**Proiect sisteme de gestiune a bazelor de date**

**Informatii despre marci si modele de masini**

**Profesor:Vladoiu Monica**

**Studenta:Podeanu Stefania Adriana**

**Grupa:40317B**

**An II Sem II**

**Introducere**

**Tema proiectului:**

Sa se creeze o baza de date care contine informatii despre marci si modele de masini.

**Cerintele proiectului:**

1. Sa se construiasca diagrama entitate-legatura pentru aceasta baza de date, daca ati ales sa construiti o baza de date noua, sau sa se corecteze diagrama construita in sem. I, conform cu observatiile facute individual, la examen.
2. Sa se translateze baza de date contruita in semestrul I folosind suportul MySQL pentru baze de date tranzactionale.
3. Sa se afiseze relatiile (tabelele) din baza de date in cate pagini web este nevoie folosing un limbaj de control.
4. Sa se ofere posibilitatea realizarii de interogari care exemplifica operatiile de baza ale algebrei relationale (vezi tema BD din sem. I: cite o operatie pentru reuniune - 0.5 puncte, diferenta- 0.5 puncte, selectie - 1 punct, proiectie - 1 punct, si doua operatii de jonctiune - 2x1 punct).  
   Operatiile respective vor fi efectuate in cadrul unor interograri obisnuite asupra bazei de date, cum ar fi adaugari, stergeri, cautari si modificari de inregistrari, care au sens in contextul aplicatiei respective.
5. a. Se vor aduce toate relatiile (tabelele) din baza de date in FN3 si, acolo unde este cazul, si in FNBC.

**Descrierea problemei**

***Ce doreste utilizatorul?***

Utilizatorul doreste ca informatiile despre toate tipurile de masini sa fie accesibile pentru a putea realize diferite comparatii intre diferite marci si modele de masini pe baza unor criterii esentiale cum ar fi: numele marcii si a modelului, anul de fabricatie sau anul de incetare a productiei, pe baza culorilor sau a gradului de opacitate/metalizare a masini,anumite dotari sau tipuri de caroserii etc. care ar putea sta in topul intereselor pentru o posibila cumparare, inchiriere sau chiar a unui posibil parteneriat cu alte firme producatoare.

***Ce se urmareste?***

Utilizatorul urmareste sa redea in mod real,rapid si eficient detalii despre fiecare masina astfel incat acestea sa fie prezentate intr-o lumina clara.

***Care este problema?***

Problema este reprezentata de o dezorganizare a informatiilor, o cautare ineficienta si uneori chiar informratii incomplete despre anumite marci sau modele de masini.

***Care este*** ***solutia problemei?***

Solutia problemei o reprezinta implementarea unei baze de date in care se va structura informatiile esentiale despre fiecare marca si model cu caracteristicile specifice.

***Pe cine ajuta?***

Aceasta baza de date ajuta orice utilizator intresat de domeniul autoturismelor, deoarece informatia regasita este fragmentata in functie de ariile principale (criteriile principale) de interes ale utilizatorului cum ar fi: numele marcii si a modelului, anul de fabricatie sau anul de incetare a productiei, pe baza culorilor sau a gradului de opacitate/metalizare a masini,anumite dotari sau tipuri de caroserii etc.

**Diagrama entitate-legatura(ED)**

Pentru a solutiona problema ceruta de utilizator se va trece la etapa de proiectare a bazei de date conceptuale, in care informatiile adunate in faza de analiza sunt folosite pentru a dezvolta in continuare o descrire la un nivel abstract a datelor dn baza de date si a constrangerilor valabile pentru aceasta .

Scopul principal al acestei etape este de a creea o descriere simpla a datelor de care se va tine cont de criteriile principale de alegere a unui autoturism din punctul de vedere al oricarui utilizator.Pentru aceasta etapa s-a folosit modelarea Entitate-Legatura, cu ajutorul caruia am construit diagram asociata

Tabelele principale care fac referire la cerintele utilizatorului sunt **MARCI si MODELE**, iar tabelele adiacente care fac referire la diverse preferinte sunt: **DIMENSIUNI, MOTOARE, DOTARI, CULORI**. Cu atributele specifice fiecareia.

Tabelul pentru informatii din punct de vedere al posibil parteneriat a fost considerat **SEDII, PRODUCTII** si **ISTORIC\_VANZARI**.

De asemenea, putem preciza faptul ca fiecare entitate cu exceptia entitatii incomplete TIPURI CAROSERII contine cate o cheie primara care determina o mai buna cautare a unei anumite marci/model unice cu anumite specificatii caracteristice.

**Multimi Entitati- descrieri si legaturi**

***Multimea entitate MARCI*** se refera la tipurile de marci existente pe piata automobilelor (Audi,BMW , Renault etc.) are mai multe atribute specifice dintre care in figura am inclus : **id\_marca: (cheie primara), an\_incepere\_marca, nume\_marca.**

***Multimea entitate MODELE*** descrie pentru fiecare marca existenta pe piata, modelele corespunzatoare acesteia .De exemplu: marca audii are mai multe modele cum ar fi: A4,A5 etc.In diagrama am inclus:  **id\_model (cheie primara), nume model.**

***Multimea entitate MOTOARE***  cotureaza faptul ca mai multe tipuri de motoare pot exista pe mai multe marci, acesta entitate este desenata cu atributele specifice pentru fiecare motor astfel incat sa se faca o diferentiere clara**: cod\_motor(cheie primara), tip\_motor(benzina/diesel), putere, distributie, nr cilindrii.**

Dotarile fiecarei masini sunt tratate intr-o entitate separata denumita ***DOTARI*** care are rolul de a pune in atentia utilizatorului ca mai mai multe dotari specificate pot fi pe mai multe modele.Aceasta entitate are ca atribute : **id\_dotari(cheie primara),tip\_dotare.**

De asemenea ***entitatea TIPURI CAROSERIE*** descrie **numele caroseriei**, dar si **anul platformei** masinii(unele masini au un format diferit in functie de anul fabricatiei) , aceasta neavand cheie primara fiind o **entitate-incompleta.**Avand in vedereca exista mai multe tipuri de caroserii s-a luat in considerare ca, unei caroserii pot sa ii corespunda mai multe dimensiuni, iar aceleasi dimensiuni pot corespunde mai multor caroserii.

Datorita acestui fapt s-a implementat ***entitatea DIMENISIUNI*** cu mai multe attribute, cum ar fi: **id\_dimensiuni(cheie primara),greutate, volum portbagaj,inaltime, latime,lungime.**

***Multimea entitate SEDII*** desemneaza toate informatiile importante referitoare la contact si adresa sediului.Aceasta entitate este unica pentru fiecare marca descrisa, iar fiecare marca are un sediu principal.SEDII contine atributele: **cod\_sediu(cheie primara),nr\_contact,site\_web,localitate**.

Am luat in considerare faptul ca fiecare marca are un specific de culori, o marca nu poate avea mai multe culori, dar aceeasi culoare poate corespunde mai multor marci.

**Multimea entitate CULORI** are ca atribute : **cod culoare**(**cheie primara),grad culoare, nume culoare**.De asemenea, aceasta entitate este dispusa intr-o **ierarhie *ISA cu multimile entitate:*** ***OPACE*** si ***METALIZATE*** fiecare avand caracteristici diferite:

Pentru **Entitate Opace** :

**id\_opacitate(cheie primara),grad\_opacitate,grad\_luciu.**

**Pentru Entitatea Metalizate:**

**id\_metalzare(cheie primara),grad\_metalizare.**

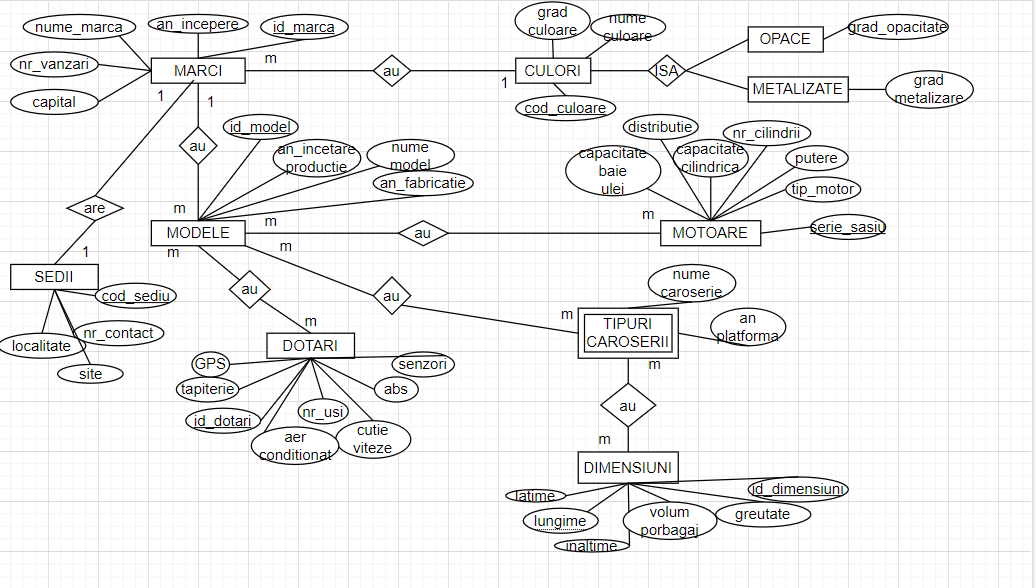
**Etapa a II-a**

Etapa de dezvoltare a proiectului cu **TEMA INFORMATII DESPRE MARCI SI MODELE de MASINI** aadus modificari asupra bazei de date proiectate initial si implicit a diagramei entitate legatura.

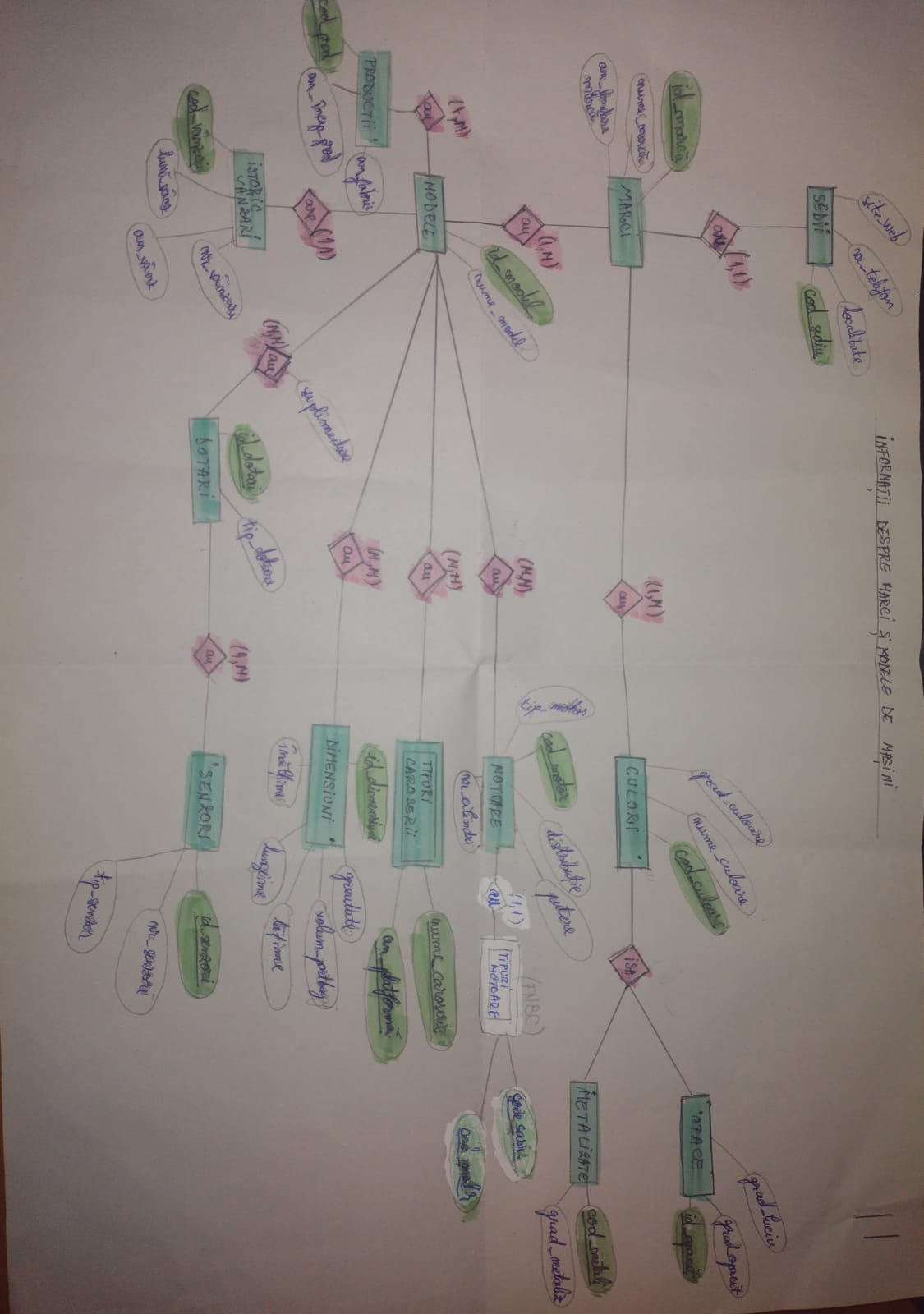
Pentru a avea o viziune mai clara asupra modificarilor am considerat de adaugat ambele diagrame.

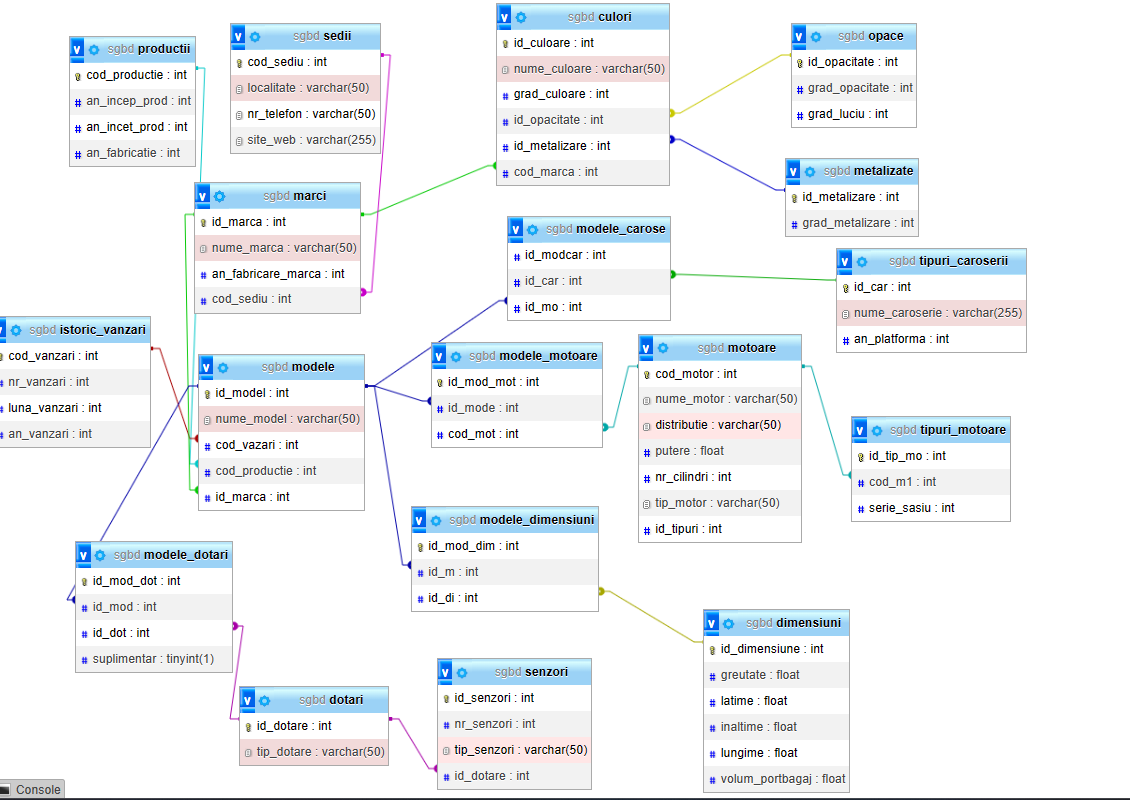
Aceasta etapa (II) de dezvoltare si corectare a acestui proiect a fost construita pe baza regulilor si a formelor normale existente la nivelul bazelor de date: **FN1, FN2,FN3,FNBC.**

**Diagrama Entitatate-Legatura(figura 1 ,etapa1)**



**Diagrama Entitate-Legatura(figura2, etapa2)**

****

****

**Forme Normale FN1,FN2,FN3(aplicate pentru baza de date)**

**FN1**

O relatie este in FN1 daca orice camp are doar valori atomice, adica nu poate fi constituit din liste sau multimi. In plus, un tuplu nu trebuie sa contina atribute sau grupuri de atribute care se repeta.

Procedura de constructire relatii FN1

Pentru a construi relatii care sa respecte restrictiile FN1 de fiecare data cand este necesar un atribut multivaloare sau un grup repetitiv de atribute se va contrui o relatie separata in care sa includ datele continute de atributul multivaloare sau de grupul repetitiv intr-o relatie separata, legatura facandu-se prin intermediul cheii straine.

Deoarece, in etapa initiala de proiectare nu am avut liste sau multimi.Normalizarea pentru **FN1** a fost respectata implicit.

**FN2**

O relatie este in FN2, daca este in FN1 si daca oricare dintre atributele non-cheie e dependent functional de intreaga cheie primara (de toate atributele cheii primare).

Conditia ca o relatie sa fie in FN2 este incalcata daca un camp non-cheie ne da informatii despre un subset al unei chei, acest lucru fiind relevant pentru chei compuse.

Pentru aceasta normalizare am eliminat atributele care nu erau dependente de cheia primara in intregime:

In entitatea **Marci** am eliminat **nr\_vanzari** si **capital**.Am creat alta entitate denumita **Istoric\_Vanzari** care contin mai multe detalii.Atributul capital fiind eliminat complet.

In entitatea **Modele** am eliminat atributele **an\_incetare\_productie** si **an\_fabricare**, acestea fiind puse intr-o alta entitate denumita **Productii** care are **cod\_productie** ( **cheie primara)** si depind functional doar de aceasta.

**FN3**

O relatie este in FN3 daca este in FN2 si daca atributele non-cheie nu sunt dependente tranzitiv de cheia primara a relatiei, ci direct. (Nu este permis ca un atribut non-cheie sa depinda de un camp care nu este cheia primara).

Conditia pentru ca o relatie sa fie in FN3 nu este respectata daca un camp non-cheie ne da informatii despre un alt camp non-cheie.

Pentru **FN3** in entitatea **Motoare** am regasit cateva atribute care depindeau unele de celelalte : **nr\_cilindri si capacitate\_cilindrica**, ambele avand aceeasi semnificatie in cee ace priveste capacitatea cilindirilor.Am eliminat atributul **capacitate\_cilindrica**, deoarece **nr\_cilindri** este determinat de **cod\_motor** care este **cheie primara** in aceasta entitate.

**FNBC**

O relatie este in FNBC daca dependentele functionale netriviale care se manifesta in cadrul relatiei contin in partea dreapta ca determinant o super-cheie.

Partea stanga a oricarei dependente functionale netriviale trebuie sa fie o supercheie (sa includa o cheie).

Pentru ca o relatie sa fie in zona de incidenta a FNBC trebuie sa fie indeplinite urmatoarele conditii:

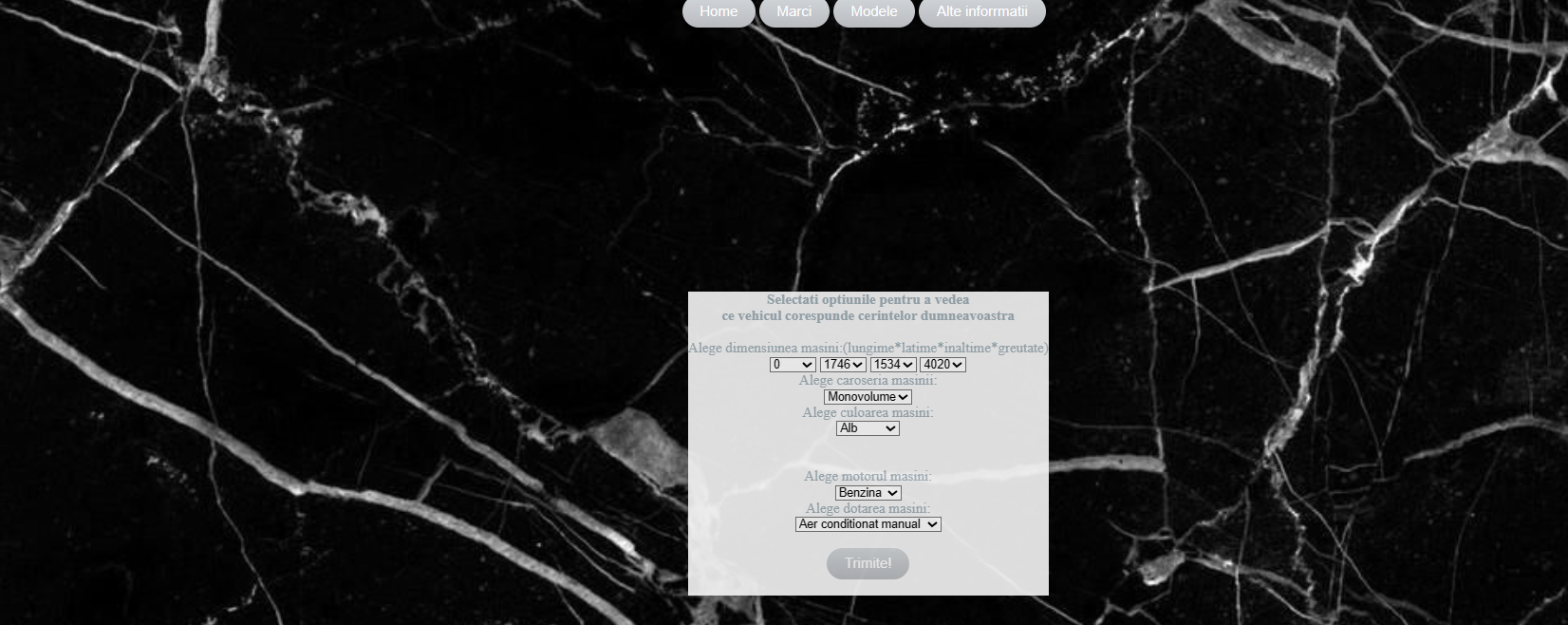
Sa aiba cel putin dpua chei canditat, care sa se suprapuna (pe unele dintre atribute) si cel putin una din cele doua chei sa fie compusa.

Cel mai simplu mod de a aborda FNBC este prin intelegerea faptului ca nu trebuie sa existe dependente functionale intre cheile candidat.

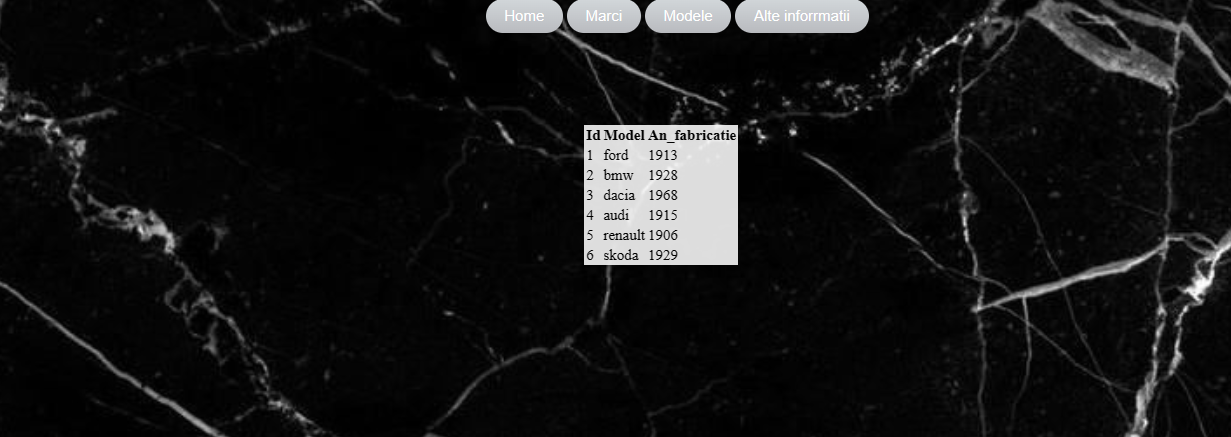
In cadrul bazei de date am intalnit cazul unei super-chei formate din doua atribute serie\_sasiu si cod\_motor.Prima furnizand informatii cu privire la toate caracteristicile specific unei masini, incluzand si caracteristici ale motorului.Cea de-a doua cod\_motor include caracteristici specifice motorului.Din aceasta cauza, am considerat pastrarea acestora si spargerea in doua entitati separate: Motoare cu cod\_motor (cheie primara),tip\_motor(benzina/motorina),distributie,putere,nr\_cilindri si Tipuri\_Motoare(entitate incomplete) cu chei primare (serie\_sasiu si cod\_motor).

Crearea paginilor web

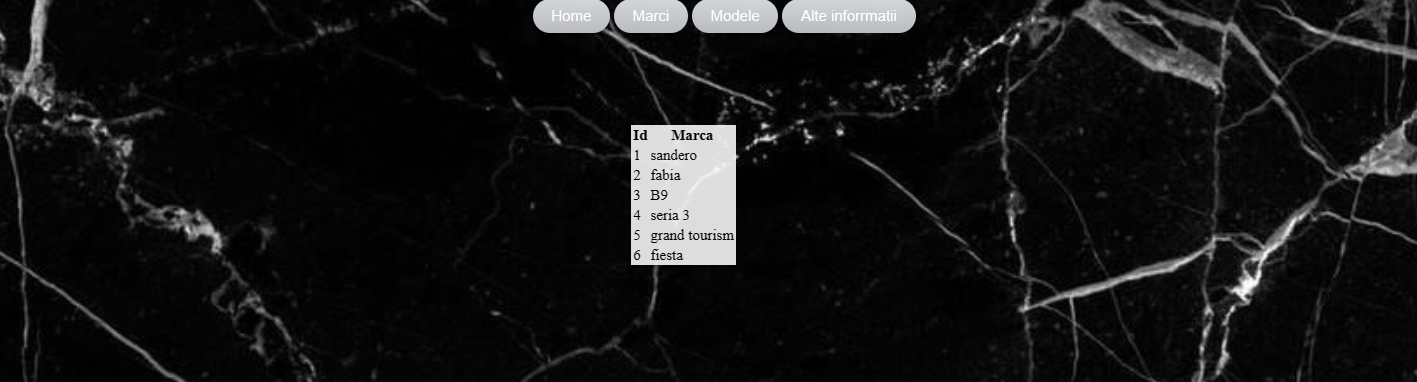
Home:



Pagina Marci:

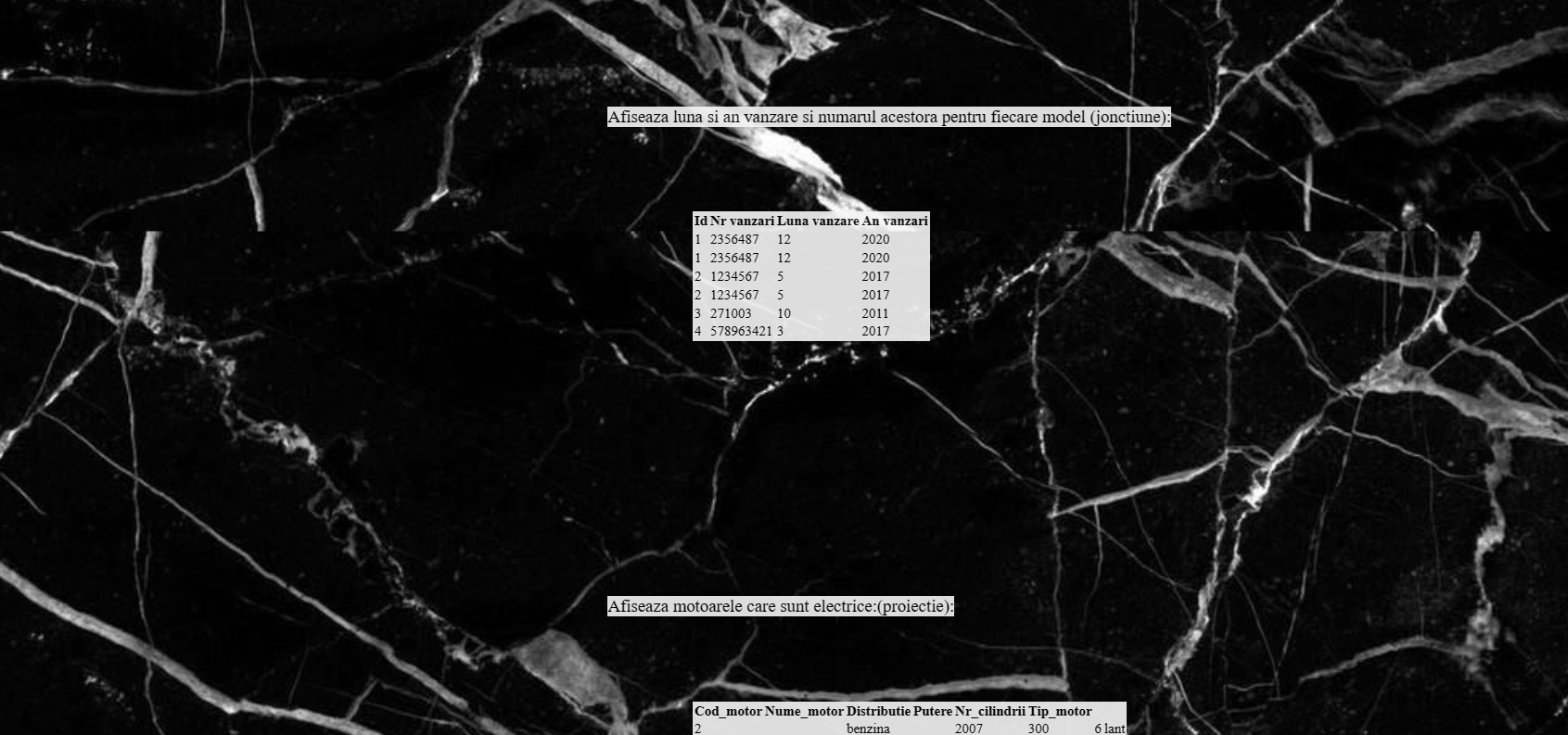


Pagina Modele:



Pagina Alte Informatii:



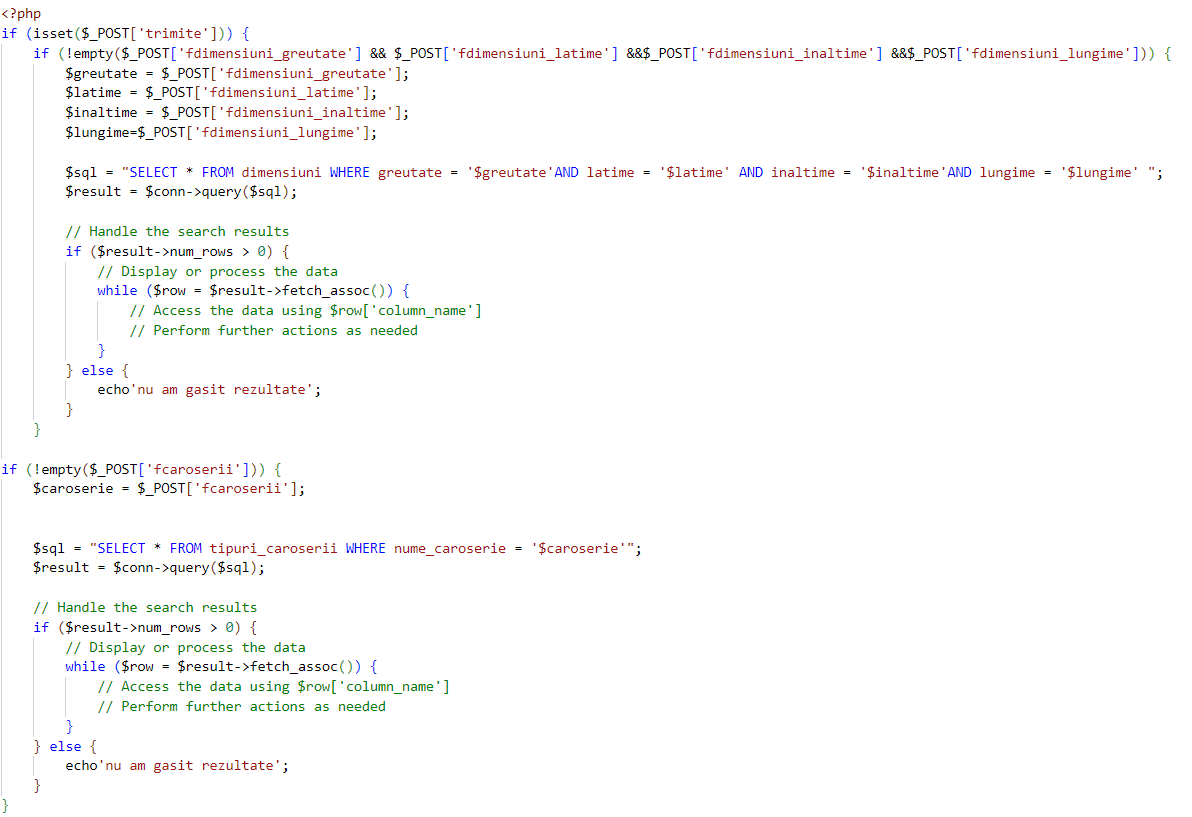


In cadrul acestor pagini am accesat datele introduse in baza de date locala:

**Optiunile:**



**Relatia dintre formular si baza de date:**



Operatii din algebra relationala

**Selectia**

***Cerinta 1: Utilizatorul doreste sa afiseze numele marcii si primul an al fabricarii acesteia pentru anul 1906.Pentru cunoasterea istoricului activarii aceteia pe piata automobilelor.***

Operatie in algebra relationala:

**∏ nume\_marca (Marca)Ꝺ(an\_fabricatie=’1906’) (Marca)**

****

**Reuniunea**

***Cerinta 2: Utilizatorul doreste sa afiseaze numarul de vanzari, luna si anul pentru fiecare model vandut.***

Operatie in algebra relationala:

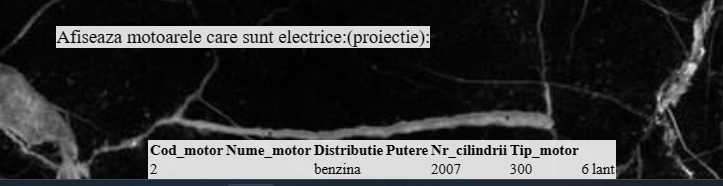
**Π nr\_vanzari,luna\_vanzari,an\_vanzari(ρ istoric\_vanzari(Istoric\_vanari) ∪ ρ modele (Modele))**

**Proiectie**

**Cerinta 3: Utilizatorul doreste sa afiseze codul motor pentru care tipul de motor este ‘electric’.**

Operatie in algebra relationala:

***Π cod\_motor(σ tip\_motor = "electric" (Motoare))***

******

***Diferenta***

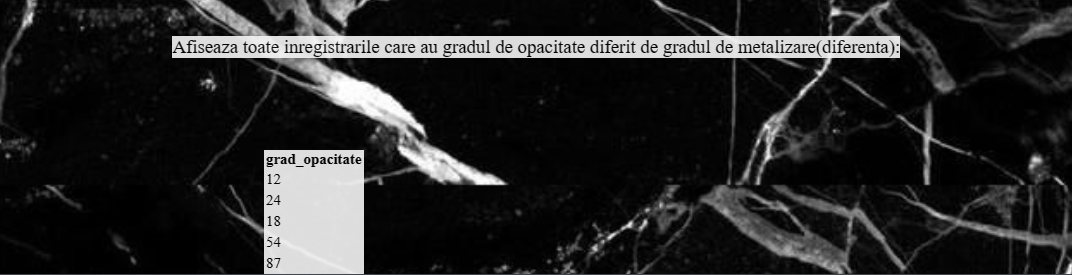
***Cerinta 3: Utilizatorul doreste sa se afiseze toate inregistrarile care au gradul de opacitate diferit de gradul de metalizare.***

***Operatie in algebra relationala:***

**Π grad\_opacitate (σ ¬∃grad\_Metalizare (Metalizate) (ρ opace (Opace)))**

***Instructiune Mysql:***

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) grad\_opacitate FROM opace as opa where [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) exists ([select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) grad\_metalizare from metalizate as meta where(opa.grad\_opacitate=meta.grad\_metalizare));

******

***Jonctiuni***

***Cerinta 5: Utilizatorul doreste sa afiseze pentru fiecare marca sediul corespunzator.***

***Operatie in algebra relationala:***

**Πlocalitate,nr\_telefon ,site\_web(Sedii×Marci|cod\_sediu =cod\_sediu)**

**Instructiune MySql:**

SELECT \* FROM sedii INNER JOIN marci where marci.cod\_sediu=sedii.cod\_sediu;

****

**Jonctiune**

***Cerinta 6:***

***Utilizatorul doreste sa afiseaze luna si anul vanzari si numarul acestora  pentru fiecare model (jonctiune)***

***Operatie in algebra relationala:***

**Π luna\_vanzari,an\_vanzari,nr\_vanzari (Istoric\_vanzari×Model|cod\_vanzari =cod\_vanzari)**

Interogare MySql

SELECT \* FROM istoric\_vanzari INNER JOIN modele  where modele.cod\_vanzari=istoric\_vanzari.cod\_vanzari;

