

UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

**PROIECT
BAZE DE DATE**

PROFESOR COORDONATOR:
VASILE SILVIU LAURENȚIU

STUDENT:
BRÎNCEANU RALUCA-ALEXANDRA

BUCUREȘTI
2024

UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

GESTIONAREA BAZEI DE DATE A UNEI GRĂDINIȚE

PROFESOR COORDONATOR:
VASILE SILVIU LAURENȚIU

STUDENT:
BRÎNCEANU RALUCA-ALEXANDRA

BUCUREȘTI
2024

CUPRINS

1. Prezentarea modelului	5
2. Regulile modelului	6
3. Diagrama Entitate-Relație	7
3.1. Reprezentarea diagramei	7
3.2. Descrierea entităților, atributelor, cheilor, relațiilor și a cardinalităților	8
3.2.1. Descrierea entităților, atributelor și a cheilor	8
3.2.1.1. Tabelul SERVICII	8
3.2.1.2. Tabelul PRESCOLARI	8
3.2.1.3 Tabelul ANGAJATI	9
3.2.1.4 Tabelul POSTURI	9
3.2.1.5 Tabelul SALI	9
3.2.1.6 Tabelul ECHIPAMENTE	10
3.2.1.7 Tabelul FURNIZORI	10
3.2.2. Descrierea relațiilor și a cardinalităților	11
3.2.2.1 Relația: SERVICII sunt solicitate de PRESCOLARI	11
3.2.2.2 Relația: SERVICII sunt prestate de ANGAJATI	11
3.2.2.3 Relația: PRESCOLARI sunt asociați cu ANGAJATI	11
3.2.2.4 Relația: POSTURI sunt ocupate de ANGAJATI	11
3.2.2.5 Relația: ANGAJATI lucrează în SALI	12
3.2.2.6 Relația: SALI conțin ECHIPAMENTE	12
3.2.2.7 Relația: ECHIPAMENTE sunt furnizate de FURNIZORI	12
4. Diagrama Conceptuală	13
4.1. Reprezentarea diagramei	13
4.2. Descrierea constrângerilor de integritate	14
4.2.1 Tabelul POSTURI	14
4.2.2 Tabelul SALI	14
4.2.3 Tabelul SERVICII	15
4.2.4 Tabelul PRESCOLARI	15
4.2.5 Tabelul ANGAJATI	16
4.2.6 Tabelul FURNIZORI	17
4.2.7 Tabelul ECHIPAMENTE	17
4.2.8 Tabelul INSCRIERI	18

4.2.9 Tabelul REPARTIARE_SALI	19
4.2.10 Tabelul INVENTARE	19
4.2.11 Tabelul LIVRARI	20
4.3. Schemele relaționale	20
4.3.1 Schemele relaționale.....	20
4.3.2 Descrierea constrângerilor ON DELETE CASCADE / SET NULL	21
5. Implementarea într-un SGBD	22
5.1 Crearea tabelelor și a constrângerilor	22
5.1.1 Crearea tabelului POSTURI	22
5.1.2 Crearea tabelului SALI	23
5.1.3 Crearea tabelului SERVICII	23
5.1.4 Crearea tabelului PRESCOLARI	23
5.1.5 Crearea tabelului ANGAJAT	24
5.1.6 Crearea tabelului FURNIZORI.....	24
5.1.7 Crearea tabelului ECHIPAMENTE	25
5.1.8 Crearea tabelului INSCRIERI	25
5.1.9 Crearea tabelului REPARTIZARE_SALI	26
5.1.10 Crearea tabelului INVENTARE.....	26
.....	26
5.1.11 Crearea tabelului LIVRARI.....	27
5.2 Introducerea datelor în tabele	28
5.2.1 Introducerea datelor în tabelul POSTURI	28
5.2.2 Introducerea datelor în tabelul SALI	29
5.2.3 Introducerea datelor în tabelul SERVICII	29
5.2.4 Introducerea datelor în tabelul PRESCOLARI.....	30
5.2.5 Introducerea datelor în tabelul ANGAJATI	30
5.2.6 Introducerea datelor în tabelul FURNIZORI.....	30
5.2.7 Introducerea datelor în tabelul ECHIPAMENTE	31
5.2.8 Introducerea datelor în tabelul INSCRIERI	31
5.2.9 Introducerea datelor în tabelul REPARTIZARE_SALI	32
5.2.10 Introducerea datelor în tabelul INVENTARE.....	32
5.2.11 Introducerea datelor în tabelul LIVRARI.....	32

1. Prezentarea modelului

Tema aleasă pentru proiectul la disciplina Baze de date este “Gestionarea bazei de date a unei grădinițe”. În realizarea serviciilor, angajații folosesc echipamente și materiale, provenite de la mai mulți furnizori.

Administrarea unei grădinițe presupune, în primul rând, evidența copiilor înscriși, a serviciilor pe care le solicită și a interacțiunii acestora cu angajații. Se monitorizează, de asemenea, furnizarea echipamentelor necesare și distribuția lor în interiorul clinicii.

În realizarea bazei de date, am considerat câteva aspecte care definesc grădinița: aceasta se extinde pe două etaje și parter, iar programul cu publicul este zilnic, între orele 9-17. Aceste informații vor limita introducerea înregistrărilor în baza de date. Implementarea acesteia a fost realizată folosind MySQL 8.0.34.

2. Regulile modelului

- Pot exista servicii nesolicitate, dar și servicii solicitate de mai mulți preșcolari. $M(0)$

Pot exista preșcolari care nu solicită niciun serviciu la un anumit moment dat, dar și preșcolari care solicită mai multe servicii. $M(0)$

- Un serviciu poate fi prestat de unul sau mai mulți angajați. $M(1)$

Pot exista angajați care nu prestează niciun serviciu la un anumit moment dat, dar și angajați care prestează mai multe servicii. $M(0)$

- Un preșcolar poate fi asociat cu unul sau mai mulți angajați. $M(1)$

Pot exista angajați care nu răspund de niciun preșcolar în mod direct (ex. director), dar și angajați responsabili de mai mulți preșcolari. $M(0)$

- Un angajat trebuie să ocupe un singur post. $1(1)$

Pot exista posturi care să nu fie ocupate de niciun angajat, dar și posturi ocupate de mai mulți angajați. $M(0)$

- Pot exista angajați care nu lucrează în nicio sală, dar și angajați care lucrează în mai multe săli. $M(0)$

O sală poate să nu fie repartizată niciunui angajat la un anumit moment dat, dar poate fi repartizată și mai multor angajați. $M(0)$

- O sală poate conține unul sau mai multe echipamente. $M(1)$

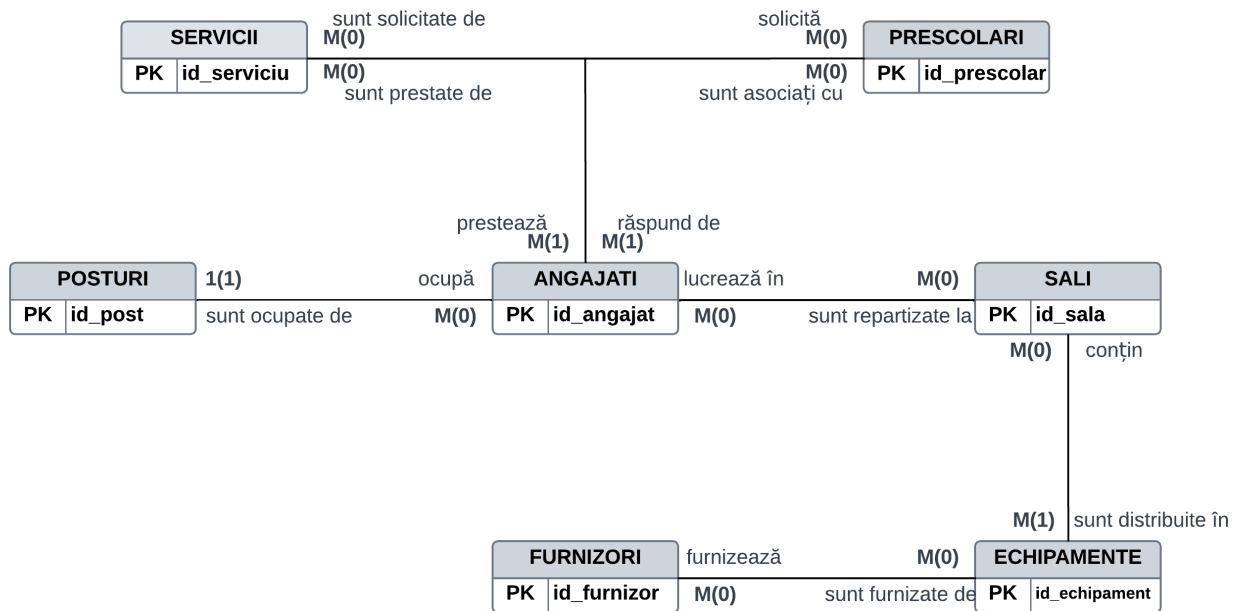
Pot exista echipamente care nu sunt distribuite în nicio sală, dar și echipamente distribuite în mai multe săli. $M(0)$

- Există echipamente care nu sunt furnizate de niciun furnizor, dar și echipamente furnizate de mai mulți furnizori. $M(0)$

Pot exista furnizori care nu furnizează niciun echipament la un anumit moment dat, dar și furnizori care furnizează mai multe echipamente. $M(0)$

3. Diagrama Entitate-Relație

3.1. Reprezentarea diagramei



3.2. Descrierea entităților, atributelor, cheilor, relațiilor și a cardinalităților

3.2.1. Descrierea entităților, atributelor și a cheilor

3.2.1.1. Tabelul SERVICII

- conține informații despre serviciile ce pot fi oferite preșcolarilor;

Cheie	Atribut	Tip de date	Descriere
PK	id_serviciu	integer	Identificatorul unic al serviciului
-	denumire_serv	varchar(50)	Denumirea serviciului
-	specializare	varchar(50)	Specializarea din care aparține serviciul. Aceasta poate fi una din următoarele: secretariat, educație, activități, îngrijire, sănătate
-	pret	double	Prețul serviciului

3.2.1.2. Tabelul PRESCOLARI

- conține informații despre copiii înscriși în cadrul grădiniței;

Cheie	Atribut	Tip de date	Descriere
PK	id_prescolar	integer	Identificatorul unic al preșcolarului
-	nume	varchar(50)	Numele preșcolarului
-	prenume	varchar(50)	Prenumele preșcolarului
-	data_nastere	date	Data nașterii preșcolarului
-	telefon_parinte	varchar(10)	Numărul de telefon al părintelui
-	email_parinte	varchar(50)	Adresa de email a părintelui
-	nevoi_speciale	boolean	1 (true): dacă preșcolarul necesită suport suplimentar din cauza unor cerințe particulare. Altfel, 0 (false).

3.2.1.3 Tabelul ANGAJATI

- conține informații despre angajații grădiniței;

Cheie	Atribut	Tip de date	Descriere
PK	id_angajat	integer	Identificatorul unic al angajatului
FK	id_post	integer	Identificatorul postului pe care îl deține angajatul
-	nume	varchar(50)	Numele preșcolarului
-	prenume	varchar(50)	Prenumele preșcolarului
-	telefon	varchar(10)	Numărul de telefon al angajatului
-	email	varchar(50)	Adresa de email a angajatului
-	salariu	double	Salariul angajatului

3.2.1.4 Tabelul POSTURI

- conține informații despre posturile pe care le pot deține angajații

Cheie	Atribut	Tip de date	Descriere
PK	id_post	integer	Identificatorul unic al postului
-	denumire_post	varchar(50)	Denumirea postului (educator etc.)
-	salariu_min	integer	Salariul minim care poate fi oferit pentru post
-	studii_necesare	varchar(30)	Nivelul minim de studii necesar pentru post

3.2.1.5 Tabelul SALI

- conține informații despre sălile în care se desfășoară activitățile

Cheie	Atribut	Tip de date	Descriere
PK	id_sala	integer	Identificatorul unic al sălii
-	tip	varchar(50)	Tipul sălii. Poate fi unul din următoarele: sală clasă, sală mese, sală odihnă, sală sport, sală activități, sală festivitate, sală medic, recepție
-	suprafață	double	Suprafața încăperii, măsurată în m2
-	etaj	integer	Etajul pe care se află sala. Se încadrează între 0 și 2 (grădinița are 2 etaje)

3.2.1.6 Tabelul ECHIPAMENTE

- conține informații despre echipamentele și materialele generale de care este nevoie în grădiniță

Cheie	Atribut	Tip de date	Descriere
PK	id_echipament	integer	Identificatorul unic al echipamentului
-	denumire	varchar(50)	Denumirea echipamentului
-	categorie	varchar(50)	Categoria din care face parte echipamentul
-	pret	double	Prețul echipamentului

3.2.1.7 Tabelul FURNIZORI

- conține informații despre furnizorii cu care poate colabora grădinița;

Cheie	Atribut	Tip de date	Descriere
PK	id_furnizor	integer	Identificatorul unic al furnizorului
-	nume	varchar(50)	Numele furnizorului
-	telefon	varchar(10)	Numărul de telefon al furnizorului
-	email	varchar(50)	Adresa de email a furnizorului

3.2.2. Descrierea relațiilor și a cardinalităților

3.2.2.1 Relația: *SERVICII* sunt solicitate de *PRESCOLARI*

→ Cardinalitate minimală

- Câți preșcolari trebuie să solicite același serviciu? 0
- Câte servicii trebuie să fie solicitate de un preșcolar? 0

→ Cardinalitate maximală

- Câți preșcolari pot solicita același serviciu? M
- Câte servicii pot fi solicitate de un preșcolar? M

3.2.2.2 Relația: *SERVICII* sunt prestate de *ANGAJATI*

→ Cardinalitate minimală

- Câți angajați trebuie să presteze același serviciu? 1
- Câte servicii trebuie să fie prestate de un angajat? 0

→ Cardinalitate maximală

- Câți angajați pot presta același serviciu? M
- Câte servicii pot fi prestate de un angajat? M

3.2.2.3 Relația: *PRESCOLARI* sunt asociați cu *ANGAJATI*

→ Cardinalitate minimală

- Câți angajați trebuie să răspundă de același preșcolar? 1
- Câți preșcolari trebuie să fie asociați cu un angajat? 0

→ Cardinalitate maximală

- Câți angajați pot să răspundă de același preșcolar? M
- Câți preșcolari trebuie să fie asociați cu un angajat? M

3.2.2.4 Relația: *POSTURI* sunt ocupate de *ANGAJATI*

→ Cardinalitate minimală

- Câți angajați trebuie să ocupe același post? 0
- Câte posturi trebuie să fie ocupate de un angajat? 1

→ Cardinalitate maximală

- Câți angajați pot să ocupe același post? M
- Câte posturi pot fi ocupate de un angajat? 1

3.2.2.5 Relația: *ANGAJATI* lucrează în *SALI*

→ Cardinalitate minimală

- Câți angajați trebuie să lucreze în aceeași sală? 0
- Câte săli trebuie să fie repartizate la un angajat? 0

→ Cardinalitate maximală

- Câți angajați pot să lucreze în aceeași sală? M

3.2.2.6 Relația: *SALI* conțin *ECHIPAMENTE*

→ Cardinalitate minimală

- Câte echipamente trebuie să fie distribuite în aceeași sală? 1
- Câte săli trebuie să conțină un echipament? 0
 - un echipament poate fi numai cumpărat

→ Cardinalitate maximală

- Câte echipamente pot fi distribuite în aceeași sală? M
- Câte săli pot conține un echipament? M

3.2.2.7 Relația: *ECHIPAMENTE* sunt furnizate de *FURNIZORI*

→ Cardinalitate minimală

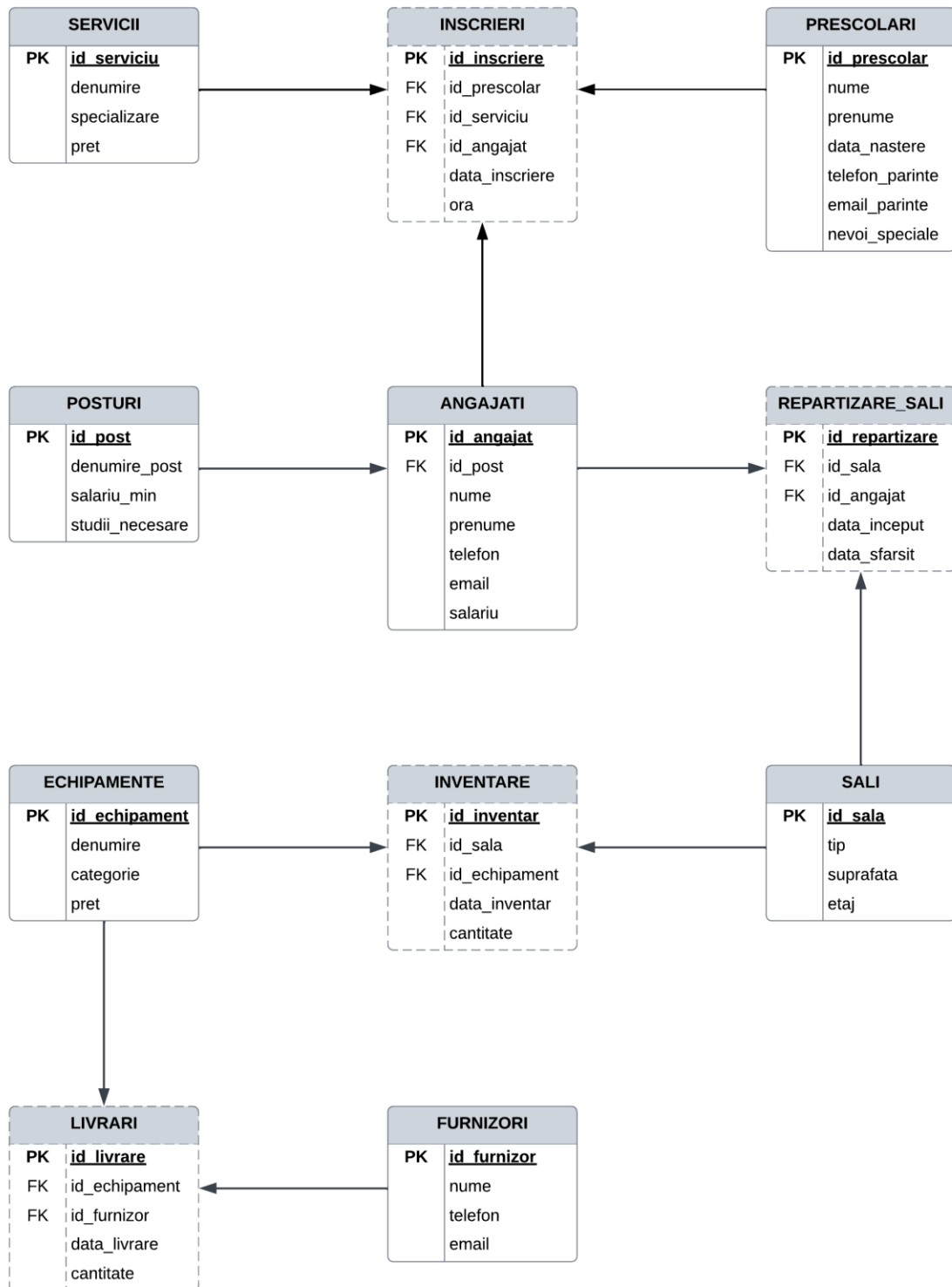
- Câte echipamente trebuie să furnizeze același furnizor? 0
- Câți furnizori trebuie să furnizeze un echipament? 0

→ Cardinalitate maximală

- Câte echipamente pot fi furnizate de același furnizor? M
- Câți furnizori pot să furnizeze un echipament?

4. Diagrama Conceptuală

4.1. Reprezentarea diagramei



4.2. Descrierea constrângerilor de integritate

4.2.1 Tabelul POSTURI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_POSTURI	id_post	Identificatorul unic al postului
NOT NULL	-	denumire_post	Fiecare post trebuie să aibă o denumire
	-	salariu_min	Fiecare post trebuie să aibă un salariu minim
CHECK	STUDII_CK	studii_necesare IN ('medii', 'superioare')	Se va verifica dacă studiile minime necesare sunt medii sau superioare
	SAL_MIN_CK	salariu_min	Se va verifica dacă salariul minim este mai mare decât minimul pe economie (salariu_min > 2079)
UNIQUE	DEN_POST_UQ	denumire_post	Nu pot exista mai multe posturi cu aceeași denumire

4.2.2 Tabelul SALI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_SALA	id_sala	Identificatorul unic al sălii
NOT NULL	-	tip	Fiecare sală este de un anumit tip
	-	suprafata	Fiecare sală are o anumită suprafață
	-	etaj	Fiecare sală este situată pe un anumit etaj
CHECK	TIP_CK	tip IN ('sala clasa', 'sala mese', 'sala odihna', 'sala sport', 'sala activitati', 'sala festivitate', 'sala medic', 'receptie')	Tipul sălii trebuie să fie unul dintre următoarele: sală clasă / mese / odihnă / sport / activități / medic și recepție
	SUPRAFATA_C	suprafata > 0	Suprafața este un număr pozitiv
	ETAJ_CK	etaj >= 0 AND etaj <= 1	Etajul se încadrează între 0 (parter) și 1

4.2.3 Tabelul SERVICII

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_SERVICIU	id_serviciu	Identificatorul unic al serviciului
NOT NULL	-	denumire_serv	Fiecare serviciu trebuie să aibă o denumire
	-	specializare	Fiecare serviciu trebuie să aibă o specializare
CHECK	SPECIALIZARE_CHK	specializare IN ('secretariat', 'educatie', 'activitati', 'ingrijire', 'sanatate')	Specializarera trebuie să fie una dintre cele enumerate
	PRET_SERV_CHK	pret >= 0	Prețul este un număr pozitiv
UNIQUE	DEN_SERV_UQ	denumire_serv	Fiecare serviciu trebuie să aibă o denumire unică

4.2.4 Tabelul PRESCOLARI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_PRESCOLAR	id_prescolar	Identificatorul unic al preșcolarului
NOT NULL	-	nume	Fiecare preșcolar trebuie să aibă un nume de familie
	-	prenume	Fiecare preșcolar trebuie să aibă un prenume
	-	data_nastere	Fiecare preșcolar trebuie să aibă o dată de naștere
	-	telefon_parinte	Fiecare preșcolar trebuie să aibă un număr de telefon al părintelui
	-	email_parinte	Fiecare preșcolar trebuie să aibă o adresă de email a părintelui
	-	nevoi_speciale	Fiecărui preșcolar trebuie să i se specifice dacă are nevoi speciale
CHECK	EMAIL_CHK	email_parinte LIKE '%@%.%'	Adresa de email a părintelui trebuie să respecte structura unui email

4.2.5 Tabelul ANGAJATI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_ANGAJAT	id_angajat	Identificatorul unui angajat
FOREIGN KEY	FK_POST_ANG	(id_post) REFERENCES posturi (id_post)	Coloana id_post referă coloana id_post din tabelul POSTURI
NOT NULL	-	id_post	Fiecare angajat trebuie să ocupe un post
	-	nume	Fiecare angajat trebuie să aibă un nume de familie
	-	prenume	Fiecare angajat trebuie să aibă un prenume
	-	telefon	Fiecare angajat trebuie să aibă un număr de telefon
	-	email	Fiecare angajat trebuie să aibă o adresă de email
	-	salariu	Fiecare angajat trebuie să aibă un salariu
CHECK	EMAIL_ANG_CHK	email LIKE '%@%.%'	Emailul angajatului trebuie să respecte structura unui email
	SAL_ANG_CHK	salariu > 0	Salariul este un număr pozitiv
UNIQUE	TEL_ANG_UQ	telefon	Fiecare angajat are un număr de telefon unic
	EMAIL_ANG_UQ	email	Fiecare angajat are o adresă de email unică

4.2.6 Tabelul FURNIZORI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_FURNIZOR	id_furnizor	Identificatorul unic al unui furnizor
NOT NULL	-	nume	Fiecare furnizor trebuie să aibă un nume
	-	telefon	Fiecare furnizor trebuie să aibă un număr de telefon
	-	email	Fiecare furnizor trebuie să aibă o adresă de email
CHECK	EMAIL_FZ_CK	email LIKE '%@%.%'	Emailul furnizorului trebuie să respecte structura unui email
UNIQUE	NUME_FZ_UQ	nume	Fiecare furnizor are un nume unic
	TEL_FZ_UQ	telefon	Fiecare furnizor are un număr de telefon unic
	EMAIL_FZ_UQ	email	Fiecare furnizor are o adresă de email unică

4.2.7 Tabelul ECHIPAMENTE

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_ECHIPAMENT	id_echipament	Identificatorul unic al unui echipament
NOT NULL	-	denumire	Fiecare echipament trebuie să aibă o denumire
	-	categorie	Fiecare echipament trebuie să facă parte dintr-o categorie
CHECK	PRET_CK	pret > 0	Prețul este un număr pozitiv
UNIQUE	DEN_ECH_UQ	denumire	Fiecare echipament are o denumire unică

4.2.8 Tabelul INSCRIERI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_INSCRIERE	id_inscriere	Identificatorul unic al unei înscrieri
FOREIGN KEY	FK_PRESCOLAR	(id_prescolar) REFERENCES prescolari(id_prescolar)	Coloana id_prescolar referă coloana id_prescolar din tabelul PRESCOLARI
	FK_SERVICIU	(id_serviciu) REFERENCES servicii(id_serviciu)	Coloana id_serviciu referă coloana id_serviciu din tabelul SERVICII
	FK_ANGAJAT	(id_angajat) REFERENCES angajati(id_angajat)	Coloana id_angajat referă coloana id_angajat din tabelul ANGAJATI
NOT NULL	-	data_inscriere	Fiecare înscriere trebuie să aibă o dată
	-	ora	Fiecare înscriere trebuie să aibă o oră
CHECK	ORA_CK	ora >= 9 AND ora <= 17	Grădinița are program de la 9- 17
UNIQUE	INSCRIERE_UQ	(id_prescolar, id_serviciu, id_angajat)	Pentru a nu se suprapune înscrierile, tuplul trebuie să fie unic

4.2.9 Tabelul REPARTIARE_SALI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_REP	id_repartizare	Identificatorul unic al unei repartizări
FOREIGN KEY	FK_ANG_REP	(id_angajat) REFERENCES angajati(id_sangajat)	Coloana id_angajat referă coloana id_angajat din tabelul ANGAJATI
	FK_SALA_REP	(id_sala) REFERENCES sali(id_sala)	Coloana id_sala referă coloana id_sala din tabelul SALI
NOT NULL	-	id_sala	Fiecare repartizare trebuie să fie făcută pentru o sală
	-	data_inceput	Fiecare repartizare trebuie să conțină o dată de început
	-	data_sfarsit	Fiecare repartizare trebuie să conțină o dată de sfârșit
CHECK	DATA_REP_CK	data_sfarsit >= data_inceput	Data la care se termină repartizarea trebuie să fie mai nouă decât data începerii
UNIQUE	REP_UQ	(id_sala, data_inceput, data_sfarsit)	Repartizările se fac pe an

4.2.10 Tabelul INVENTARE

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_INVENTAR	id_inventar	Identificatorul unic al unui inventar
FOREIGN KEY	FK_SALA_INV	(id_sala) REFERENCES sali(id_sala)	Coloana id_sala referă coloana id_sala din tabelul SALI
	FK_ECH_INV	(id_echipament) REFERENCES echipamente(id_echipament)	Coloana id_echipament referă coloana id_echipament din tabelul ECHIPAMENTE
NOT NULL	-	data_inventar	fiere inventar trebuie să conțină data
	-	cantitate	Fiecare inventar trebuie să conțină cantitatea
CHECK	CANTITATE_CK	cantitate > 0	Fiecare inventar trebuie să conțină cantitatea
UNIQUE	INVENTAR_UQ	(id_sala, id_echipament, data_inventar)	Inventarele se fac în zile diferite

4.2.11 Tabelul LIVRARI

Tipul constrângerii	Denumirea constrângerii	Coloanele (sau verificările) la care se referă constrângerile	Explicații
PRIMARY KEY	PK_LIVRARE	id_livrare	Identificatorul unic al unei livrări
FOREIGN KEY	FK_ECH_LIV	(id_echipament) REFERENCER echipamente(id_echipament)	Coloana id_echipament referă coloana id_echipament din tabelul ECHIPAMENTE
	FK_FZ_LIV	(id_furnizor) REFERENCES furnizori(id_furnizor)	Coloana id_furnizor referă coloana id_furnizor din tabelul FURNIZORI
NOT NULL	-	data_livrare	Fiecare livrare trebuie să se realizeze la o anumită dată
	-	cantitate	Fiecare livrare trebuie să conțină cantitatea
CHECK	CANT_LIV_CK	cantitate > 0	Cantitate este un număr pozitiv

4.3. Schemele relaționale

4.3.1 Schemele relaționale

Schemele relaționale ale diagramei conceptuale atașate sunt următoarele:

SERVICII (id_serviciu#, denumire_serv, specializare, pret);

PRESCOLARI (id_prescolar#, nume, prenume, data_nastere, telefon_parinte, email_parinte, nevoi_speciale);

ANGAJATI (id_angajat#, id_post, nume, prenume, telefon, email, salariu)

POSTURI (id_post#, denumire_post, salariu_min, studii_necesare);

SALI (id_sala#, tip, suprafata, etaj);

ECHIPAMENTE (id_echipament#, denumire, categorie, pret);

FURNIZORI (id_furnizor#, nume, telefon, email);

INSCRIERI (id_inscriere#, id_prescolar, id_serviciu, id_angajat, data_inscriere, ora);

REPARTIZARE_SALI (id_repartizare#, id_sala, id_angajat, data_inceput, data_sfarsit);

INVENTARE (id_inventar#, id_sala, id_echipament, data_inventar, cantitate);

LIVRARI (id_livrare#, id_echipament, id_furnizor, data_livrare, cantitate);

4.3.2 Descrierea constrângerilor ON DELETE CASCADE / SET NULL

Constrângerile de ștergere se aplică cheilor externe, conform constrângerilor NOT NULL. Dacă o coloană trebuie să fie diferită de *null*, atunci la ștergerea ei se va folosi o constrângere de tipul *on delete cascade*, prin care se șterge întreaga înregistrare. Altfel, coloanei i se aplică *on delete set null*.

- FK_POST_ANG, on delete cascade
 - La ștergerea unui post, toți angajații care ocupă postul respectiv vor fi șterși.
- FK_PRESCOLAR, on delete cascade
 - La ștergerea unui preșcolar, toate înscrierile care au același id_prescolar vor fi șterse.
- FK_SERVCIU, on delete cascade
 - La ștergerea unui serviciu, toate înscrierile care au acel serviciu vor fi șterse.
- FK_ANGAJAT, on delete set null
 - La ștergerea unui angajat, coloana id_angajat va fi setată la null pentru înscrierile făcute.
- FK_ANG_REP, on delete set null
 - La ștergerea unui angajat, coloana id_angajat va fi setată la null pentru repartizările făcute angajatului șters.
- FK_SALA_REP, on delete cascade
 - La ștergerea unei săli, toate repartizările făcute pentru sala respectivă vor fi șterse.
- FK_SALA_INV, on delete set null
 - La ștergerea unei săli, coloana id_sala va fi setată la null pentru inventarele care conțineau sala ștersă.
- FK_ECH_INV, on delete set null
 - La ștergerea unui echipament, coloana id_echipament va fi setată la null pentru inventarele care conțineau echipamentul șters.
- FK_ECH_LIV, on delete set null
 - La ștergerea unui echipament, coloana id_echipament va fi setată la null pentru livrările care conțineau echipamentul șters.
- FK_FZ_LIV, on delete set null
 - FK_FZ_LIV, on delete set null
 - La ștergerea unui furnizor, coloana id_furnizor va fi setată la null pentru livrările care realizate de furnizorul șters.

5. Implementarea într-un SGBD

5.1 Crearea tabelelor și a constrângerilor

Ordinea în care se creează tabelele contează, deoarece tabelele care conțin chei externe trebuie să refere coloane din tabele deja existente. Toate constrângerile menționate anterior trebuie să se regăsească în implementarea tabelelor. Dacă tabelele există deja, acestea vor fi șterse folosind comanda *drop table if exists*.

- `drop table if exists servicii;`
- `drop table if exists prescolari;`
- `drop table if exists angajati;`
- `drop table if exists posturi;`
- `drop table if exists sali;`
- `drop table if exists echipamente;`
- `drop table if exists furnizori;`
- `drop table if exists inscrieri;`
- `drop table if exists repartizare_sali;`
- `drop table if exists inventare;`
- `drop table if exists livrari;`

5.1.1 Crearea tabelului POSTURI

```
• create table posturi
  (
    id_post integer,
    denumire_post varchar(50) not null,
    salariu_min integer not null,
    studii_necesare varchar(30),

    constraint PK_POSTURI primary key (id_post),
    constraint STUDII_CK check (studii_necesare in ('medii', 'superioare')),
    constraint SAL_MIN_CK check (salariu_min > 2079),
    constraint DEN_POST_UQ unique (denumire_post)
  );
```

5.1.2 Crearea tabelului SALI

- `create table sali`

```
(
    id_sala integer,
    tip varchar(50) not null,
    suprafata double not null,
    etaj integer not null,

    constraint PK_SALA primary key (id_sala),
    constraint TIP_CK check (tip in
        ('sala clasa', 'sala mese', 'sala odihna', 'sala sport',
        'sala activitati', 'sala festivitate', 'sala medic', 'receptie')),
    constraint SUPRAFATA_CK check (suprafata > 0),
    constraint ETAJ_CK check (etaj >= 0 and etaj <=1)
);
```

5.1.3 Crearea tabelului SERVICII

- `create table servicii`

```
(
    id_serviciu integer,
    denumire_serv varchar(50) not null,
    specializare varchar(50) not null,
    pret double,

    constraint PK_SERVICIU primary key (id_serviciu),
    constraint SPECIALIZARE_CK check (specializare in
        ('secretariat', 'educatie', 'activitati', 'ingrijire', 'sanatate')),
    constraint PRET_SERV_CK check (pret >= 0),
    constraint DEN_SERV_UQ unique (denumire_serv)
);
```

5.1.4 Crearea tabelului PRESCOLARI

- `create table prescolari`

```
(
    id_prescolar integer,
    nume varchar(50) not null,
    prenume varchar(50) not null,
    data_nastere date not null,
    telefon_parinte varchar(10) not null,
    email_parinte varchar(50) not null,
    nevoi_speciale boolean not null,

    constraint PK_PRESCOLAR primary key (id_prescolar),
    constraint EMAIL_CK check (email_parinte like '%@%.%')
);
```

5.1.5 Crearea tabelului ANGAJAT

- **create table** angajati

```
(  
    id_angajat integer,  
    id_post integer not null,  
    nume varchar(50) not null,  
    prenume varchar(50) not null,  
    telefon varchar(10) not null,  
    email varchar(50) not null,  
    salariu double not null,  
  
    constraint PK_ANGAJAT primary key (id_angajat),  
    constraint FK_POST_ANG foreign key (id_post) references  
        posturi (id_post) on delete cascade,  
    constraint EMAIL_ANG_CK check (email like '%@%.%'),  
    constraint SAL__ANG_CK check (salariu > 0),  
    constraint TEL_ANG_UQ unique (telefon),  
    constraint EMAIL_ANG_UQ unique (email)  
);
```

5.1.6 Crearea tabelului FURNIZORI

- **create table** furnizori

```
(  
    id_furnizor integer,  
    nume varchar(50) not null,  
    telefon varchar(10) not null,  
    email varchar(50) not null,  
  
    constraint PK_FURNIZOR primary key (id_furnizor),  
    constraint EMAIL_FR_CK check (email like '%@%.%'),  
    constraint NUME_FZ_UQ unique (nume),  
    constraint TEL_FZ_UQ unique (telefon),  
    constraint EMAIL_FZ_UQ unique (email)  
);
```


5.1.7 Crearea tabelului ECHIPAMENTE

- **create table** echipamente

```
(  
    id_echipament integer,  
    denumire varchar(50) not null,  
    categorie varchar(50) not null,  
    pret double,  
  
    constraint PK_ECHIPAMENT primary key (id_echipament),  
    constraint PRET_CK check (pret > 0),  
    constraint DEN_ECH_UQ unique (denumire)  
);
```

5.1.8 Crearea tabelului INSCRIERI

- **create table** inscrieri

```
(  
    id_inscriere integer,  
    id_prescolar integer,  
    id_serviciu integer,  
    id_angajat integer,  
    data_inscriere date not null,  
    ora integer not null,  
  
    constraint PK_INSCRIERE primary key (id_inscriere),  
    constraint FK_PRESCOLAR foreign key (id_prescolar) references  
        prescolari(id_prescolar) on delete cascade,  
    constraint FK_SERVICIU foreign key (id_serviciu) references  
        servicii(id_serviciu) on delete cascade,  
    constraint FK_ANGAJAT foreign key (id_angajat) references  
        angajati(id_angajat) on delete set null,  
    constraint ORA_CK check (ora >= 9 and ora <= 17),  
    constraint INSCRIERE_UQ unique (id_prescolar, id_serviciu, id_angajat)  
);
```

5.1.9 Crearea tabelului REPARTIZARE_SALI

- **create table** repartizare_sali

```
(  
    id_repartizare integer,  
    id_sala integer not null,  
    id_angajat integer,  
    data_inceput date not null,  
    data_sfarsit date not null,  
  
    constraint PK_REP primary key (id_repartizare),  
    constraint FK_ANG_REP foreign key (id_angajat) references  
        angajati(id_angajat) on delete set null,  
    constraint FK_SALA_REP foreign key (id_sala) references  
        sali(id_sala) on delete cascade,  
    constraint DATA_REP_CK check (data_sfarsit >= data_inceput),  
    constraint REP_UQ unique (id_sala, data_inceput, data_sfarsit)  
);
```

5.1.10 Crearea tabelului INVENTARE

- **create table** inventare

```
(  
    id_inventar integer,  
    id_sala integer,  
    id echipament integer,  
    data_inventar date not null,  
    cantitate integer not null,  
  
    constraint PK_INVENTAR primary key (id_inventar),  
    constraint FK_SALA_INV foreign key (id_sala) references  
        sali(id_sala) on delete set null,  
    constraint FK_ECH_INV foreign key (id_echipament) references  
        echipamente(id_echipament) on delete set null,  
    constraint CANTITATE_CK check (cantitate > 0),  
    constraint INVENTAR_UQ unique (id_sala, id_echipament, data_inventar)  
);
```

5.1.11 Crearea tabelului LIVRARI

- `create table livrari`

```
(  
    id_livrare integer,  
    id_echipament integer,  
    id_furnizor integer,  
    data_livrare date not null,  
    cantitate integer not null,  
  
    constraint PK_LIVRARE primary key (id_livrare),  
    constraint FK_ECH_LIV foreign key (id_echipament) references  
        echipamente(id_echipament) on delete set null,  
    constraint FK_FZ_LIV foreign key (id_furnizor) references  
        furnizori(id_furnizor) on delete set null,  
    constraint CANT_LIV_CK check (cantitate > 0)  
);
```

5.2 Introducerea datelor în tabele

Ordinea introducerii datelor are importanță. De exemplu, înregistrările din tabelul ANGAJATI sunt dependente de existența unor înregistrări în tabelul POSTURI, caz în care contează ordinea tabelelor la inserare. În plus, înregistrările introduse trebuie să respecte constrângerile de integritate impuse.

5.2.1 Introducerea datelor în tabelul POSTURI

- `insert into posturi
values (10, 'educator', 4500, 'superioare');`
- `insert into posturi
values (11, 'asistent educator', 2200, 'superioare');`
- `insert into posturi
values (12, 'medic', 6000, 'superioare');`
- `insert into posturi
values (13, 'asistent medical', 4500, 'superioare');`
- `insert into posturi
values (14, 'director', 5000, 'superioare');`
- `insert into posturi
values (15, 'bucatar', 3000, 'medii');`
- `insert into posturi
values (16, 'secretara', 3500, 'superioare');`
- `insert into posturi
values (17, 'menajera', 3000, 'medii');`

5.2.2 Introducerea datelor în tabelul SALI

- `insert into sali`
`values (1, 'sala clasa', 40, 1);`
- `insert into sali`
`values (2, 'sala clasa', 45, 0);`
- `insert into sali`
`values (3, 'sala mese', 60, 0);`
- `insert into sali`
`values (4, 'sala odihna', 95, 1);`
- `insert into sali`
`values (5, 'sala sport', 80, 0);`
- `insert into sali`
`values (6, 'sala activitati', 60, 1);`
- `insert into sali`
`values (7, 'sala festivitate', 85, 0);`
- `insert into sali`
`values (8, 'sala medic', 30, 0);`
- `insert into sali`
`values (9, 'receptie', 45, 0);`

5.2.3 Introducerea datelor în tabelul SERVICII

- `insert into servicii`
`values (50, 'taxa inscriere', 'secretariat', 3000);`
- `insert into servicii`
`values (51, 'taxa scolarizare', 'secretariat', 45000);`
- `insert into servicii`
`values (52, 'odihna', 'ingrijire', 5000);`
- `insert into servicii`
`values (53, 'baschet', 'activitati', 0);`
- `insert into servicii`
`values (54, 'balet', 'activitati', 0);`
- `insert into servicii`
`values (55, 'limba straina', 'educatie', 0);`

5.2.4 Introducerea datelor în tabelul PRESCOLARI

- `insert into prescolari`
`values (100, 'Bratosin', 'Sonia', '2020-06-12', '0264526502', 'voicu.bratosin@yahoo.com', true);`
- `insert into prescolari`
`values (101, 'Palici', 'Andrei', '2021-10-25', '0765533726', 'dinu_palici79@gmail.com', false);`
- `insert into prescolari`
`values (102, 'Albu', 'Eduard', '2020-07-16', '0351543546', 'ant66craciun@gmail.com', false);`
- `insert into prescolari`
`values (103, 'Stoica', 'Ania', '2021-11-02', '0711737630', 'iacob64@yahoo.com', false);`
- `insert into prescolari`
`values (104, 'Pop', 'Melania', '2021-03-19', '0265098905', 'imatei@stefan.org', true);`
- `insert into prescolari`
`values (105, 'Gherban', 'Melania', '2021-04-15', '0754160453', 'anaban@gmail.com', true);`
- `insert into prescolari`
`values (106, 'Pop', 'Alec', '2021-07-16', '0379437049', 'mihaipop@yahoo.com', false);`
- `insert into prescolari`
`values (107, 'Stoica', 'Valentin', '2020-10-06', '0765551448', 'ilestoica@gmail.com', false);`
- `insert into prescolari`
`values (108, 'Paul', 'Iancu', '2021-11-24', '0756647978', 'cosmin17@gmail.com', true);`
- `insert into prescolari`
`values (109, 'Pavel', 'Smaranda', '2020-06-14', '0358197115', 'iurie.patrascu@gmail.com', false);`

5.2.5 Introducerea datelor în tabelul ANGAJATI

- `insert into angajati`
`values (1, 10, 'Dascalu', 'Sonia', '0706047433', 'sonia.dascalu@sch.ro', 66000);`
- `insert into angajati`
`values (2, 11, 'Radoi', 'Catinca', '0279522704', 'catinca.radoi@sch.ro', 38000);`
- `insert into angajati`
`values (3, 12, 'Stan', 'Andrei', '0211946437', 'medic@sch.ro', 100000);`
- `insert into angajati`
`values (4, 13, 'Preda', 'Lucia', '0373379299', 'asistent@sch.ro', 60000);`
- `insert into angajati`
`values (5, 14, 'Nicolescu', 'Dariana', '0738256199', 'director@sch.ro', 72000);`
- `insert into angajati`
`values (6, 15, 'Stan', 'Mina', '0336068895', 'bucatarie@sch.ro', 39400);`
- `insert into angajati`
`values (7, 16, 'Necula', 'Ariana', '0330477136', 'secretariat@sch.ro', 54000);`
- `insert into angajati`
`values (8, 17, 'Zaharia', 'Sabina', '0256486511', 'serviciu@sch.ro', 43800);`

5.2.6 Introducerea datelor în tabelul FURNIZORI

- `insert into furnizori`
`values (10, 'Furnizor ABC', '0755551141', 'furnizor_abc@gmail.com');`
- `insert into furnizori`
`values (11, 'Furnizor XYZ', '0711112248', 'furnizor_xyz@yahoo.com');`
- `insert into furnizori`
`values (12, 'Furnizor DEF', '0955552284', 'furnizor_def@gmail.com');`
- `insert into furnizori`
`values (13, 'Furnizor X', '0758852257', 'furnizor_x@gmail.com');`

5.2.7 Introducerea datelor în tabelul ECHIPAMENTE

- `insert into echipamente values (100, 'Laptop', 'Tehnologie', 6000);`
- `insert into echipamente values (101, 'Set Lego', 'Jocuri Educationale', null);`
- `insert into echipamente values (102, 'Proiector', 'Tehnologie', 2200);`
- `insert into echipamente values (103, 'Puzzle', 'Jocuri Educationale', null);`
- `insert into echipamente values (104, 'Masa', 'Mobilier', 700);`
- `insert into echipamente values (105, 'Scaun', 'Mobilier', 500);`
- `insert into echipamente values (106, 'Pat', 'Mobilier', 1200);`
- `insert into echipamente values (107, 'Dulap', 'Mobilier', 1000);`

5.2.8 Introducerea datelor în tabelul INSCRIERI

- `insert into inscrieri values (1000, 100, 50, 7, '2023-07-15', 14);`
- `insert into inscrieri values (1001, 100, 51, 7, '2023-07-16', 10);`
- `insert into inscrieri values (1002, 100, 52, 7, '2023-07-16', 10);`
- `insert into inscrieri values (1003, 100, 54, 7, '2023-09-10', 9);`
- `insert into inscrieri values (1004, 104, 53, 5, '2023-09-05', 11);`
- `insert into inscrieri values (1005, 102, 53, 5, '2023-09-06', 12);`
- `insert into inscrieri values (1006, 105, 53, 7, '2023-07-11', 10);`
- `insert into inscrieri values (1007, 105, 50, 7, '2023-07-11', 10);`
- `insert into inscrieri values (1008, 105, 51, 5, '2023-07-12', 15);`
- `insert into inscrieri values (1009, 105, 52, 7, '2023-07-12', 15);`

5.2.9 Introducerea datelor în tabelul REPARTIZARE_SALI

- `insert into repartizare_sali
values (1, 3, 6, '2022-09-05', '2023-06-25');`
- `insert into repartizare_sali
values (2, 9, 7, '2022-08-20', '2023-07-20');`
- `insert into repartizare_sali
values (3, 8, 3, '2022-09-05', '2023-06-25');`
- `insert into repartizare_sali
values (4, 1, 1, '2022-09-05', '2023-06-25');`
- `insert into repartizare_sali
values (5, 1, 2, '2022-09-05', '2023-06-25');`

5.2.10 Introducerea datelor în tabelul INVENTARE

- `insert into inventare
values (100, 4, 106, '2022-10-10', 35);`
- `insert into inventare
values (101, 1, 100, '2022-08-12', 2);`
- `insert into inventare
values (102, 7, 102, '2022-09-10', 2);`
- `insert into inventare
values (103, 8, 105, '2022-09-10', 5);`
- `insert into inventare
values (104, 9, 105, '2022-09-02', 4);`

5.2.11 Introducerea datelor în tabelul LIVRARI

- `insert into livrari
values (10, 100, 13, '2021-07-25', 5);`
- `insert into livrari
values (11, 102, 11, '2020-11-24', 4);`
- `insert into livrari
values (12, 105, 10, '2022-06-14', 10);`
- `insert into livrari
values (13, 104, 12, '2020-07-13', 40);`