

Universitatea din București

Facultatea de Matematică și Informatică

Specializarea Calculatoare și Tehnologia Informației

Proiect

Baze de Date

Profesor coordonator:

Vasile Silviu Laurențiu

Student:

Soisun Mina-David

Universitatea din București

Facultatea de Matematică și Informatică

Specializarea Calculatoare și Tehnologia Informației

Managementul unei Farmacii

Profesor coordonator:

Vasile Silviu Laurențiu

Student:

Soisun Mina-David

Cuprins

1. Introducere	4
1.1. Motivația alegerii lucrării	4
1.2. Software folosit	4
1.3. Prezentarea modelului	4
1.4. Prezentarea regulilor modelului	5
2. Diagrama entitate-relație	6
2.1. Descrierea entităților	7
2.2. Descrierea relațiilor și cardinalităților	11
3. Diagrama conceptuală	13
3.1. Descrierea constrângerilor de integritate	14
3.2. Schemele relaționale	16
4. Implementare	17
4.1. Crearea tabelelor	17
4.2. Inserarea în tabele	22
4.3. Ștergerea tabelelor	31
5. Concluzie	32
6. Bibliografie	32

1. Introducere

1.1. Motivația alegerii lucrării

Am ales această temă datorită conexiunii mele cu mediul medical, în special cel farmaceutic, mamei mele lucrând în domeniu. Am vrut să mă diferențiez de celelalte tipuri de magazine „standard” prin introducerea conceptului de rețetă și de echipamente în schema bazei mele de date.

1.2. Software folosit

Ca SGBD (Sistem de gestiune a bazelor de date) am folosit MySQL Workbench datorită ușurinței configurării server-ului local pentru implementarea efectivă a bazei de date. Programul oferă o interfață prietenoasă pentru dezvoltarea și interacțiunea cu bazele de date, în același timp funcționalitatea de **Reverse Engineering** oferă un mod foarte ușor pentru a realiza **diagrama conceptuală**.

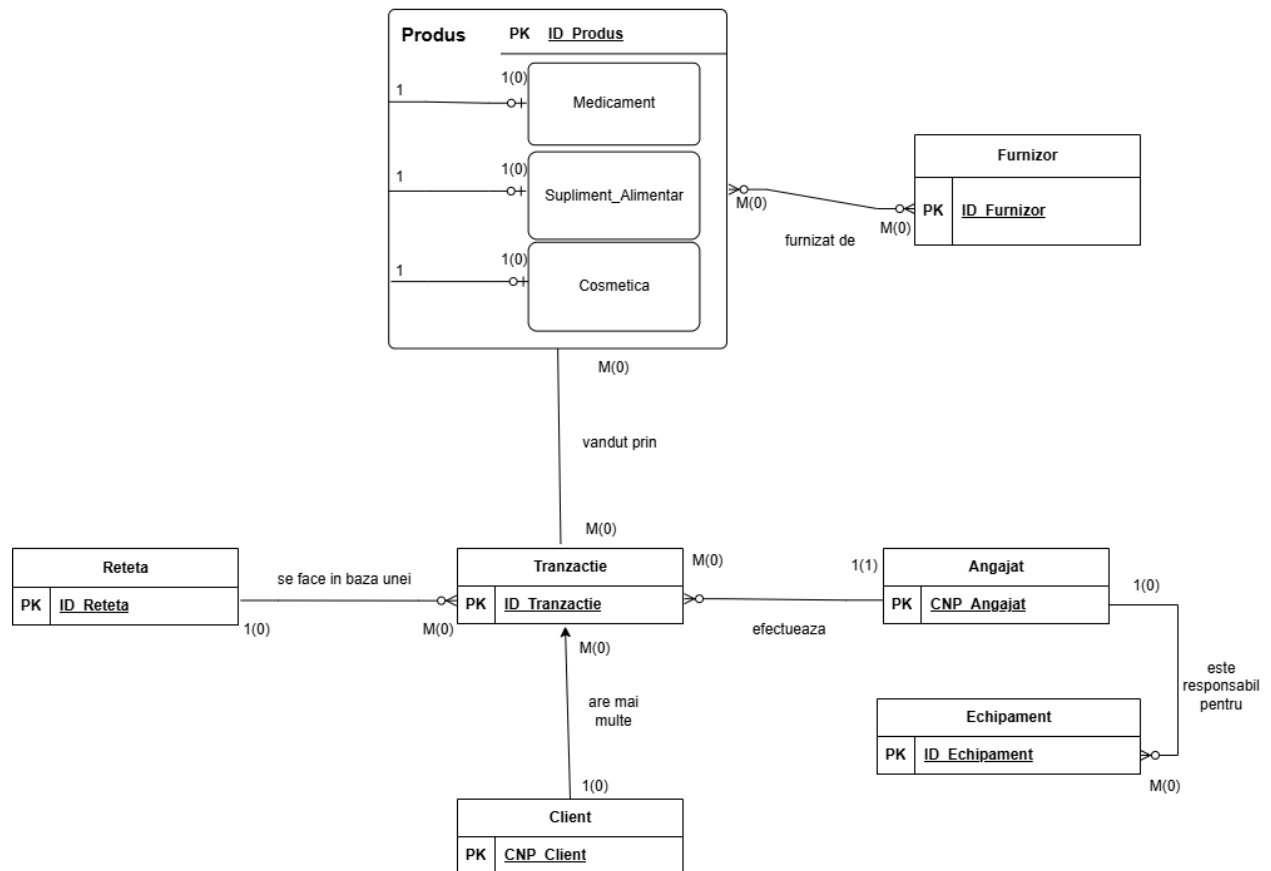
1.3. Prezentarea modelului

În acest model, farmacia comercializează mai multe tipuri de produse: medicamente OTC (over-the-counter), cele care se eliberează fara rețetă, medicamente cu substanțe active reglementate de lege, deci cele care se eliberează cu rețetă, suplimente alimentare și produse cosmetice. Mi-am propus sa modelez tranzacționarea efectivă de produse, sistemul de eliberare prin rețete dar și să integrez intervalele de mentenanță ale echipamentelor.

1.4. Prezentarea regulilor modelului

1. Entitatea principală o constituie Produsul, care servește drept catalog general, dar care fizic aparține unui Stoc (loturi), care la rândul său provine de la un Furnizor.
2. Produsele se încadrează într-o categorie specifică (Medicament, Supliment Alimentar sau Cosmetic) pentru a facilita gestionarea atributelor unice (precum substanța activă pentru medicamente sau zona de aplicare pentru cosmetice).
3. Un Client poate alege să își asocieze identitatea cu o tranzacție (pentru fidelitate), dar sistemul permite și tranzacții anonime (unde câmpul client este NULL).
4. Angajații pot dispune de un Echipament (ex: cântar, calculator), fiind direct responsabili pentru acesta prin asocierea CNP-ului lor cu echipamentul respectiv.
5. Tranzacțiile care nu implică medicamente cu regim special vor avea în câmpul ID_Reteta valoarea NULL; în caz contrar, vânzarea se face pe baza unei rețete prescrise de un medic.

2. Diagrama entitate-relație



2.1. Descrierea entităților

1. Angajat: Conține informații despre angajații farmaciei

Atribut	Tip de date	Descriere
CNP_Angajat	BIGINT	PK CNP-ul angajatului
Nume	VARCHAR(25)	Numele de familie al angajatului
Prenume	VARCHAR(25)	Prenumele angajatului
Funcție	VARCHAR(50)	Funcția pe care o ocupă angajatul
Data_angajarii	DATE	Data la care a fost angajat
Telefon	VARCHAR(13)	Numărul de telefon al angajatului

2. Echipament: Conține informații despre echipamentele din cadrul farmaciei

Atribut	Tip de date	Descriere
ID_Echipament	INT	PK Identificator unic al echipamentului
CNP_Angajat_responsabil	BIGINT	FK Realizează asocierea cu tabelul Angajat
Nume Echipament	VARCHAR(25)	Numele echipamentului
Serie	VARCHAR(25)	Codul de serie al echipamentului
Data_Revizie	DATE	Data la care trebuie făcută următoarea revizie tehnică a echipamentului

3. Client: Conține informații despre clienții înscriși în programul de fidelitate al farmaciei

Atribut	Tip de Date	Descriere
CNP_Client	BIGINT	PK CNP-ul clientului
Nume	VARCHAR(25)	Numele de familie al clientului
Prenume	VARCHAR(25)	Prenumele clientului
Telefon	VARCHAR(13)	Numărul de telefon al clientului
Data_inscriere	DATE	Data înscrierii în programul de fidelitate
Serie_card	BIGINT	Seria cardului fizic

4. Furnizor: Conține informații despre furnizori

Atribut	Tip de Date	Descriere
ID_Furnizor	INT	PK Identificator unic al furnizorului
Nume	VARCHAR(50)	Numele Furnizorului
Adresa	VARCHAR(100)	Adresa sediului furnizorului
Telefon	VARCHAR(13)	Numărul de telefon al furnizorului
Persoana_contact	VARCHAR(50)	Numele si prenumele persoanei de contact al furnizorului

5. Produs: Conține informații despre produse

Atribut	Tip de Date	Descriere
ID_Produs	INT	PK Identificator unic al produsului
Nume_Produs	VARCHAR(50)	Numele produsului

Pret_unit_achizitie	DECIMAL(5,2)	Prețul la care a fost achiziționat produsul de la furnizor
Pret_unit_vanzare	DECIMAL(5,2)	Prețul cu care este vândut produsul clienților

6. Medicament: Conține informații despre medicamentele vândute în farmacie

Atribut	Tip de Date	Descriere
ID_Produs	INT	PK, FK Face legătura dintre subtabel și tabelul părinte
Substanta_activa	VARCHAR(50)	Substanța activă a medicamentului
Concentratie	VARCHAR(10)	Concentrația și unitatea de măsură a substanței active din medicament
Forma_farmaceutica	VARCHAR(20)	Forma fizică a medicamentului
Necesita_rețetă	CHAR(1)	Condiția de eliberare în baza unei rețete

7. Supliment_alimentar: Conține informații despre suplimentele alimentare vândute în farmacie

Atribut	Tip de Date	Descriere
ID_Produs	INT	PK, FK Face legătura dintre subtabel și tabelul părinte
Beneficiu_principal	VARCHAR(50)	Aria sănătății îmbunătățită de supliment
Forma_prezentare	VARCHAR(30)	Forma sub care se prezintă produsul
Public_tinta	VARCHAR(50)	Publicul care este intenționat să consume suplimentul

8. Cosmetic: Conține informații despre cosmeticele vândute în farmacie

Atribut	Tip de Date	Descriere
ID_Produs	INT	PK, FK Face legătura dintre subtabel și tabelul părinte
Functie_principala	VARCHAR(30)	Scopul principal pe care îl îndeplinește produsul
Cantitate	VARCHAR(20)	Cantitatea si unitatea de măsură a produsului

9. Rețetă: Conține informații despre rețete

Atribut	Tip de Date	Descriere
ID_Reteta	INT	PK Identificator unic al rețetei
Medic_prescriptor	VARCHAR(70)	Numele și prenumele medicului care a prescris rețeta
Data_prescriere	DATE	Data la care a fost eliberata rețeta

10. Tranzacție: Conține informații despre tranzacțiile efectuate în farmacie

Atribut	Tip de Date	Descriere
ID_Tranzactie	INT	PK Identificator unic al tranzacției
Data_Ora_Tranzactie	DATE	Data si ora la care a fost făcută tranzacția
CNP_Angajat	NUMBER(13)	FK Face legătura cu tabelul Angajat
CNP_Client	BIGINT	FK Face legătura cu tabelul Client
ID_Reteta	INT	FK Face legătura cu tabelul rețeta

2.2. Descrierea relațiilor și cardinalităților

Produs – Furnizor $M(0) - M(0)$

Un produs poate fi furnizat de **mai mulți** furnizori.

Un produs poate să nu fie furnizat de **niciun** furnizor

Un furnizor poate furniza **mai multe** produse.

Un furnizor poate să nu furnizeze **niciun** produs.

Produs – Tranzacție $M(0) - M(0)$

Un produs poate să apară în **mai multe** tranzacții

Un produs poate să nu apară în **nicio** tranzacție

O tranzacție poate să conțină **mai multe** produse

O tranzacție poate să nu conțină **niciun** produs

Rețetă – Tranzacție $1(0) - M(0)$

O tranzacție poate fi făcută în baza **unei** rețete

O tranzacție poate să **nu** fie făcută în baza **unei** rețete

O rețetă poate să apară în **mai multe** tranzacții

O rețetă poate să nu apară în **nicio** tranzacție

Angajat – Tranzacție $1(1) - M(0)$

Un angajat poate să efectueze **mai multe** tranzacții

Un angajat poate să nu efectueze **nicio** tranzacție

O tranzacție este efectuată de **minim un** angajat

O tranzacție este efectuată de **maxim un** angajat

Client – Tranzacție $1(0) - M(0)$

Un client poate să efectueze **mai multe** tranzacții

Un client poate să nu efectueze **nicio** tranzacție

O tranzacție este efectuată de **minim un** client

O tranzacție poate să nu fie efectuată cu **niciun** client

Angajat – Echipament $1(0)-M(0)$

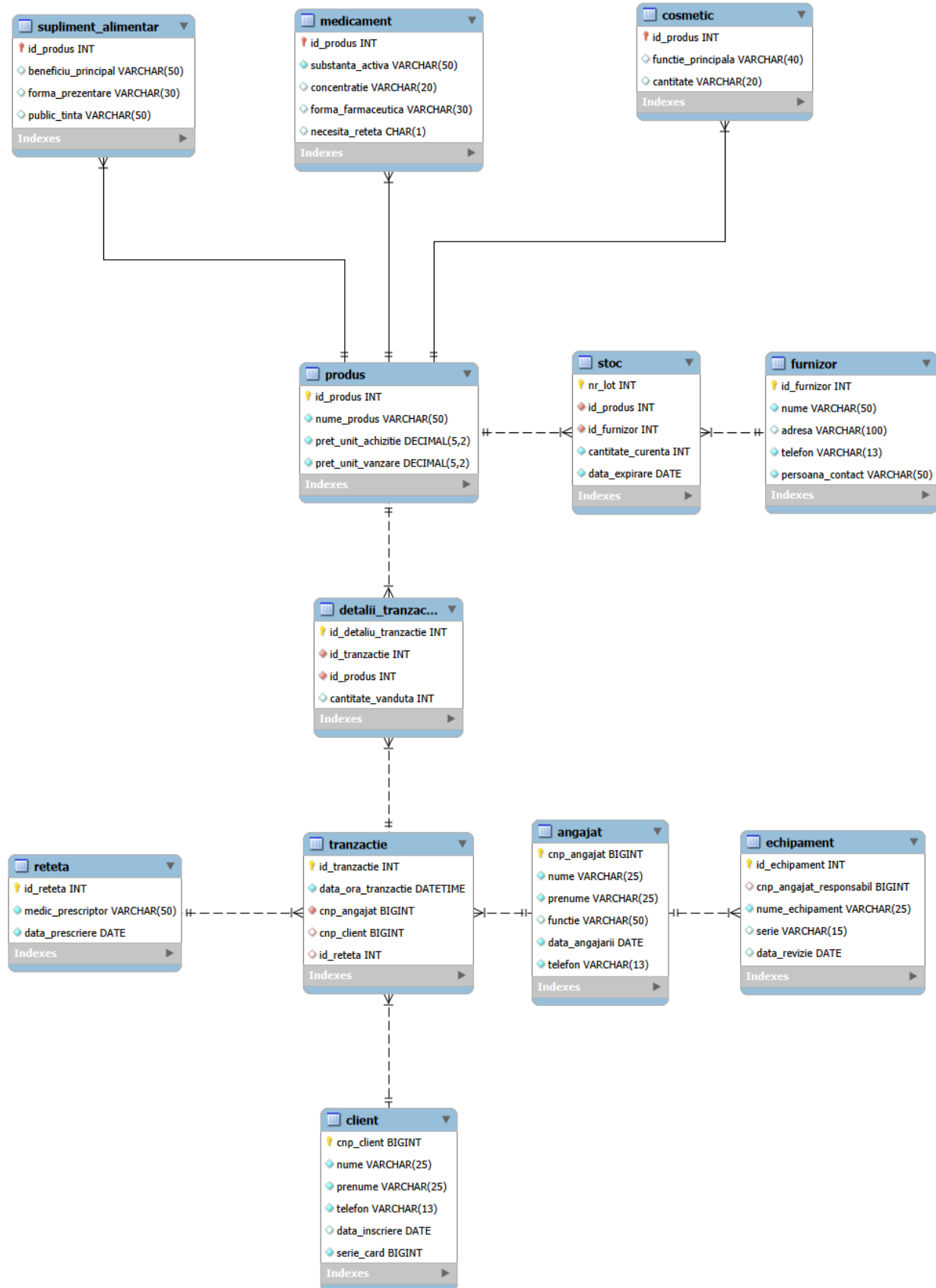
Un angajat poate fi responsabil pentru **mai multe** echipamente

Un angajat poate să nu fie responsabil pentru **niciun** echipament

Un echipament poate să nu fie responsabilitatea **niciunui** angajat

Un echipament este responsabilitatea a **maxim unui** angajat

3. Diagrama conceptuală



3.1. Descrierea constrângerilor de integritate

1. Reguli Angajat

- CNP_Angajat: Primary Key, check(length(cnp_angajat)=13 AND substr(cnp_angajat,1,1) in (1,2,5,6))
- Nume: not null
- Prenume: not null
- Functie: check in ('asistent farmacist', 'farmacist', 'chimist', 'contabil', 'curatenie')
- Data_angajarii: not null
- Telefon: not null

2. Reguli Echipament

- ID_Echipament: Primary Key
- CNP_Angajat_Responsabil: foreign key către Angajat(CNP_Angajat)
- Nume_Echipament: not null
- Serie: unique

3. Reguli Client:

- CNP: Primary key și check(length(cnp_client)=13 AND substr(cnp_client,1,1) in (1,2,5,6))
- Nume: not null
- Prenume: not null
- Telefon: not null
- Serie_card : not null si unique

4. Reguli Furnizor:

- ID_Furnizor: primary key
- Nume: not null
- Telefon: not null
- Persoana_contact: not null

5. Reguli Produs:

- ID_Produs: primary key
- Nume: Not null
- Pret_unit_achizitie: Not NULL și check (pret_unit_achizitie > 0)
- Pret_unit_vanzare: Not NULL și check (pret_unit_vanzare > 0)

6. Reguli Medicament:

- ID_Produs: Primary Key și foreign key către Produs(ID_Produs)
- Substanta_activa: not null
- Necesita_reteta: Default ,n' check (necesita_reteta in (,y', ,n'))

7. Reguli Supliment alimentar:

- ID_Produs: Primary Key și foreign key către Produs(ID_Produs)
- Beneficiu_principal: Check (beneficiu_principal in ('Imunitate', 'Digestie', 'Articulatii', 'Energie', 'General'))

8. Reguli Cosmetic:

- ID_Produs: Primary Key și foreign key către Produs(ID_Produs)

9. Reguli Reteta:

- ID_Produs: Primary key
- Medic_prescriptor: not null
- Data_prescriere: not null

10.Reguli Tranzactie:

- ID_Tranzactie: Primary Key
- Data_Tranzactie: not null
- CNP_Angajat: Not Null si Foreign Key către Angajat(CNP_Angajat)
- CNP_Client: Foreign Key către Client(CNP_Client)
- ID_Reteta: Foreign key către Reteta(ID_Reteta)

11.Reguli Detalii_Tranzactie

- ID_Detaliu_Tranzactie: Primary Key
- ID_Tranzactie: Not null și Foreign Key către Tranzactie(ID_Tranzactie)
- ID_Produs: Not Null și Foreign Key către Produs(ID_Produs)
- Cantitate_vanduta: Check (cantitate_vanduta) > 0

12.Reguli Stoc:

- Nr_lot: Primary Key
- ID_Produs: Not Null și Foreign Key către Produs(ID_Produs)
- ID_Furnizor: Not Null și Foreign Key către Furnizor(ID_Furnizor)
- Cantitate_curenta: Not Null și Check (cantitate_curenta >=0)
- Data_Expirare: Not Null

3.2. Schemele relaționale

ANGAJAT(cnp_angajat#, nume, prenume, functie, data_angajarii, telefon)

ECHIPAMENT(id_echipament#, cnp_angajat_responsabil[FK], nume_echipament, serie, data_revizie)

CLIENT(cnp_client#, nume, prenume, telefon, data_inscriere, serie_card)

FURNIZOR(id_furnizor#, nume, adresa, telefon, persoana_contact)

PRODUS(id_produs#, nume_produs, pret_unit_achizitie, pret_unit_vanzare)

MEDICAMENT(id_produs#[FK], substanta_activa, concentratie, forma_farmaceutica, necesita_reteta)

SUPLIMENT_ALIMENTAR(id_produs#[FK], beneficiu_principal, forma_prezentare, public_tinta)

COSMETIC(id_produs# [FK], functie_principala, cantitate)

RETETA (id_reteta#, medic_prescriptor, data_prescriere)

TRANZACTIE(id_tranzactie#, data_tranzactie, cnp_angajat[FK], cnp_client [FK], id_reteta [FK])

DETALII_TRANZACTIE(id_detaliu_tranzactie#, id_tranzactie[FK], id_produs [FK], cantitate_vanduta)

STOC(nr_lot#, id_produs [FK], id_furnizor [FK], cantitate_curenta, data_expirare)

4. Implementare

4.1. Crearea tabelelor

```
create table angajat (  
    cnp_angajat bigint primary key,  
    nume varchar(25) not null,  
    prenume varchar(25) not null,  
    functie varchar(50),  
    data_angajarii date not null,  
    telefon varchar(13) not null,  
  
    constraint ang_functie_ck check (functie in ('asistent farmacist',  
    'farmacist', 'contabil', 'chimist', 'inginer chimist', 'curatenie')),  
    constraint ang_cnp_ck check(length(cnp_angajat)=13 and  
    substr(cnp_angajat,1,1) in ('1','2','5','6'))  
);  
  
create table echipament(  
    id echipament int primary key,  
    cnp_angajat_responsabil bigint,  
    nume echipament varchar(25) not null,  
    serie varchar(15),  
    data_revizie date,  
    constraint ech_serie_uk unique (serie),  
    constraint fk_ang_echip foreign key (cnp_angajat_responsabil)  
references angajat(cnp_angajat) on delete cascade on update cascade  
);
```

```

create table client(
    cnp_client bigint primary key,
    nume varchar(25) not null,
    prenume varchar(25) not null,
    telefon varchar(13) not null,
    data_inscriere date,
    serie_card bigint not null,
    constraint cl_cnp_ck check(length(cnp_client)=13 and
substr(cnp_client,1,1) in ('1','2','5','6')),
    constraint cl_serie_uk unique (serie_card)
);

create table furnizor(
    id_furnizor int primary key,
    nume varchar(50) not null,
    adresa varchar(100),
    telefon varchar(13) not null,
    persoana_contact varchar(50) not null
);

create table produs(
    id_produs int primary key,
    nume_produs varchar(50) not null,
    pret_unit_achizitie decimal(5,2) not null,
    pret_unit_vanzare decimal(5,2) not null,
    check (pret_unit_achizitie > 0),
    check (pret_unit_vanzare > 0)
);

```

```

create table medicament(
    id_produș int primary key,
    substanta_activa varchar(50) not null,
    concentratie varchar(20),
    forma_farmaceutica varchar(30),
    necesita_reteta char(1) default 'n' check (necesita_reteta in ('y',
'n')),

    constraint fk_med_prod foreign key (id_produș) references
produș(id_produș) on delete cascade on update cascade
);

create table supliment_alimentar(
    id_produș int primary key,
    beneficiu_principal varchar(50) not null,
    forma_prezentare varchar(30),
    public_tinta varchar(50),

    constraint fk_sup_prod foreign key (id_produș) references
produș(id_produș) on delete cascade on update cascade

    constraint ck_beneficiu check (beneficiu_principal in ('Imunitate',
'Digestie', 'Articulații', 'Energie', 'General'))
);

create table cosmetic(
    id_produș int primary key,
    funcție_principală varchar(40),
    cantitate varchar(20),

```

```

        constraint fk_cosm_prod foreign key (id_produs) references
        produs(id_produs) on delete cascade on update cascade
    );

create table reteta(
    id_reteta int primary key,
    medic_prescriptor varchar(50) not null,
    data_prescriere date not null
);

create table tranzactie(
    id_tranzactie int primary key,
    data_tranzactie date not null,
    cnp_angajat bigint not null,
    cnp_client bigint,
    id_reteta int,

    constraint fk_tr_ang foreign key (cnp_angajat) references
    angajat(cnp_angajat) on delete cascade on update cascade,
    constraint fk_tr_cl foreign key (cnp_client) references
    client(cnp_client) on delete cascade on update cascade,
    constraint fk_tr_ret foreign key (id_reteta) references
    reteta(id_reteta) on delete cascade on update cascade
);

create table detalii_tranzactie (
    id_detaliu_tranzactie int primary key,
    id_tranzactie int not null,
    id_produs int not null,
    cantitate_vanduta int check (cantitate_vanduta > 0),

```

```

        constraint fk_det_tr foreign key (id_tranzactie) references
tranzactie(id_tranzactie) on delete cascade on update cascade,

        constraint fk_det_prod foreign key (id_produs) references
produs(id_produs) on delete cascade on update cascade
);

create table stoc(
    nr_lot int primary key,
    id_produs int not null,
    id_furnizor int not null,
    cantitate_curenta int not null check (cantitate_curenta >= 0),
    data_expirare date not null,

    constraint fk_stoc_prod foreign key (id_produs) references
produs(id_produs) on delete cascade on update cascade,

    constraint fk_stoc_furn foreign key (id_furnizor) references
furnizor(id_furnizor) on delete cascade on update cascade
);

```

4.2. Inserarea în tabele

-- 1. Angajat

```
insert into angajat values (1800101111111, 'Popescu', 'Ion',  
'farmacist', to_date('01-01-2020','dd-mm-yyyy'), '0722111111');  
  
insert into angajat values (1850505222222, 'Ionescu', 'Maria',  
'asistent farmacist', to_date('15-03-2021','dd-mm-yyyy'),  
'0722222222');  
  
insert into angajat values (1901010333333, 'Georgescu', 'Vlad',  
'contabil', to_date('20-05-2019','dd-mm-yyyy'), '0722333333');  
  
insert into angajat values (2951212444444, 'Dumitrescu', 'Ana',  
'farmacist', to_date('01-08-2022','dd-mm-yyyy'), '0722444444');  
  
insert into angajat values (1880808555555, 'Radu', 'Mihai', 'chimist',  
to_date('10-02-2023','dd-mm-yyyy'), '0722555555');  
  
insert into angajat values (2990101666666, 'Stan', 'Elena',  
'curatenie', to_date('01-01-2024','dd-mm-yyyy'), '0722666666');  
  
insert into angajat values (1750505777777, 'Matei', 'Alexandru',  
'farmacist', to_date('15-11-2018','dd-mm-yyyy'), '0722777777');  
  
insert into angajat values (2800303888888, 'Nistor', 'Cristina',  
'asistent farmacist', to_date('20-09-2020','dd-mm-yyyy'),  
'0722888888');  
  
insert into angajat values (1920404999999, 'Florea', 'George',  
'farmacist', to_date('05-05-2021','dd-mm-yyyy'), '0722999999');  
  
insert into angajat values (2850606000000, 'Dima', 'Simona', 'asistent  
farmacist', to_date('12-12-2022','dd-mm-yyyy'), '0722000000');
```

-- 2. Client

```
insert into client values (1700101111111, 'Avram', 'Dan',
'0744111111', str_to_date('01-01-2023', '%d-%m-%Y'), 18824);

insert into client values (2800202222222, 'Barbu', 'Alina',
'0744222222', str_to_date('15-02-2023', '%d-%m-%Y'), 29546);

insert into client values (1900303333333, 'Costea', 'Paul',
'0744333333', str_to_date('20-03-2023', '%d-%m-%Y'), 46536);

insert into client values (2950404444444, 'Diaconu', 'Ioana',
'0744444444', str_to_date('10-04-2023', '%d-%m-%Y'), 67415);

insert into client values (1850505555555, 'Enache', 'Sorin',
'0744555555', str_to_date('05-05-2023', '%d-%m-%Y'), 52998);

insert into client values (2880606666666, 'Filip', 'Carmen',
'0744666666', str_to_date('12-06-2023', '%d-%m-%Y'), 66473);

insert into client values (1780707777777, 'Grigore', 'Victor',
'0744777777', str_to_date('25-07-2023', '%d-%m-%Y'), 91398);

insert into client values (2920808888888, 'Hristea', 'Monica',
'0744888888', str_to_date('30-08-2023', '%d-%m-%Y'), 12206);

insert into client values (1960909999999, 'Iacob', 'Rares',
'0744999999', str_to_date('15-09-2023', '%d-%m-%Y'), 91478);

insert into client values (2991010000000, 'Luca', 'Diana',
'0744000000', str_to_date('01-10-2023', '%d-%m-%Y'), 4341);

insert into client values (1880808999111, 'Vasile', 'Bogat',
'0799888777', str_to_date('01-11-2023', '%d-%m-%Y'), 99101);

insert into client values (2900909888222, 'Ana', 'Cumparatoarea',
'0766555444', str_to_date('05-11-2023', '%d-%m-%Y'), 99102);--
```

3. Furnizor

```
insert into furnizor values (10, 'Farmaceutica SA', 'Bucuresti',
'0212000001', 'Ion Director');

insert into furnizor values (20, 'BioHealth SRL', 'Cluj',
'0264000002', 'Maria Manager');

insert into furnizor values (30, 'MediPlus', 'Timisoara',
'0256000003', 'Andrei Vanzari');
```

```

insert into furnizor values (40, 'PharmaLine', 'Iasi', '0232000004',
'Elena Contact');

insert into furnizor values (50, 'Sanatate Totala', 'Constanta',
'0241000005', 'George Agent');

insert into furnizor values (60, 'Natural Plant', 'Brasov',
'0268000006', 'Ana Logistic');

insert into furnizor values (70, 'EuroPharm', 'Oradea', '0259000007',
'Mihai Depozit');

insert into furnizor values (80, 'Global Med', 'Craiova',
'0251000008', 'Cristina Sef');

insert into furnizor values (90, 'VitaCare', 'Galati', '0236000009',
'Dan Distributie');

insert into furnizor values (100, 'DermoPro', 'Sibiu', '0269000010',
'Simona Relatii');

```

-- 4. Reteta

```

insert into reteta values (1001, 'Dr. Ionescu', to_date('01-11-
2023','dd-mm-yyyy'));

insert into reteta values (1002, 'Dr. Popa', to_date('02-11-2023','dd-
mm-yyyy'));

insert into reteta values (1003, 'Dr. Stan', to_date('03-11-2023','dd-
mm-yyyy'));

insert into reteta values (1004, 'Dr. Marin', to_date('04-11-
2023','dd-mm-yyyy'));

insert into reteta values (1005, 'Dr. Dumitru', to_date('05-11-
2023','dd-mm-yyyy'));

insert into reteta values (1006, 'Dr. Vasilescu', to_date('06-11-
2023','dd-mm-yyyy'));

insert into reteta values (1007, 'Dr. Gheorghe', to_date('07-11-
2023','dd-mm-yyyy'));

insert into reteta values (1008, 'Dr. Lazar', to_date('08-11-
2023','dd-mm-yyyy'));

```



```
insert into reteta values (1009, 'Dr. Munteanu', to_date('09-11-2023','dd-mm-yyyy'));
```

```
insert into reteta values (1010, 'Dr. Urs', to_date('10-11-2023','dd-mm-yyyy'));
```

-- 5. Echipament

```
insert into echipament values (1, 1800101111111, 'Casa Marcat 1', 'SN001', to_date('01-01-2023','dd-mm-yyyy'));
```

```
insert into echipament values (2, 1850505222222, 'Casa Marcat 2', 'SN002', to_date('01-01-2023','dd-mm-yyyy'));
```

```
insert into echipament values (3, 1901010333333, 'Laptop Contabilitate', 'SN003', to_date('01-06-2023','dd-mm-yyyy'));
```

```
insert into echipament values (4, 1880808555555, 'Microscop', 'SN004', to_date('15-02-2023','dd-mm-yyyy'));
```

```
insert into echipament values (5, 2990101666666, 'Aspirator Industrial', 'SN005', to_date('10-01-2024','dd-mm-yyyy'));
```

```
insert into echipament values (6, 1750505777777, 'Monitor Prescriptii', 'SN006', to_date('20-03-2023','dd-mm-yyyy'));
```

```
insert into echipament values (7, 2800303888888, 'Scanner Coduri', 'SN007', to_date('05-04-2023','dd-mm-yyyy'));
```

```
insert into echipament values (8, 1920404999999, 'Imprimanta Retete', 'SN008', to_date('12-05-2023','dd-mm-yyyy'));
```

```
insert into echipament values (9, 2850606000000, 'cantar precizie', 'SN009', to_date('30-06-2023','dd-mm-yyyy'));
```

```
insert into echipament values (10, 2951212444444, 'Frigider Medicamente', 'SN010', to_date('01-07-2023','dd-mm-yyyy'));
```

-- 6. Produs

-- Medicamente

```
insert into produs values (101, 'Paracetamol', 5.00, 10.00);
```

```
insert into produs values (102, 'Ibuprofen', 8.00, 15.00);
```

```
insert into produs values (103, 'Aspirina', 6.00, 12.00);
```

```
insert into produs values (104, 'Amoxicilina', 15.00, 30.00);
insert into produs values (105, 'Algocalmin', 7.00, 14.00);
insert into produs values (106, 'Nurofen', 12.00, 24.00);
insert into produs values (107, 'Augmentin', 20.00, 45.00);
insert into produs values (108, 'Coldrex', 18.00, 35.00);
insert into produs values (109, 'Fervex', 16.00, 32.00);
insert into produs values (110, 'No-Spa', 14.00, 28.00);
insert into produs values (111, 'Sirop Tuse', 15.00, 30.00);
```

-- Suplimente

```
insert into produs values (201, 'Vitamina C', 10.00, 20.00);
insert into produs values (202, 'Magneziu', 15.00, 30.00);
insert into produs values (203, 'Calciu', 12.00, 25.00);
insert into produs values (204, 'Omega 3', 30.00, 60.00);
insert into produs values (205, 'Probiotice', 25.00, 50.00);
insert into produs values (206, 'Zinc', 18.00, 36.00);
insert into produs values (207, 'Vitamina D3', 22.00, 44.00);
insert into produs values (208, 'Fier', 20.00, 40.00);
insert into produs values (209, 'Multivitamine', 35.00, 70.00);
insert into produs values (210, 'Colagen', 50.00, 100.00);
```

-- Cosmetice

```
insert into produs values (301, 'Crema Fata', 40.00, 80.00);
insert into produs values (302, 'Lotiune Corp', 30.00, 60.00);
insert into produs values (303, 'Sampon', 25.00, 50.00);
insert into produs values (304, 'Balsam', 25.00, 50.00);
insert into produs values (305, 'Apa Micelara', 20.00, 40.00);
```

```
insert into produs values (306, 'Ser Fata', 55.00, 110.00);
insert into produs values (307, 'Gel Curatare', 35.00, 70.00);
insert into produs values (308, 'Crema maini', 15.00, 30.00);
insert into produs values (309, 'Protectie Solara', 45.00, 90.00);
insert into produs values (310, 'Deodorant', 12.00, 24.00);
insert into produs values (311, 'Crema Premium', 60.00, 120.00);
```

-- 7. Medicament

```
insert into medicament values (101, 'Paracetamol', '500mg',
'Comprimat', 'n');
insert into medicament values (102, 'Ibuprofen', '400mg', 'Drajeuri',
'n');
insert into medicament values (103, 'Acid Acetilsalicilic', '500mg',
'Comprimat', 'n');
insert into medicament values (104, 'Amoxicilina', '500mg', 'Capsule',
'y');
insert into medicament values (105, 'Metamizol', '500mg',
'Comprimat', 'y');
insert into medicament values (106, 'Ibuprofen', '200mg', 'Sirop',
'n');
insert into medicament values (107, 'Amoxicilina+Acid', '875mg',
'Comprimat', 'y');
insert into medicament values (108, 'Paracetamol+VitC', '500mg',
'Plicuri', 'n');
insert into medicament values (109, 'Paracetamol+Fenir', '500mg',
'Plicuri', 'n');
insert into medicament values (110, 'Drotaverina', '40mg',
'Comprimat', 'n');
insert into medicament values (111, 'Codeina', '100ml', 'Sirop', 'n');
```

-- 8. Supliment_alimentar

```
insert into supliment_alimentar values (201, 'Imunitate',  
'Comprimat', 'General');  
  
insert into supliment_alimentar values (202, 'Energie',  
'Efervescent', 'Adulti');  
  
insert into supliment_alimentar values (203, 'Articulatii',  
'Comprimat', 'Batrani');  
  
insert into supliment_alimentar values (204, 'General', 'Capsule',  
'Adulti');  
  
insert into supliment_alimentar values (205, 'Digestie', 'Plicuri',  
'General');  
  
insert into supliment_alimentar values (206, 'Imunitate', 'Capsule',  
'Copii');  
  
insert into supliment_alimentar values (207, 'Imunitate', 'Picaturi',  
'Copii');  
  
insert into supliment_alimentar values (208, 'Energie', 'Sirop',  
'Copii');  
  
insert into supliment_alimentar values (209, 'General', 'Jeleuri',  
'Copii');  
  
insert into supliment_alimentar values (210, 'Articulatii', 'Pudra',  
'Sportivi');
```

-- 9. Cosmetic

```
insert into cosmetic values (301, 'Hidratare', '50ml');  
insert into cosmetic values (302, 'Hidratare', '250ml');  
insert into cosmetic values (303, 'Curatare', '400ml');  
insert into cosmetic values (304, 'Ingrijire', '200ml');  
insert into cosmetic values (305, 'Demachiere', '500ml');  
insert into cosmetic values (306, 'Anti-Age', '30ml');
```

```

insert into cosmetic values (307, 'Anti-Acnee', '150ml');
insert into cosmetic values (308, 'Reparare', '75ml');
insert into cosmetic values (309, 'Protectie UV', '50ml');
insert into cosmetic values (310, 'Antiperspirant', '150ml');
insert into cosmetic values (311, 'Hidratare', '50ml');

```

-- 10. Stoc

```

insert into stoc values (5001, 101, 10, 100, to_date('01-01-2025','dd-mm-yyyy'));
insert into stoc values (5002, 104, 20, 50, to_date('01-06-2024','dd-mm-yyyy'));
insert into stoc values (5003, 201, 30, 200, to_date('15-12-2024','dd-mm-yyyy'));
insert into stoc values (5004, 301, 100, 20, to_date('20-11-2025','dd-mm-yyyy'));
insert into stoc values (5005, 107, 20, 30, to_date('01-03-2024','dd-mm-yyyy'));
insert into stoc values (5006, 204, 40, 60, to_date('10-10-2024','dd-mm-yyyy'));
insert into stoc values (5007, 305, 100, 45, to_date('01-01-2026','dd-mm-yyyy'));
insert into stoc values (5008, 110, 10, 150, to_date('14-02-2025','dd-mm-yyyy'));
insert into stoc values (5009, 210, 50, 10, to_date('01-09-2024','dd-mm-yyyy'));
insert into stoc values (5010, 310, 90, 80, to_date('01-05-2025','dd-mm-yyyy'));

```

-- 11. Tranzactie

```
insert into tranzactie values (9001, str_to_date('01-12-2023', '%d-%m-%Y'), 18001011111111, 17001011111111, null);
```

```
insert into tranzactie values (9002, str_to_date('02-12-2023', '%d-%m-%Y'), 18001011111111, 28002022222222, null);
```

```
insert into tranzactie values (9003, str_to_date('03-12-2023', '%d-%m-%Y'), 29512124444444, 19003033333333, null);
```

```
insert into tranzactie values (9004, str_to_date('04-12-2023', '%d-%m-%Y'), 29512124444444, 29504044444444, null);
```

```
insert into tranzactie values (9005, str_to_date('05-12-2023', '%d-%m-%Y'), 19204049999999, 18505055555555, 1001);
```

```
insert into tranzactie values (9006, str_to_date('06-12-2023', '%d-%m-%Y'), 19204049999999, 28806066666666, 1002);
```

```
insert into tranzactie values (9007, str_to_date('07-12-2023', '%d-%m-%Y'), 17505057777777, 17807077777777, 1003);
```

```
insert into tranzactie values (9008, str_to_date('08-12-2023', '%d-%m-%Y'), 17505057777777, 29208088888888, 1004);
```

```
insert into tranzactie values (9009, str_to_date('09-12-2023', '%d-%m-%Y'), 18001011111111, 19609099999999, 1005);
```

```
insert into tranzactie values (9010, str_to_date('10-12-2023', '%d-%m-%Y'), 18001011111111, 29910100000000, 1006);
```

```
insert into tranzactie values (9011, str_to_date('11-12-2023', '%d-%m-%Y'), 17505057777777, 18808089991111, null);
```

```
insert into tranzactie values (9012, str_to_date('16-12-2023', '%d-%m-%Y'), 28506060000000, 29009098882222, null);
```

-- 12. Detalii_tranzactie

```
insert into detalii_tranzactie values (1, 9001, 101, 2);
```

```
insert into detalii_tranzactie values (2, 9002, 102, 1);
```

```
insert into detalii_tranzactie values (3, 9003, 201, 3);
```

```
insert into detalii_tranzactie values (4, 9004, 303, 1);
```

```
insert into detalii_tranzactie values (5, 9005, 104, 2);
```

```
insert into detalii_tranzactie values (6, 9006, 107, 1);
insert into detalii_tranzactie values (7, 9007, 105, 5);
insert into detalii_tranzactie values (8, 9008, 107, 2);
insert into detalii_tranzactie values (9, 9009, 101, 1);
insert into detalii_tranzactie values (10, 9009, 201, 1);
insert into detalii_tranzactie values (11, 9011, 111, 1);
insert into detalii_tranzactie values (12, 9012, 311, 3);
commit;
```

4.3. Ștergerea tabelelor

```
drop table if exists detalii_tranzactie;
drop table if exists tranzactie;
drop table if exists stoc;
drop table if exists reteta;
drop table if exists medicament;
drop table if exists supliment_alimentar;
drop table if exists cosmetic;
drop table if exists produs;
drop table if exists echipament;
drop table if exists furnizor;
drop table if exists client;
drop table if exists angajat;
```

5. Concluzie

Acest proiect mi-a oferit oportunitatea de a proiecta și dezvolta eficient și corect o bază de date.

Am explorat și utilizat sintaxa MySQL cu ajutorul aplicației MySQL Workbench. În plus, am folosit JavaScript prin NodeJS, Docker pentru hostarea serverului de API și HTML pentru a crea o interfață grafică care să permită manipularea datelor într-un mod intuitiv și eficient. Această experiență mi-a consolidat abilitățile tehnice și m-a ajutat să înțeleg mai bine cerințele unei platforme de management al unei farmacii.

6. Bibliografie

- <https://nodejs.org/en>
- <https://www.drawio.com/>
- <https://www.mysql.com/>
- <https://stackoverflow.com/>
- <https://www.docker.com/>