

Proiect Baze de Date

Gestionarea unui magazin online cu articole vestimentare

Titlul proiectului

Gestionarea unui magazin online cu articole vestimentare

Descrierea proiectului

Proiectul își propune să ofere analiza, proiectarea și implementarea unei baze de date care să gestioneze work flow-ul unui magazin online cu articole vestimentare.

Proiectul permite:

- Autentificarea clientilor în conturi personale pentru plasarea comenzi.
- Gestionarea coșului de cumpărături, cu opțiuni de plată numerar și card.
- Asocierea fiecărei comenzi cu o adresă de livrare.
- Urmărirea procesului de livrare în toate etapele acestuia.
- Alegerea specificațiilor produsului (mărime, lungime, lățime).
- Gestionarea stocurilor și interacțiunilor cu curierii de către angajați.

Limitări:

- Modalități de plată disponibile: numerar sau card.
- Plata cu cardul necesită fonduri disponibile.
- Comenzi permise doar cu un cont creat prealabil.
- E-mailurile trebuie să conțină litere, cifre, „@” și punct, iar acestea trebuie să fie unice.
- Numărul de telefon trebuie să conțină doar cifre.
- E-mailurile și numerele de telefon sunt unice pentru fiecare utilizator.

Modelul bazei de date (entități, atribute)

1. Adresa:

- cod_adresa (NUMBER (7), PK, generat automat prin incrementare)
- strada (VARCHAR2 (100))
- oras (VARCHAR2 (50))
- judet (VARCHAR2 (100))
- zip_code (NUMBER (10), validat prin expresii regulate)

2. Articole vestimentare:

- cod_articol (NUMBER (10), PK, generat automat prin incrementare)
- pret (validare > 0) (NUMBER(6, 3))
- marime (valori acceptate: S, M, L, XL, XXL) (CHAR (5))
- firma (CHAR (15))
- tip (valori acceptate: hanorac, tricou, încăltăminte) (CHAR (20))

3. Clienți:

- cod_client (NUMBER(10), PK, generat automat prin incrementare)
- nume (CHAR (30))
- prenume (CHAR (30))

4. Conturi:

- email (VARCHAR2 (40), unic, validat prin expresii regulate)
- parola (VARCHAR2 (50))
- numar_telefon (VARCHAR2 (12), unic, validat prin expresii regulate)
- cod_client (NUMBER (10), FK)

5. Detalii comandă:

- cod_comanda (NUMBER (10), PK, generat automat prin incrementare))
- data_plasare (DATE, validare să fie > data curentă)
- modalitate_plata (VARCHAR2 (10), valori acceptate: numerar, card)
- cod_client (NUMBER (10), FK)

6. Detalii curier:

- cod_curier (NUMBER (10), PK, generat automat prin incrementare))
- data_ridicare (DATE, validare să fie > data curentă)
- data_predare (DATE, validare să fie > data curentă)
- cod_client (NUMBER (10), FK)
- cod_comanda (NUMBER (10), FK compus)

7. Clienti-articole vestimentare (relație N:M):

- clienti_cod_client (NUMBER (10), PK, FK, generat automat prin incrementare)
- ar_ve_cod_articol (NUMBER (10), PK, FK, generat automat prin incrementare)

Relații între entități și integrări

- **1:M:** Clienti → Conturi, Clienti → Detalii comandă, Detalii comandă → Detalii curier.
- **N:M:** Clienti ↔ Articole vestimentare printr-o entitate de legătură (clienti_ar_ve_fk).

Diagrama logică

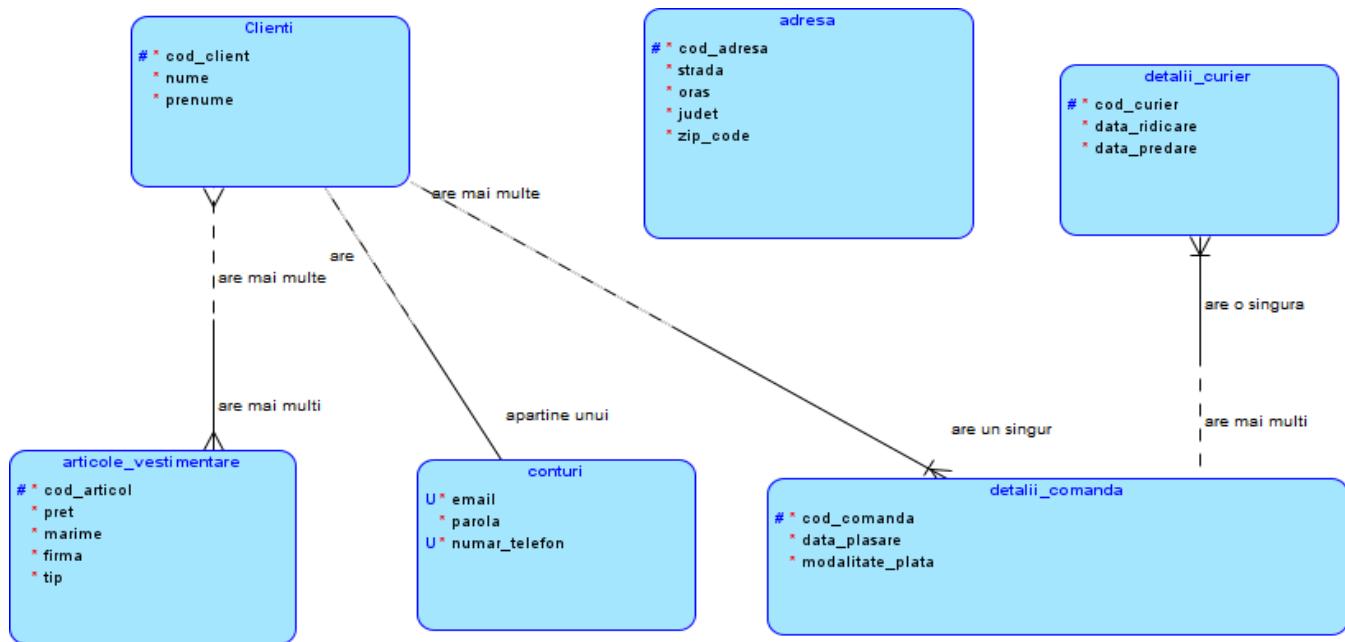
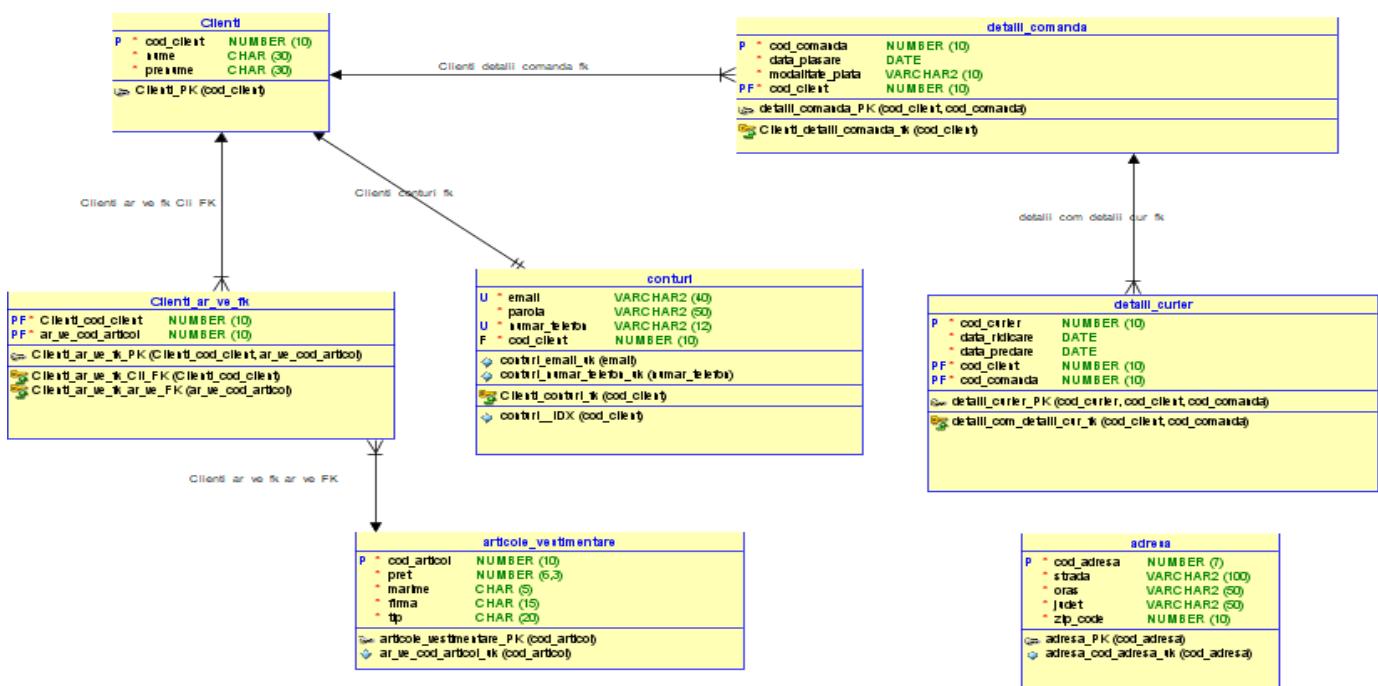


Diagrama relatională



Descrierea a constrângerilor utilizate

1. Constrângeri de cheie primară (PK - Primary Key):

- **Descriere:** Asigură unicitatea fiecărui rând dintr-un tabel și identificarea clară a acestuia.
- **Exemple:**
 - cod_client în tabela **Clienti**.
 - cod_articol în tabela **Articole vestimentare**.
 - cod_comanda în tabela **Detalii comandă**.
- **Utilitate:**
 - Previne duplicarea înregistrărilor.
 - Permite relaționarea corectă între tabele prin chei externe.

2. Constrângeri de cheie externă (FK - Foreign Key):

- **Descriere:** Menține legătura dintre entități, asigurând consistența referințelor.
- **Exemple:**
 - cod_client în tabela **Conturi** referință la tabela **Clienti**.
 - cod_comanda în tabela **Detalii curier** referință la tabela **Detalii comandă**.
- **Utilitate:**
 - Asigură existența înregistrărilor referite înainte de a fi utilizate.
 - Previne ștergerea datelor legate în mod incorrect.

3. Constrângeri de unicitate (UK - Unique Key):

- **Descriere:** Asigură unicitatea valorilor dintr-o coloană sau un grup de coloane.
- **Exemple:**
 - email în tabela **Conturi**.
 - numar_telefon în tabela **Conturi**.
- **Utilitate:**
 - Previne înregistrările duplicate, de exemplu, crearea de conturi multiple cu același email sau număr de telefon.

4. Constrângeri de verificare (CK - Check Constraint):

- **Descriere:** Validează datele introduse în coloane, conform unor reguli definite.
- **Exemple:**
 - pret în tabela **Articole vestimentare** are constrângerea > 0 .
 - modalitate_plata în tabela **Detalii comandă** trebuie să fie numerar sau card.
 - marime în tabela **Articole vestimentare** este limitată la valori predefinite (S, M, L, XL, XXL).
- **Utilitate:**
 - Previne introducerea de date incorecte sau nerealiste.
 - Simplifică validările la nivel de aplicație.

5. Constrângeri Not Null (NN):

- **Descriere:** Obligatoriu pentru coloane care nu pot conține valori lipsă.
- **Exemple:**
 - nume și prenume în tabela **Clienți**.
 - parola în tabela **Conturi**.
- **Utilitate:**
 - Asigură completitudinea datelor esențiale.

6. Domenii predefinite (Domain Constraints):

- **Descriere:** Limitează valorile acceptate la o listă de opțiuni.
- **Exemple:**
 - marime și tip în tabela **Articole vestimentare**.
 - modalitate_plata în tabela **Detalii comandă**.
- **Utilitate:**
 - Simplifică verificarea datelor și menține consistența valorilor.