dr. ing. Mironeanu Cătălin

Descrierea Proiectului

Proiectul are ca scop gestionarea eficientă a activităților unui dealership auto, incluzând evidența vehiculelor disponibile, tranzacțiile realizate, test drive-urile programate, dotările suplimentare, precum și detaliile clienților. Aplicația permite vizualizarea și administrarea acestor informații printr-o bază de date optimizată, reducând riscul erorilor și asigurând integritatea datelor.

Acest sistem este conceput pentru a fi intuitiv, oferind un echilibru între flexibilitate și complexitate, prin utilizarea unor constrângeri bine definite, cum ar fi cheile primare, străine, unicitatea, și validările de tip. Designul bazei de date asigură normalizarea pentru a evita redundanța datelor și pentru a facilita interogările rapide și sigure.

Proiectul reprezintă o soluție informatică destinată gestionării unui dealership auto, acoperind multiple aspecte operaționale, de la administrarea vehiculelor până la gestionarea tranzacțiilor financiare și relaționarea cu clienții. Sistemul oferă o bază de date bine structurată pentru stocarea și manipularea eficientă a informațiilor, asigurând integritatea și accesibilitatea acestora.

➤ **Gestionarea vehiculelor**

Dealership-ul administrează un stoc variat de vehicule, atât noi, cât și second-hand. Fiecare vehicul este înregistrat în baza de date cu detalii importante precum:

- marca și modelul,
- anul fabricației,
- tipul de combustibil (benzină, motorină),
- prețul și starea (nou sau second-hand).

Sunt incluse informații suplimentare despre kilometraj și istoricul de întreținere, care influențează prețul acestora. Vehiculele dispun, de asemenea, de o categorie de dotări ce include caracteristici precum camere spate, senzori de parcare, camere 360 sau plafon panoramic.

Aceste dotări pot fi personalizate pentru fiecare vehicul și adaugă costuri suplimentare la prețul de bază.

➤ **Test drive-uri**

O funcționalitate importantă a sistemului este gestionarea test drive-urilor. Clienții pot programa test drive-uri pentru vehiculele disponibile.

➤ **Achiziția vehiculelor**

Achiziția vehiculelor este un proces bine definit în cadrul dealership-ului. Sistemul permite clienților să cumpere vehicule integral, utilizând metode de plată diverse:

- numerar,
- transfer bancar,
- opțiuni de finanțare (rate lunare, cu dobânzi variabile).

➤ **Gestionarea clienților**

Fiecare client este înregistrat cu informații detaliate:

- nume, prenume,
- seria și numărul actului de identitate,
- adresa (strada, oraș),
- numărul de telefon și adresa de email.

Aceste date sunt esențiale pentru identificarea clienților și gestionarea operațiunilor asociate, cum ar fi achizițiile sau programările test drive-urilor. Sistemul oferă o imagine completă asupra istoricului fiecărui client, permițând dealership-ului să personalizeze experiența utilizatorului.

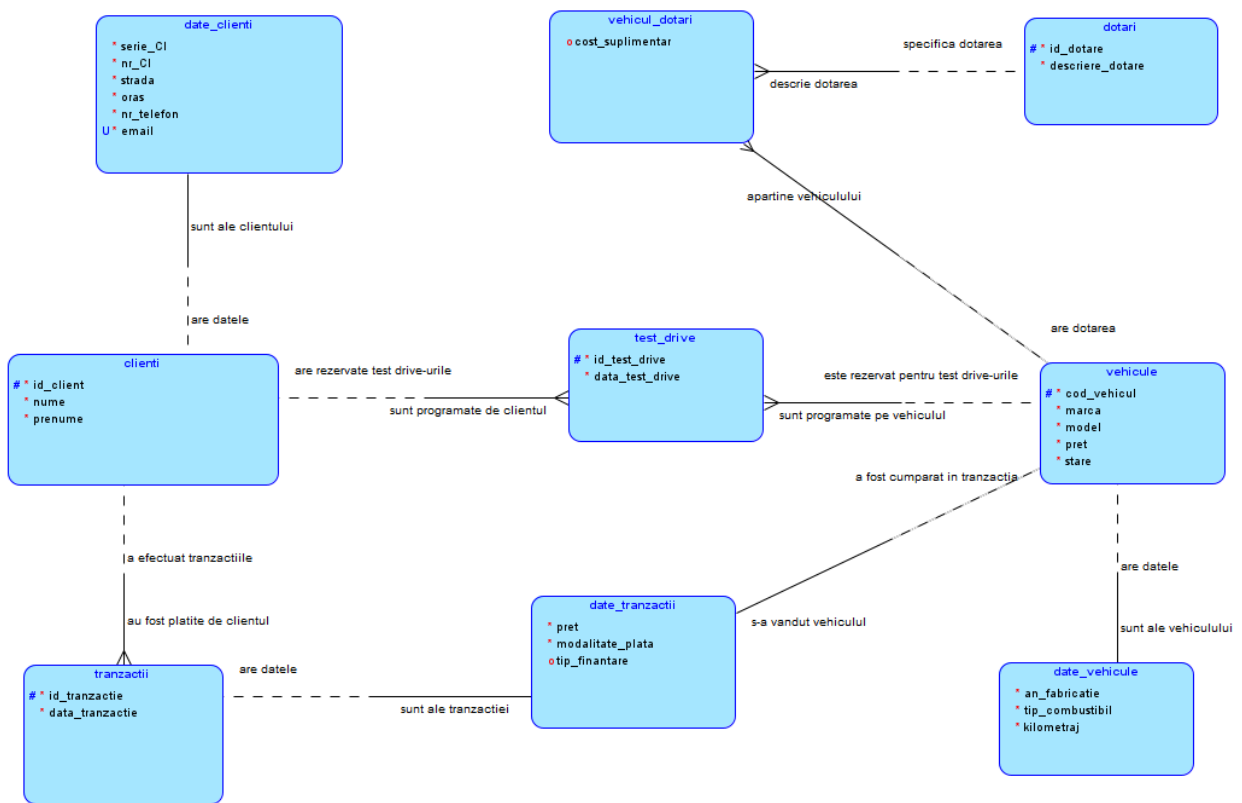
➤ **Tranzacții și raportare**

Tranzacțiile de vânzare sunt un alt element central al sistemului. Fiecare tranzacție este înregistrată cu următoarele informații:

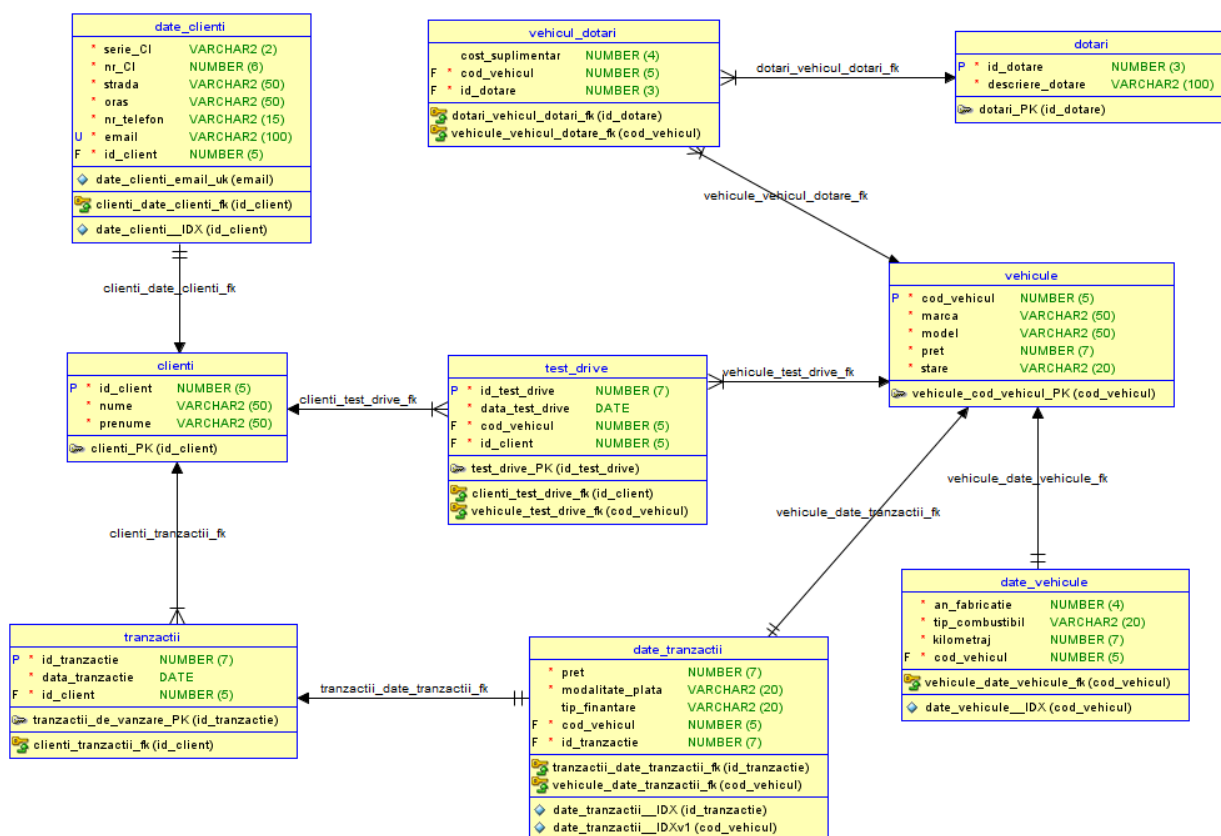
- vehiculul achiziționat,
- clientul,
- metoda de plată,
- detalii financiare legate de preț și reduceri.

Sistemul permite generarea de rapoarte, cum ar fi analiza celor mai vândute modele, metodele de plată preferate și performanțele financiare. Aceste rapoarte ajută la luarea deciziilor strategice și la optimizarea stocului.

Modelul Logic



Modelul Relațional



Explicații Structurale

Baza de date este formată din următoarele entități principale:

- **clienti:**
Gestionează informațiile personale ale clienților, cum ar fi numele, prenumele și identificatorul unic al fiecărui client.
- **date_clienti:**
Reprezintă detalii suplimentare despre clienți, cum ar fi seria și numărul cărții de identitate, adresa, numărul de telefon și emailul. Este asociată în mod direct cu entitatea **clienti**.
- **vehicule:**
Reprezintă vehiculele disponibile în dealership, incluzând informații despre marcă, model, preț, stare și alte detalii relevante.

- **date_vehicule:**
Detaliază caracteristicile suplimentare ale vehiculelor, cum ar fi anul fabricației, tipul combustibilului și kilometrajul. Este asociată în mod direct cu entitatea **Vehicule**.
- **dotari:**
Include detalii despre dotările suplimentare disponibile pentru vehicule, precum sistem audio, senzori de parcare, și alte funcționalități suplimentare.
- **vehicul_dotari:**
Relaționează vehiculele cu dotările lor, incluzând costul suplimentar asociat fiecărei dotări.
- **tranzactii:**
Ține evidența achizițiilor realizate de clienți, incluzând data tranzacției și relația cu clientul și vehiculul implicat.
- **date_tranzactii:**
Detaliază informațiile financiare legate de tranzacții, cum ar fi prețul final, modalitatea de plată și detaliile despre finanțare. Este asociată cu tranzacțiile și vehiculele implicate.
- **test_drive:**
Permite programarea test drive-urilor pentru clienți, incluzând informații despre client, vehicul și data test drive-ului.

Relatiile dintre entitati:

1. clienti - date_clienti

- **Relație: 1:1**
Fiecare client are asociate un set unic de detalii adiționale (adresă, număr de telefon, email), iar fiecare înregistrare din **date_clienti** aparține unui singur client. Relația este esențială pentru a separa informațiile personale de detaliile de contact și adresă.
- **Cheie străină:**
id_client în tabela **date_clienti** este cheie străină către id_client din tabela **clienti**.

2. clienti - test_drive

- **Relație:** 1:M

Un client poate programa mai multe test drive-uri pentru vehicule diferite, dar fiecare test drive este asociat unui singur client. Relația permite urmărirea test drive-urilor individuale pentru fiecare client.

- **Cheie străină:**

id_client în tabela **test_drive** este cheie străină către id_client din tabela **clienti**.

3. clienti - tranzactii

- **Relație:** 1:M

Un client poate efectua mai multe tranzacții de achiziție, dar fiecare tranzacție este legată de un singur client. Relația reflectă istoria de achiziții a fiecărui client.

- **Cheie străină:**

id_client în tabela **tranzactii** este cheie străină către id_client din tabela **clienti**.

4. vehicule - date_tranzactii

- **Relație:** 1:1

Fiecare vehicul poate fi vândut într-o singură tranzacție, iar fiecare tranzacție este asociată unui singur vehicul. Relația este necesară pentru a urmări detaliile financiare ale fiecărei achiziții.

- **Cheie străină:**

cod_vehicul în tabela **date_tranzactii** este cheie străină către cod_vehicul din tabela **vehicule**.

5. Tranzacții - date_tranzactii

- **Relație:** 1:1

Fiecare tranzacție are detalii financiare asociate, cum ar fi prețul total, modalitatea de plată sau tipul de finanțare. Fiecare set de detalii financiare este asociat unei singure tranzacții.

- **Cheie străină:**

id_tranzactie în tabela **date_tranzactii** este cheie străină către id_tranzactie din tabela **tranzactii**.

6. vehicule - date_vehicule

- **Relație:** 1:1

Fiecare vehicul are informații suplimentare, precum anul fabricației, tipul de combustibil și kilometrajul. Relația este de tip 1:1, deoarece aceste informații sunt unice pentru fiecare vehicul.

- **Cheie străină:**

cod_vehicul în tabela **date_vehicule** este cheie străină către cod_vehicul din tabela **vehicule**.

7. vehicule - test_drive

- **Relație:** 1:M

Un vehicul poate fi asociat cu mai multe test drive-uri, deoarece mai mulți clienți pot solicita să testeze același vehicul. Fiecare test drive este asociat unui singur vehicul.

- **Cheie străină:**

cod_vehicul în tabela **test_drive** este cheie străină către cod_vehicul din tabela **Vehicule**.

8. vehicule - vehicul_dotari

- **Relație:** 1:M

Un vehicul poate avea mai multe dotări asociate, cum ar fi pachet multimedia, scaune încălzite etc. Fiecare dotare este legată de un vehicul specific.

- **Cheie străină:**

cod_vehicul în tabela **vehicul_dotari** este cheie străină către cod_vehicul din tabela **vehicule**.

9. dotari - vehicul_dotari

- **Relație: 1:M**

O dotare (cum ar fi un sistem audio sau un pachet de siguranță) poate fi asociată cu mai multe vehicule, dar fiecare înregistrare din tabela **vehicul_dotari** este legată de o dotare specifică.

- **Cheie străină:**

id_dotare în tabela **vehicul_dotari** este cheie străină către id_dotare din tabela **dotari**.

Structura și Inter-relaționarea Tabelelor

Proiectarea bazei de date respectă regulile de normalizare până la forma normală 3 (3NF) pentru a elimina redundanțele și pentru a asigura integritatea datelor. Acest lucru se realizează prin împărțirea datelor în tabele distincte, conectate prin chei primare și externe.

- **1NF:** Toate valorile din coloane sunt atomice și nu există grupuri de valori repetate.
- **2NF:** Toate coloanele non-cheie depind în totalitate de cheia primară.
- **3NF:** Nu există dependențe tranzitive între coloanele non-cheie.

Mai jos, analiza normalizării este prezentată pentru fiecare tabel:

- **Tabela clienti**

- **Atribute:** id_client (PK), nume, prenume.
- **1NF:** Fiecare coloană conține valori atomice, iar fiecare înregistrare este unică. De exemplu, câmpurile nume și prenume nu conțin valori combinate sau liste.
- **2NF:** Toate atributele (nume, prenume) depind în întregime de cheia primară id_client.
- **3NF:** Nu există dependențe tranzitive între coloanele non-cheie, iar nume și prenume nu depind unul de celălalt.

- **Tabela date_clienti**

- **Attribute:** serie_ci, nr_ci, strada, oras, nr_telefon, email, id_client (FK).
- **1NF:** Toate valorile sunt atomice (e.g., strada nu conține mai multe adrese într-un singur câmp).
- **2NF:** Atributele non-cheie depind complet de cheia compusă formată din serie_ci și nr_ci, iar id_client este o cheie străină care leagă de tabela clienti.
- **3NF:** Nu există dependențe tranzitive. De exemplu, email sau nr_telefon nu depind de alte coloane din tabel.
- **Tabela vehicule**
 - **Attribute:** cod_vehicul (PK), marca, model, pret, stare.
 - **1NF:** Valorile coloanelor sunt atomice (e.g., marca conține doar un singur brand, nu o listă de mărci).
 - **2NF:** Toate atributele depind complet de cheia primară cod_vehicul.
 - **3NF:** Nu există dependențe tranzitive. De exemplu, stare nu depinde de marca sau model.
- **Tabela date_vehicule**
 - **Attribute:** cod_vehicul (FK), an_fabricatie, tip_combustibil, kilometraj.
 - **1NF:** Toate valorile sunt atomice (e.g., tip_combustibil nu combină mai multe tipuri).
 - **2NF:** Toate atributele depind de cheia străină cod_vehicul.
 - **3NF:** Nu există dependențe tranzitive. De exemplu, kilometraj nu depinde de tip_combustibil.
- **Tabela dotari**
 - **Attribute:** id_dotare (PK), descriere_dotare.
 - **1NF:** Toate valorile sunt atomice (e.g., descriere_dotare nu combină mai multe dotări într-o singură intrare).
 - **2NF:** descriere_dotare depinde complet de cheia primară id_dotare.
 - **3NF:** Nu există dependențe tranzitive deoarece există o relație simplă între id_dotare și descriere_dotare.
- **Tabela vehicul_dotari**

- **Attribute:** cod_vehicul (FK), id_dotare (FK), cost_suplimentar.
- **1NF:** Toate valorile sunt atomice. Fiecare rând reprezintă o asociere unică între un vehicul și o dotare.
- **2NF:** Atributele non-cheie (cost_suplimentar) depind complet de cheia compusă formată din cod_vehicul și id_dotare.
- **3NF:** Nu există dependențe tranzitive. De exemplu, cost_suplimentar nu depinde de alte coloane.
- **Tabela tranzactii**
 - **Attribute:** id_tranzactie (PK), data_tranzactie, id_client (FK).
 - **1NF:** Fiecare coloană conține valori atomice (e.g., data este stocată într-un format standard, fără combinații).
 - **2NF:** Atributele depind complet de cheia primară id_tranzactie.
 - **3NF:** Nu există dependențe tranzitive între coloanele non-cheie, iar data_tranzactie este direct legată de tranzacție, nu de client.
- **Tabela date_tranzactii**
 - **Attribute:** pret, modalitate_plata, tip_finantare, cod_vehicul (FK), id_tranzactie (FK).
 - **1NF:** Toate valorile sunt atomice (e.g., modalitate_plata nu combină mai multe modalități).
 - **2NF:** Atributele depind complet de cheia compusă formată din cod_vehicul și id_tranzactie.
 - **3NF:** Nu există dependențe tranzitive. De exemplu, tip_finantare sau pret nu depind de alte coloane.
- **Tabela test_drive**
 - **Attribute:** id_test_drive (PK), data_test_drive, cod_vehicul (FK), id_client (FK).
 - **1NF:** Toate valorile sunt atomice, iar fiecare înregistrare este unică.
 - **2NF:** Atributele non-cheie (data_test_drive, cod_vehicul, id_client) depind complet de cheia primară id_test_drive.

- **3NF:** Nu există dependențe tranzitive între coloanele non-cheie. Datele test drive-ului sunt strict legate de ID-ul unic al test drive-ului

Descrierea coloanelor din entități

1. Entitatea clienti

- **id_client:** Cheie primară, valoare numerică, nenulă, obligatorie. Identificator unic pentru fiecare client. Este esențială pentru a lega informațiile din alte tabele.
- **nume:** Șir de caractere, nenul, obligatoriu. Reprezintă numele clientului, necesar pentru identificare. Nu poate fi NULL, deoarece identificarea clientului este obligatorie.
- **prenume:** Șir de caractere, nenul, obligatoriu. Asemeni numelui, prenumele clientului este critic pentru identificare completă.

2. Entitatea date_clienti

- **serie_ci:** Șir de caractere (2 caractere), nenul, obligatoriu. Seria actului de identitate este obligatorie pentru validarea identității clientului.
- **nr_ci:** Valoare numerică, nenulă, obligatorie. Precum seria, și numărul actului de identitate este necesar pentru a valida identitatea.
- **strada:** Șir de caractere, nenul, obligatoriu. Include informații despre strada pe care locuiește clientul.
- **oras:** Șir de caractere, nenul, obligatoriu. Include informații orașul de reședință al clientului.
- **telefon:** Șir de caractere, nenul, obligatorie. Numărul de telefon permite contactarea clientului.
- **email:** Șir de caractere, unic, nenul. Adresa de email este utilizată pentru comunicări și identificare. Aceasta este unica, intrucat o adresa de email poate aparține doar unui singur om și nu pot exista alte persoane cu aceeași adresa identică

- **id_client:** Cheie străină (FK), referință la entitatea client, nenulă și obligatorie. Identifică clientul căruia îi aparțin datele referite de campurile acestei entități

3. Entitatea vehicule

- **cod_vehicul:** Cheie primară, valoare numerică, nenulă, obligatorie. Identificator unic pentru fiecare vehicul.
- **marca:** Șir de caractere, nenul, obligatoriu. Marca vehiculului. Este o informație cheie pentru descrierea vehiculului.
- **model:** Șir de caractere, nenul, obligatoriu. Modelul vehiculului. Similar mărcii, contribuie la identificarea vehiculului, fiind o informație esențială în descrierea acestuia.
- **pret:** Valoare numerică cu zecimale, nenulă. Reflectă valoarea monetară a vehiculului, cu restricția să fie pozitivă. Valoarea nu poate fi negativă, deoarece ar fi lipsit de sens din punct de vedere economic.
- **stare** Șir de caractere, nenul. Indică dacă vehiculul este nou sau utilizat. Starea vehiculului (e.g., "nou", "utilizat"). Starea este esențială pentru procesul de vânzare, intrucat clientul trebuie sa aiba la cunostinta cat de utilizat a fost vehiculul.

4. Entitatea date_vehicule

- **an_fabricatie:** Valoare numerică, nenulă și obligatorie. Reprezinta anul de fabricație al vehiculului, informatie necesara pentru a cunoaste cat de in varsta este vehiculul.
- **tip_combustibil:** Text nenul și obligatoriu. Este o informație cheie pentru descrierea vehiculului
- **kilometraj:** Valoare numerică, obligatorie. Clientul trebuie sa aiba la cunostinta cat de mult a fost condus vehiculul în trecut.
- **cod_vehicul:** Cheie străină (FK), referință la entitatea vehicule, nenulă și obligatorie. Asociază datele cu vehiculul corespunzător.

5. Entitatea dotari

- **id_dotare:** Cheie primară, valoare numerică, autoincrement, nenulă și obligatorie. Reprezinta identificatorul unic pentru fiecare dotare pe care o poate avea un vehicul.

- **descriere_dotare**: Text (VARCHAR(100)), nenul și obligatoriu. Descrierea dotării. Aceasta nu poate fi nula, intrucat fiecare dotare este un adaos specific adaugat vehiculului și trebuie precizat în ce consta aceasta mai exact.

6. Entitatea vehicul_dotari

- **cod_vehicul**: Cheie străină , referință la tabelul vehicule, nenulă și obligatorie. Indică vehiculul care beneficiază de dotare.
- **id_dotare** :Cheie străină , referință la tabelul dotari, nenulă și obligatorie. Indică dotarea aplicată vehiculului.
- **cost_suplimentar**: Valoare numerică, nenulă și cu constrângerea CHECK > 0. Costul suplimentar al dotării. Valoarea trebuie să fie pozitivă pentru validitate economică.

7. Entitatea tranzactii

- **id_tranzactie**: Cheie primară, valoare numerică, nenulă și obligatorie. Identificator unic pentru fiecare tranzacție.
- **data_tranzactie**: Valoare de tip dată (DATE), nenulă și obligatorie. Indică data tranzacției, necesară pentru urmărirea cronologică.
- **id_client**: Cheie străină, nenulă. Referință către clientul care a realizat tranzacția.

8. Entitatea date_tranzactii

- **pret**: Valoare numerică cu zecimale, nenulă. Reflectă prețul total al tranzacției
- **modalitate_plata**: Text, nenul și obligatoriu. Specifică metoda de plată utilizată.
- **tip_finantare**: Text, opțional. Se completează doar pentru tranzacțiile efectuate prin finanțare.
- **cod_vehicul**: Cheie străină (FK), referință la entitatea vehicule, nenulă și obligatorie. Asociază tranzacția cu vehiculul achiziționat.
- **Id_tranzactie**: Cheie străină (FK), referință la entitatea tranzacții, nenulă și obligatorie. Asociază informațiile tranzacția cu tranzacția corespunzătoare.

9. Entitatea test_drive

- **id_test_drive**: Cheie primară, valoare numerică, nenulă și obligatorie. Identificator unic pentru fiecare test drive.
- **data_test_drive**: Valoare de tip dată, nenulă și obligatorie. Data programării test drive-ului, esențială pentru a putea sincroniza momentele în care clientul și respectiv vehiculul sunt disponibile pentru un test drive.
- **cod_vehicul**: Cheie străină (FK), referință la entitatea vehicule, nenulă și obligatorie. Specifică vehiculul implicat în test drive.
- **id_client**: Cheie străină (FK), referință la entitatea clienti, nenulă și obligatorie. Indică clientul care a solicitat test drive-ul.

Descrierea tuturor constrângerilor folosite și justificarea acestora

1. Constrângeri Primare (Primary Key)

Aceste constrângeri garantează unicitatea fiecărui rând din tabel asigurând identificarea unică a înregistrărilor.

- Entitatea **clienti**:
 - Cheie primară: **clienti_pk** - Acest atribut este unic pentru fiecare client și permite identificarea fiecărui client în mod individual. Fără această constrângere, ar fi dificil să distingem între doi clienți cu același nume.
- Entitatea **vehicule**:
 - Cheie primară: **vehicule_cod_vehicul_pk** - Asigură unicitatea fiecărui vehicul înregistrat. Această cheie este crucială pentru urmărirea și gestionarea vehiculelor în alte tabele precum test_drive sau date_tranzactii.
- Entitatea **dotari**:
 - Cheie primară: **dotari_pk** - Identifică unic fiecare tip de dotare. Acest lucru este important pentru a evita redundanța și pentru a menține integritatea datelor în vehicul_dotari.
- Entitatea **tranzactii**:

- Cheie primară: **tranzactii_de_vanzare_pk** - Permite identificarea și gestionarea fiecărei tranzacții în mod unic, necesar pentru corelarea cu alte tabele.
- Entitatea **test_drive**:
 - Cheie primară: **test_drive_pk** - Această cheie previne duplicarea înregistrărilor pentru același test drive efectuat de un client pentru un vehicul la o anumită dată.

2. Constrângeri de cheie străină (Foreign Key)

Constrângerile de **cheie străină** asigură consistența în relațiile dintre tabele și previn încălcarea integrității datelor.

- **clienti - date_clienti**:
 - Cheie străină: **clienti_date_clienti_fk** referă cheia primară din **clienti** (**id_client**). Aceasta asigură că fiecare set de informații suplimentare este asociat unui client existent.
- **dotari - vehicul_dotari**:
 - Cheie străină: **dotari_vehicul_dotari_fk** referă cheia primară din **dotari** (**id_dotare**). Previne atribuirea unei dotări inexistente unui vehicul.
- **vehicule - vehicul_dotari**:
 - Cheie străină: **vehicule_vehicul_dotare_fk** referă cheia primară din **vehicule** (**id_vehicul**). Aceasta se asigură că dotările sunt asociate doar vehiculelor existente.
- **clienti - test_drive**:
 - Cheie străină: **clienti_test_drive_fk** referă cheia primară din **clienti** (**id_client**). Fiecare test drive trebuie să fie asociat unui client valid.
- **vehicule - test_drive**:
 - Cheie străină: **vehicule_test_drive_fk** referă cheia primară din **vehicule** (**cod_vehicul**). Asigură că test drive-urile sunt efectuate doar pentru vehicule existente.
- **tranzactii - date_tranzactii**:

- Cheie străină: **tranzactii_date_tranzactii_fk** referă cheia primară din **tranzactii** (**id_tranzactie**). Aceasta garantează că detaliile tranzacțiilor sunt legate de tranzacții existente.
- **vehicule - date_tranzactii:**
 - Cheie străină: **vehicule_date_tranzactii_fk** referă cheia primară din **vehicule** (**cod_vehicul**): Asigură că doar vehiculele întrate în sistem pot fi incluse într-o tranzacție.
- **clienti - tranzactii:**
 - Cheie străină: **clienti_tranzactii_fk** referă cheia primară din **clienti** (**id_client**): Leagă tranzacțiile clienților existenți. Astfel, o tranzacție nu poate exista fără un client asociat.
- **vehicule - date_vehicule:**
 - Cheie străină: **vehicule_date_vehicule_fk** referă cheia primară din **vehicule** (**cod_vehicul**): Asigură că fiecare set de date suplimentare este asociat unui vehicul intrat în sistem

3. Constrângeri de unicitate (Unique Key)

Constrângerile **UNIQUE** sunt folosite pentru a preveni duplicarea datelor în coloane care trebuie să conțină valori unice.

- Entitatea **date_clienti:**
 - **Date_clienti_email_uk** pe coloana **email**: Fiecare adresă de email trebuie să fie unică pentru a preveni înregistrările duplicate. ajută la prevenirea erorilor și la asigurarea unei comunicări corecte cu clienții.

4. Constrângeri de tip NOT NULL

Constrângerile **NOT NULL** sunt utilizate pentru a garanta că anumite coloane nu pot avea valori nule, prevenind omisiunile în informațiile critice.

- Entitatea **clienti:**
 - **id_client**: Identificatorul clientului este fundamental pentru a face legătura între alte entități. Dacă această valoare ar fi null,

relațiile între tabele ar deveni invalide, iar gestionarea datelor despre clienți ar fi imposibilă.

- **nume:** Este crucial să cunoaștem numele clientului pentru identificare. Un client fără nume nu ar putea fi utilizat în procesele comerciale.
- **prenume:** Similar cu numele, prenumele este necesar pentru a completa identificarea unică a clientului. Este esențial pentru interacțiuni personalizate și pentru evitarea confuziilor în cazul clienților cu nume identice.

- Entitatea **date_clienti:**

- **serie_c:** Seria cărții de identitate este necesară pentru a valida identitatea clientului. Fără această valoare, verificarea identității ar fi incompletă.
- **nr_ci:** Numărul cărții de identitate, împreună cu seria, asigură unicitatea în identificarea fiecărui client. Această valoare este utilizată pentru a evita erorile în cazul unor înregistrări duplicate.
- **strada:** Adresa clientului este esențială pentru comunicare, livrări (dacă e cazul) sau procesarea documentelor oficiale. Un client fără adresă ar fi incomplet definit.
- **oras:** Orașul este necesar pentru a cunoaște locația clientului. Este util pentru segmentarea clienților și alte analize comerciale.
- **nr_telefon:** Numărul de telefon este indispensabil pentru contactul rapid cu clientul, fie pentru confirmarea comenzilor, programarea unui test drive, sau rezolvarea problemelor.
- **email:** Emailul este un canal de comunicare principal în multe procese (confirmări, facturi, oferte)
- **id_client:** Aceasta este cheia străină care leagă date_clienti de clienti. Dacă ar fi null, legătura între informațiile adiționale și client ar fi pierdută.

- Entitatea **vehicule:**

- Coloana **cod_vehicul:** Fiecare vehicul trebuie să aibă un cod unic pentru a fi identificat. Fără această valoare, vehiculele ar fi imposibil de gestionat în baza de date.

- **marca:** Marca este un atribut cheie pentru vehicule. Fără această informație, vehiculul nu ar putea fi prezentat sau clasificat corect.
- **model:** Modelul completează identificarea vehiculului. Un vehicul fără model nu ar putea fi descris adecvat clienților.
- **pret:** Prețul este crucial pentru orice tranzacție. Un vehicul fără preț nu poate fi vândut, iar lipsa acestei informații ar face ca datele să fie incomplete.
- **stare:** Este necesar să se știe dacă vehiculul este nou sau second-hand pentru transparență față de client și pentru determinarea valorii mai precisă a vehiculului.
- **Entitatea date_vehicule:**
 - **an_fabricatie:** Anul de fabricație este esențial pentru evaluarea vehiculului și determinarea valorii sale pe piață.
 - **tip_combustibil:** Tipul de combustibil este necesar pentru a clasifica vehiculele în funcție de specificații tehnice, util pentru clienți și întreținere.
 - **kilometraj:** Kilometrajul reflectă utilizarea vehiculului, o informație crucială pentru evaluarea stării tehnice și comerciale.
 - **cod_vehicul:** Codul vehiculului este cheia străină care leagă date_vehicule de vehicule. Fără acesta, legătura între entități ar fi pierdută.
- **Entitatea dotari:**
 - **id_dotare:** Identificatorul dotării trebuie să fie unic și obligatoriu, altfel legătura între vehicule și dotările lor suplimentare nu ar putea fi realizată.
 - **descriere_dotare:** Este esențial să știm ce reprezintă fiecare dotare. Fără o descriere, dotarea ar fi inutilă și ar crea confuzie.
- **Entitatea vehicul_dotari:**
 - **cod_vehicul:** Este necesar pentru a asocia dotările suplimentare cu vehiculele specifice. Fără această informație, dotările nu ar putea fi corelate cu vehiculele.
 - **id_dotare:** Identifică dotarea specifică asociată unui vehicul. Este necesar pentru a lega corect dotările cu vehiculele.

- **cost_suplimentar:** Valoarea suplimentară este esențială pentru a calcula prețul final al vehiculului.
- **Entitatea tranzactii:**
 - **id_tranzactie:** Această coloană identifică fiecare tranzacție în mod unic. Fără ea, evidența tranzacțiilor ar deveni imposibilă.
 - **data_tranzactie:** Data tranzacției este crucială pentru evidența cronologică și pentru rapoarte. Fără aceasta, nu s-ar putea urmări când au avut loc vânzările.
 - **id_client:** Este necesar să știm care client a efectuat tranzacția. Fără această informație, legătura dintre clienți și tranzacții ar fi pierdută.
- **Entitatea date_tranzactii:**
 - **pret:** Valoarea tranzacției este obligatorie pentru contabilitate și pentru generarea rapoartelor financiare.
 - **modalitate_plata:** Este necesar să se știe cum a fost efectuată plata (de exemplu, numerar, card, finanțare). Fără această informație, tranzacția nu ar putea fi înregistrată corect.
 - **cod_vehicul:** Este esențial să se știe ce vehicul este implicat în tranzacție. Fără această legătură, nu s-ar putea urmări achizițiile vehiculelor.
 - **id_tranzactie:** Identificatorul tranzacției este esențial pentru înregistrarea unică.
- **Entitatea test_drive:**
 - **id_test_drive:** Fiecare test drive trebuie să aibă un identificator unic pentru gestionarea programărilor.
 - **data_test_drive:** Este important să se știe când a fost programat test drive-ul. Fără această valoare, gestionarea rezervărilor ar fi imposibilă.
 - **cod_vehicul:** Fiecare test drive este asociat cu un vehicul specific. Fără această informație, nu s-ar putea stabili care vehicul trebuie să fie testat.
 - **id_client:** Este necesar să știm ce client a solicitat test drive-ul pentru evidență și contact.

5. Constrângeri de tip CHECK (CK)

- Entitatea **vehicule**:
 - **vehicule_pret_ck (pret >= 0)**: Asigură că toate vehiculele din tabelul vehicule au un preț valid, adică nu există valori negative în coloana pret. Fără această constrângere, un utilizator ar putea introduce accidental un preț negativ, ceea ce ar duce la date incoerente sau rapoarte incorecte.
 - **vehicule_stare_ck**: restricționează valorile permise în coloana stare la două opțiuni: nou sau second-hand. Această constrângere previne inserarea valorilor care nu au sens din punct de vedere al logicii aplicației, cum ar fi reparat sau alte descrieri inadecvate.
- Entitatea **date_vehicule**:
 - **date_veh_tip_comb_ck**: Această constrângere verifică că valorile din coloana tip_combustibil să fie doar dintre cele permise: 'Benzina', 'Electric', 'Hybrid' sau 'Motorina'. Previne inserarea unor valori invalide sau eronate, asigurând că tipul de combustibil este unul recunoscut și utilizat în practică.
 - **date_veh_km_ck (kilometraj >= 0)**: Această constrângere asigură că valoarea kilometrajului nu poate fi negativă. Este logic ca un vehicul să nu aibă un kilometraj negativ, astfel această constrângere elimină posibilitatea unor erori de introducere a datelor.
- Entitatea **date_tranzactii**:
 - **date_tranzactii_pret_ck (pret >= 0)**: Această constrângere garantează că prețul unei tranzacții nu poate fi negativ. Previne introducerea unor valori invalide care ar putea compromite raportarea financiară și analiza tranzacțiilor
 - **veh_dotari_cost_supl_ck (cost_suplimentar >= 0)**: Această constrângere verifică faptul că valoarea cost_suplimentar asociată dotărilor unui vehicul este pozitivă sau zero. Costurile suplimentare negative ar fi ilogice din punct de vedere economic, iar această regulă împiedică astfel de erori.

6. Index-uri

- **date_clienti__idx**: Acest index unic pe coloana **id_client** din tabelul **date_clienti** optimizează căutările și accesul la datele clienților. El asigură, de asemenea, că fiecare client are exact un set de date asociate, întărind legătura cu tabelul **clienti**.
- **date_tranzactii__idx**: Definit pe coloana **id_tranzactie** din tabelul **date_tranzactii**, acest index unic îmbunătățește performanța operațiilor de căutare și asigură unicitatea identificatorului fiecărei tranzacții.
- **date_tranzactii__idxv1**: Acest index unic pe coloana **cod_vehicul** din tabelul **date_tranzactii** previne asocierea unui vehicul cu mai multe tranzacții simultane, respectând regula 1:1 dintre vehicul și tranzacție.
- **date_vehicule__idx**: Creat pe coloana **cod_vehicul** din tabelul **date_vehicule**, acest index unic asigură că fiecare vehicul are un set de detalii asociate unic, facilitând gestionarea informațiilor despre vehicule.

7. Triggere

- **clienti_id_client_trg**: Acest trigger generează automat valori pentru coloana **id_client** în tabelul **clienti**, utilizând secvența **clienti_id_client_seq**. Este activat înainte de fiecare inserare de rând și se aplică doar atunci când coloana **id_client** este lăsată nulă. Această constrângere asigură unicitatea și consistența ID-urilor clienților, eliminând erorile cauzate de alocarea manuală a ID-urilor.
- **dotari_id_dotare_trg**: Acest trigger este folosit pentru a atribui automat ID-uri unice în coloana **id_dotare** din tabelul **dotari**, utilizând secvența **dotari_id_dotare_seq**. Prin activarea înainte de inserare, triggerul garantează că fiecare dotare are un identificator unic, ceea

ce facilitează gestionarea și identificarea precisă a dotărilor vehiculelor.

- Triggerul **test_drive_id_test_drive_trg**: Folosind secvența **test_drive_id_test_drive_seq**, acest trigger asigură generarea automată de ID-uri în coloana **id_test_drive** din tabelul **test_drive**. Activat înainte de inserare, acesta simplifică gestionarea test drive-urilor și previne conflictele de identificare
- **tranzactii_id_tranzactie_trg**: Acest trigger automatizează alocarea de ID-uri pentru tranzacții, utilizând secvența **tranzactii_id_tranzactie_seq**. Este activat înainte de inserarea unui nou rând în tabelul **tranzactii**, asigurând unicitatea fiecărei tranzacții și integritatea legăturilor între tranzacții și alte entități din sistem.
- Triggerul **vehicule_cod_vehicul_trg**: Prin utilizarea secvenței **vehicule_cod_vehicul_seq**, acest trigger generează automat valori unice pentru coloana **cod_vehicul** din tabelul **vehicule**. Activat înainte de inserare, triggerul elimină riscul de duplicare a codurilor vehiculelor și optimizează procesul de gestionare a acestora în sistem.

8. Autoincrement

- coloana **id_client** din tabelul **clienti**: autoincrementul este gestionat prin secvența **clienti_id_client_seq**, care începe de la valoarea 101 și nu păstrează valori în cache. Această secvență este utilizată împreună cu triggerul **clienti_id_client_trg**, care inserează automat un ID unic pentru fiecare client, eliminând necesitatea intervenției manuale.
- coloana **id_dotare** din tabelul **dotari**: secvența **dotari_id_dotare_seq**, care începe de la valoarea 501, este folosită pentru a genera automat valori unice. Triggerul **dotari_id_dotare_trg** completează procesul prin atribuirea automată a identificatorilor dotărilor vehiculelor.
- coloana **id_test_drive** din tabelul **test_drive**: autoincrementul este asigurat de secvența **test_drive_id_test_drive_seq**, configurată să

înceapă de la valoarea 10001. Această secvență este legată de triggerul **test_drive_id_test_drive_trg**, care automatizează alocarea ID-urilor pentru fiecare test drive.

- coloana **id_tranzactie** din tabelul **tranzactii**: Secvența **tranzactii_id_tranzactie_seq**, care începe de la valoarea 5001, gestionează autoincrementul coloanei **id_tranzactie** din tabelul **tranzactii**. Triggerul **tranzactii_id_tranzactie_trg** facilitează acest proces, asigurând unicitatea fiecărei tranzacții.
- Coloana **cod_vehicul** din tabelul **vehicule** beneficiază de autoincrement prin intermediul secvenței **vehicule_cod_vehicul_seq**, care pornește de la valoarea 1001. Aceasta este implementată în mod automat prin triggerul **vehicule_cod_vehicul_trg**, care atribuie coduri unice pentru fiecare vehicul adăugat în sistem.