

# Baza de Date Relațională a unui Service Auto



Candidat : Adrian Pop

Coordonator : Ș.l. dr. ing. Mădălin-Dorin Pop

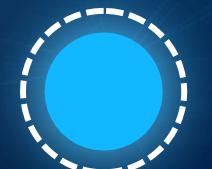
Ing. Daniela-Nicolia Pătruț

Sesiunea : Ianuarie 2024

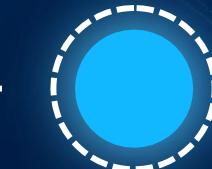
# Cuprins



Introducere  
și stadiu  
actual



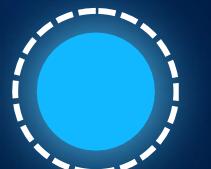
Bazele Teoretice



Soluția propusă și  
proiectarea

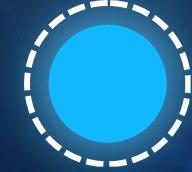


Implementare

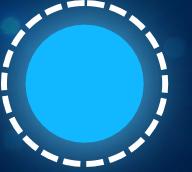


Concluzii

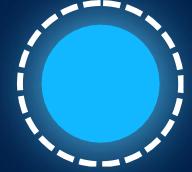
# Introducere și stadiu actual



Proiectul propune gestiunea rapidă și eficientă a datelor unui Service Auto



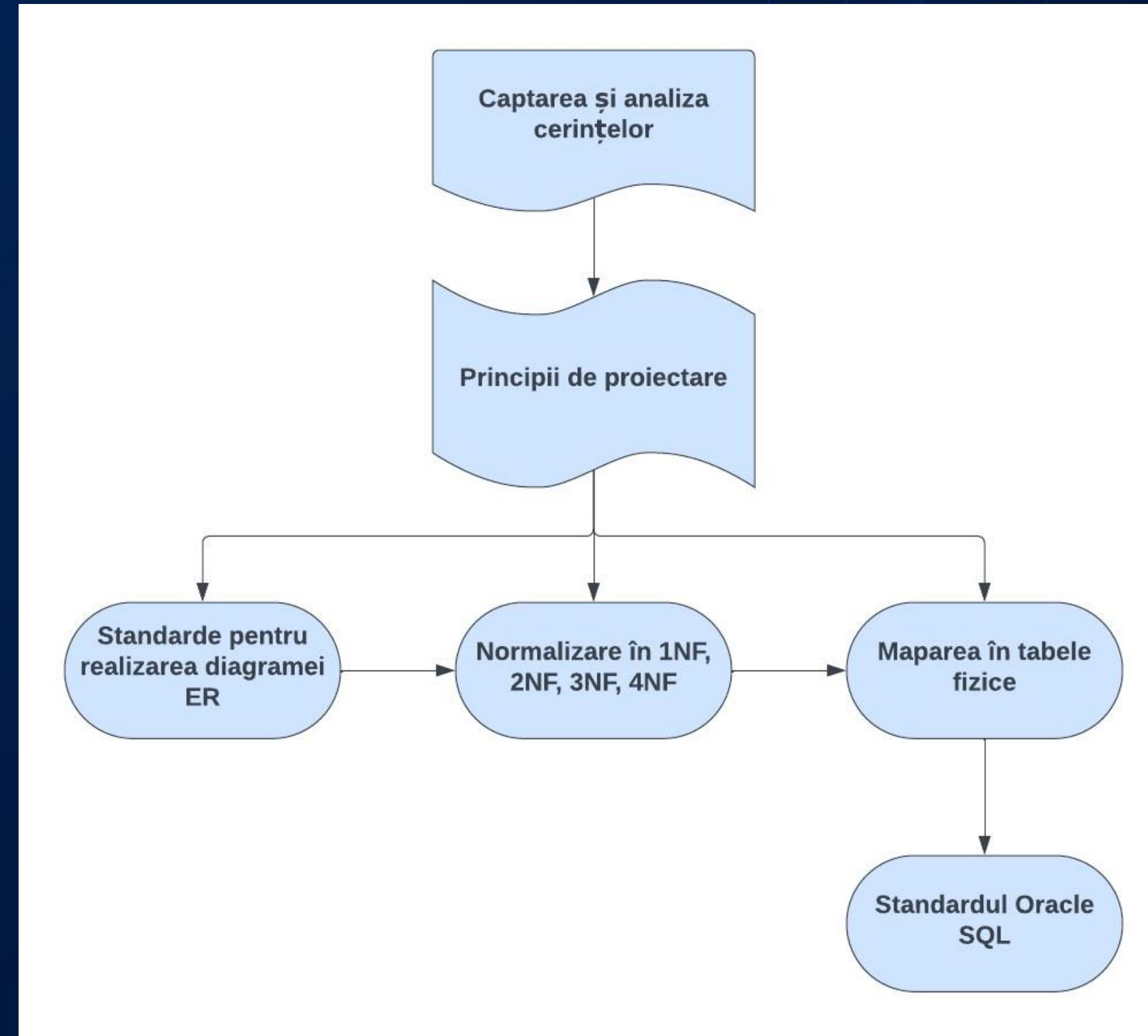
Pe parcursul realizării lucrării au fost aplicate cunoștințele acumulate în cadrul materiei



Produsul final reprezintă un model simplificat care poate funcționa bine la scară redusă și poate fi extins pe mai mult

Majoritatea service-urilor auto nu folosesc un sistem informational propriu de gestiune a datelor.

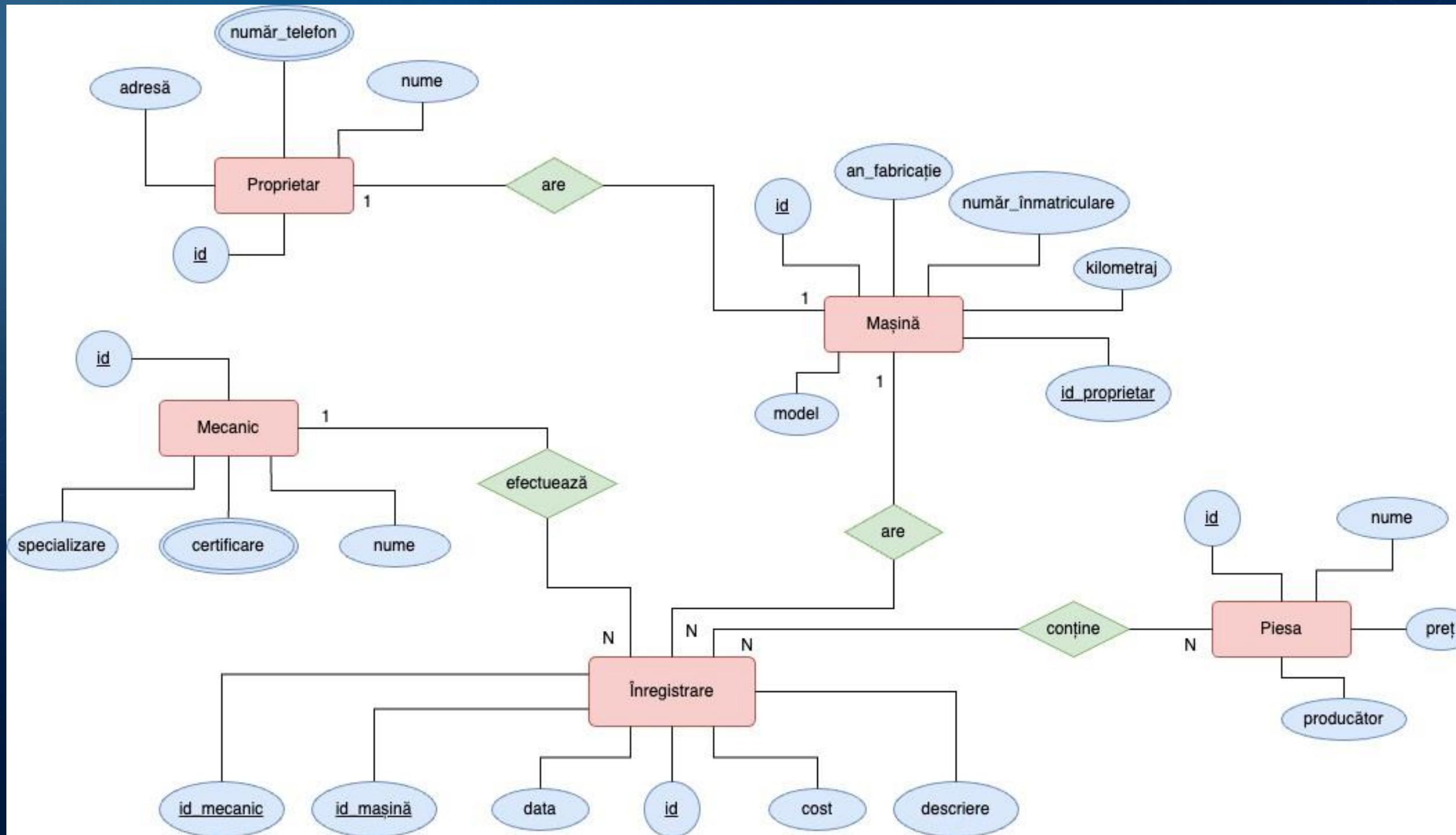
# Bazele Teoretice



Soluția propusă  
și proiectarea

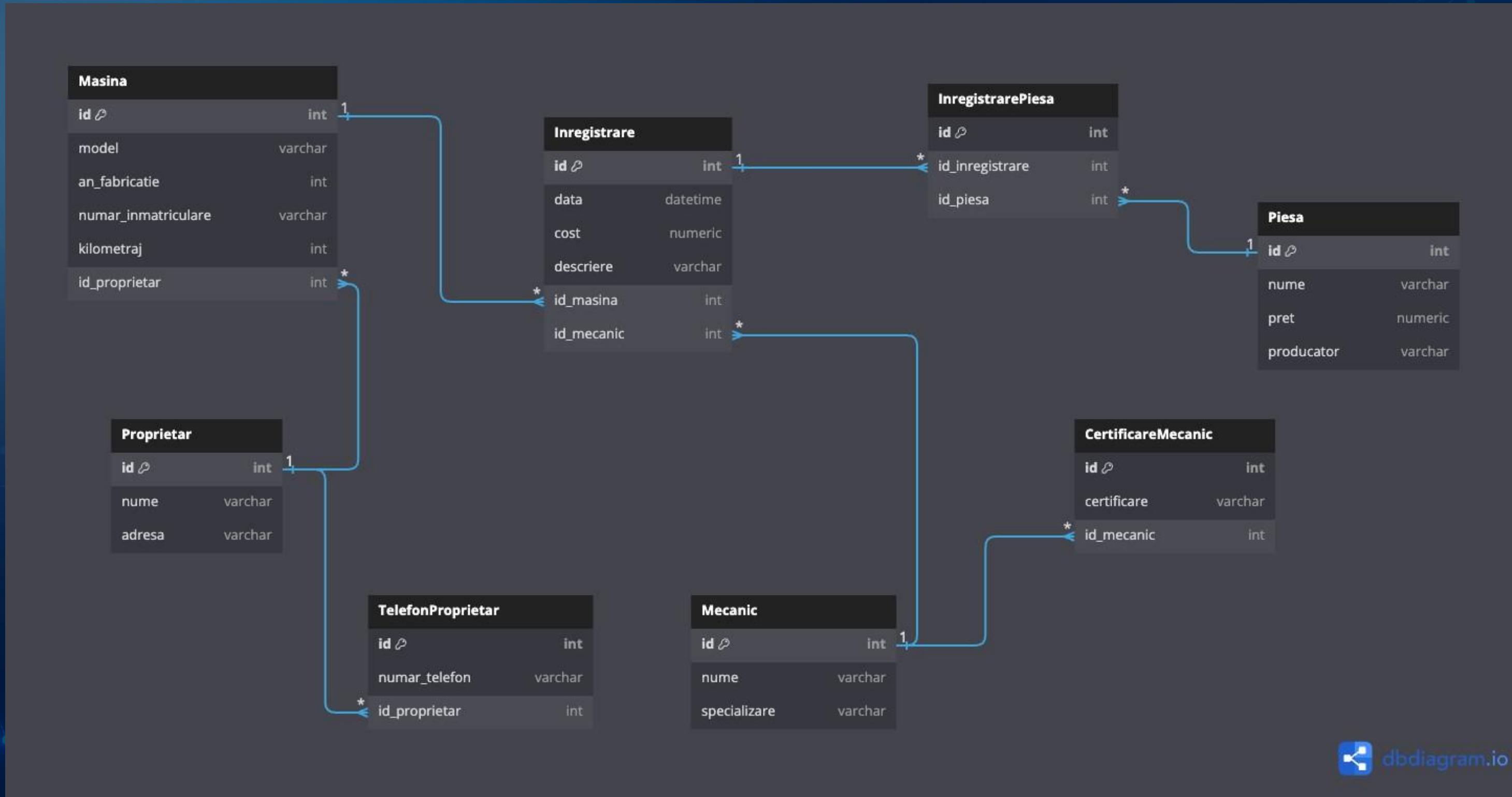


# Proiectarea conceptuală



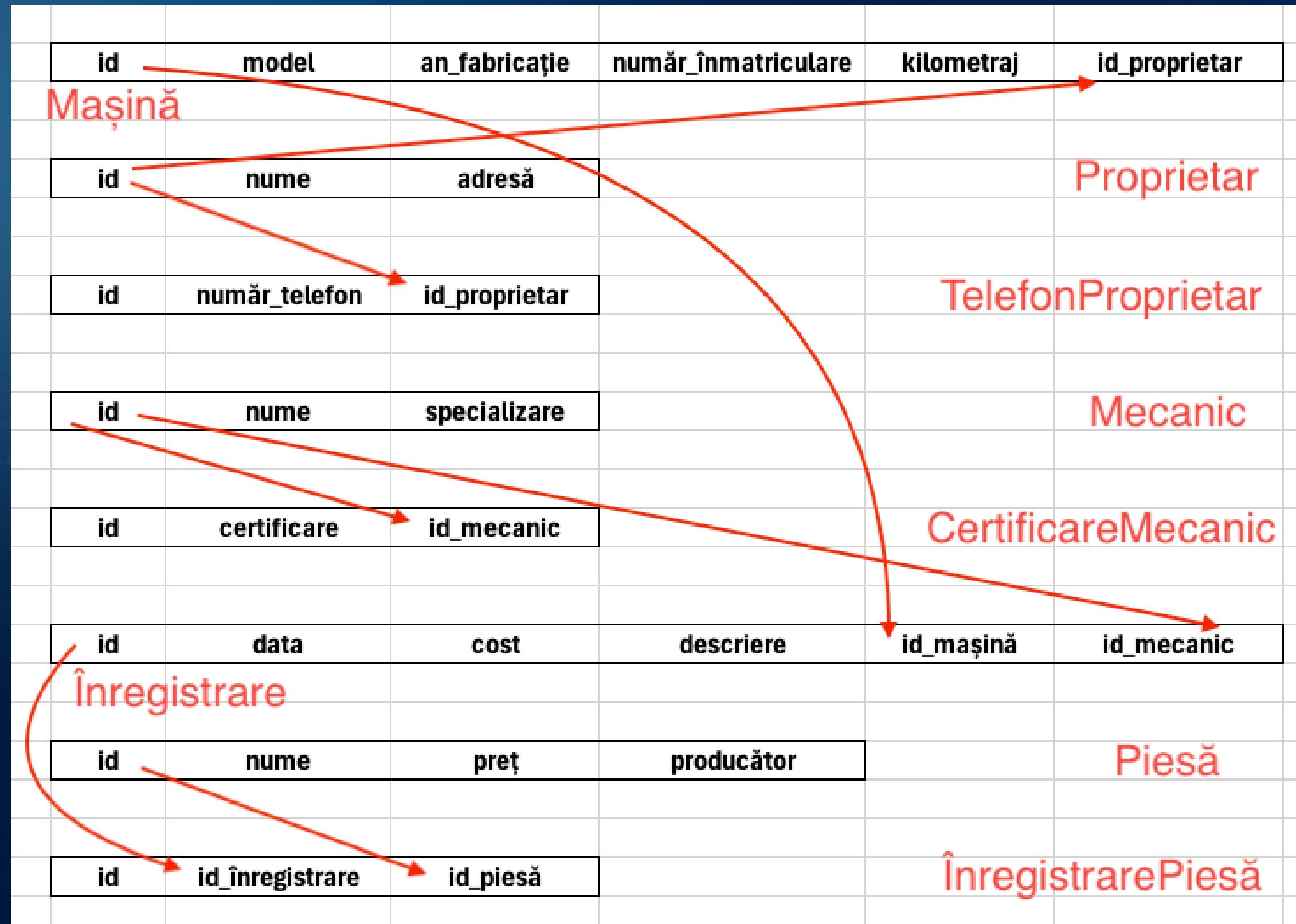
## Diagrama Entitate-Relație

# Proiectarea logică



Modelul logic obținut

# Proiectarea fizică



Modelul fizic obținut



# Implementare

# Crearea și popularea tabelelor

```
CREATE TABLE Masina (
    id NUMBER PRIMARY KEY,
    model VARCHAR2(255) NOT NULL,
    an_fabricatie NUMBER NOT NULL,
    numar_inmatriculare VARCHAR2(20) NOT NULL,
    kilometraj NUMBER NOT NULL,
    id_proprietar NUMBER NOT NULL,
    FOREIGN KEY (id_proprietar) REFERENCES Proprietar(id)
);
```

Column Name	Data Type	Nullable
ID	NUMBER	N
MODEL	VARCHAR2(255 BYTE)	N
AN_FABRICATIE	NUMBER	N
NUMAR_INMATRICULARE	VARCHAR2(20 BYTE)	N
KILOMETRAJ	NUMBER	N
ID_PROPRIETAR	NUMBER	N

```
INSERT INTO Masina (id, model, an_fabricatie, numar_inmatriculare, kilometraj, id_proprietar) VALUES (1, 'Renault Megane', 2018, 'PH-123ABC', 50000, 1);
INSERT INTO Masina (id, model, an_fabricatie, numar_inmatriculare, kilometraj, id_proprietar) VALUES (2, 'Volkswagen Golf', 2019, 'CJ-456XYZ', 60000, 2);
INSERT INTO Masina (id, model, an_fabricatie, numar_inmatriculare, kilometraj, id_proprietar) VALUES (3, 'Ford Focus', 2017, 'IS-789DEF', 75000, 3);
INSERT INTO Masina (id, model, an_fabricatie, numar_inmatriculare, kilometraj, id_proprietar) VALUES (4, 'Opel Astra', 2020, 'TM-101GHI', 40000, 4);
INSERT INTO Masina (id, model, an_fabricatie, numar_inmatriculare, kilometraj, id_proprietar) VALUES (5, 'Dacia Logan', 2016, 'DJ-202JKL', 80000, 5);
```

ID	MODEL	AN_FABRICATIE	NUMAR_INMATRICULARE	KILOMETRAJ	ID_PROPRIETAR
1	Renault Megane	2018	PH-123ABC	50000	1
2	Volkswagen Golf	2019	CJ-456XYZ	60000	2
3	Ford Focus	2017	IS-789DEF	75000	3
4	Opel Astra	2020	TM-101GHI	40000	4
5	Dacia Logan	2016	DJ-202JKL	80000	5

# Vederi

```
CREATE VIEW InformatiiInregistrare AS
SELECT *
FROM Inregistrare
WHERE id = 3;
```

```
CREATE VIEW InformatiiMasinaProprietar AS
SELECT
    m.id AS id_masina,
    m.model,
    m.an_fabricatie,
    m.numar_inmatriculare,
    p.id AS id_proprietar,
    p.nume AS nume_proprietar,
    tp.numar_telefon
FROM
    Masina m
JOIN Proprietar p ON m.id_proprietar = p.id
JOIN TelefonProprietar tp ON p.id = tp.id_proprietar
WHERE
    m.id = 3;
```

ID	DATA	COST	DESCRIERE	ID_MASINA	ID_MECHANIC
3	05/10/2023	200	Schimb placute frana	3	3

ID_MASINA	MODEL	AN_FABRICATIE	NUMAR_INMATRICULARE	ID_PROPRIETAR	NUME_PROPRIETAR	NUMAR_TELEFON
3	Ford Focus	2017	IS-789DEF	3	Adrian Ionescu	0734567890

# Proceduri

```
CREATE OR REPLACE PROCEDURE InsertPiesa(  
    p_id NUMBER,  
    p_nume VARCHAR2,  
    p_pret NUMBER,  
    p_producator VARCHAR2  
)  
AS  
BEGIN  
    INSERT INTO Piesa (id, nume, pret, producator)  
    VALUES (p_id, p_nume, p_pret, p_producator);  
    COMMIT;  
    DBMS_OUTPUT.PUT_LINE('Piesa adaugata cu succes!');  
EXCEPTION  
    WHEN OTHERS THEN  
        ROLLBACK;  
        DBMS_OUTPUT.PUT_LINE('Eroare la adaugarea piesei: ' || SQLERRM);  
END InsertPiesa;  
/  
  
BEGIN  
    InsertPiesa(6, 'Baterie auto', 120, 'Varta');  
END;  
/
```

BEGIN     InsertPiesa(6, 'Baterie auto', 120, 'Varta'); END;

Piesa adaugata cu succes!

Statement processed. 0.00 seconds

```
CREATE OR REPLACE PROCEDURE AfisareInformatiiInregistrare(  
    p_id_inregistrare NUMBER  
)  
AS  
BEGIN  
    FOR rec IN (SELECT i.id AS id_inregistrare,  
                  i.data,  
                  i.cost,  
                  p.nume AS nume_piesa,  
                  p.pret AS pret_piesa  
             FROM Inregistrare i  
             JOIN InregistrarePiesa ip ON i.id = ip.id_inregistrare  
             JOIN Piesa p ON ip.id_piesa = p.id  
             WHERE i.id = p_id_inregistrare)  
    LOOP  
        DBMS_OUTPUT.PUT_LINE('ID Inregistrare: ' || rec.id_inregistrare ||  
                             ', Data: ' || rec.data ||  
                             ', Cost: ' || rec.cost ||  
                             ', Nume Piesa: ' || rec.nume_piesa ||  
                             ', Pret Piesa: ' || rec.pret_piesa);  
    END LOOP;  
END AfisareInformatiiInregistrare;  
/  
  
BEGIN  
    AfisareInformatiiInregistrare(1);  
END;  
/
```

BEGIN     AfisareInformatiiInregistrare(1); END;

ID Inregistrare: 1, Data: 03/15/2023, Cost: 300, Nume Piesa: Placute frana, Pret Piesa: 100

Statement processed. 0.02 seconds

# Avantaje și Dezavantaje



+

Complexitatea redusă a modelului duce la timp mici de răspuns => Eficiență crescută

+

Majoritatea principiilor teoretice au fost aplicate => modelul poate fi extins

-

Nu sunt expuse toate relațiile ce ar putea fi modelate într-un service auto.

# Concluzii



Proiectul și-a atins scopul de a modela cât mai eficient o bază de date pentru un service auto

Eficiența și performanțele bazei de date se datorează principiilor de proiectare aplicate.

Există mai multe perspective din care produsul obținut poate fi scalat și îmbunătățit.

vă  
multumesc!