MINISTERUL EDUCAŢIEI ŞI CERCETĂRII ŞTIINŢIFICE

UNIVERSITATEA PETROL – GAZE DIN PLOIEŞTI

FACULTATEA LITERE ŞI ŞTIINŢE

DEPARTAMENTUL INFORMATICĂ, TEHNOLOGIA INFORMAŢIEI, MATEMATICĂ ŞI FIZICĂ

SPECIALIZAREA INFORMATICĂ

CURSURI DE ZI

**PROIECT LA DISCIPLINA BAZE DE DATE**

TEMA: Dezvoltarea unei baze de date a unui magazin cu produse pentru animale de companie.

|  |  |
| --- | --- |
|  |  |
|  | Student:  Manu Ioan |

PLOIEŞTI 2021

CUPRINS

Nevoia de aplicație pg3

Diagrama Entitate-Legătură pg5

Implementarea bazei de date în MySQL pg8

Operații asupra bazei de date pg12

1.Nevoia de aplicație

Baza de date prezentată vine în ajutorul magazinelor de dimensiuni medii care se ocupă cu comercializarea de produse pentru animale de companie.

Înainte de a se găsi o metodă eficientă de grupare a elementelor unui magazin de acest tip, se foloseau metodele tradiționale de grupare a departamenteler, angajaților sau a produselor folosindu-se diferite materiale în care se scriau aceste aspecte care duceau la funcționarea magazinului.

Deoarece volumul de produse prezente în magazin, schimbăriile periodice de angajați, furnizori sau de produse era din ce în ce mai pronunțat și managerii de magazin nu mai făceau față corespunzător, s-a introdus metoda electronică, prin construirea unei aplicații care gestionează computerizat toate informațiile specifice magazinului.

A urmat etapa de analiză a cerințelor, care constituie primul pas în proiectarea unei aplicații cu baze de date și are scopul de a înțelege ce date vor fi stocate în baza de date, cum se vor utiliza acestea în viața reală sau care sunt operațiile cele mai des realizate, fiind vorba în mare parte de ceea ce vor utlizatorii de la aplicația cu bază de date.

Cerințele prezente în formarea unui sistem care să administreze un magazin cu produse pentru animale de companie au fost:

-să ofere detalii specifice despre produsele prezente în magazin: descrierea, prețul sau numărul de produse găsite. Toate acestea fiind grupate pe departamente specifice.

-detalii referitoare la animalele cărora le sunt adresate produsele respective: o scurtă descriere, tipul de animal.

-informații specifice ale furnizorilor magazinului: adresa, nume companie, telefon.

-prezintă angajații magazinului cu detalii specifice: numele angajatului, vechimea în magazin, experiența pe câmpul muncii, salariul fiecăruia, vârstă, studiile și rolul specific pe care îl prezintă în magazin.

-repartiția pe departamente a fiecărui angajat, un telefon pentru fiecare departament în cazul în care apare o problemă specifică unuia dintre ele și păstrarea în memorie a managerilor fiecărui departament în parte.

-să ofere posibilitate de căutări rapide după diverse criterii:

* Cod
* Marcă
* Numărul de produse
* Preț

-să implementeze un sistem de coduri de bare pentru produse, pentru o găsire mai ușoară a diverselor informații specifice produsului respectiv.

-să reușească să asocieze produsele disponibile cu furnizorii specifici.

2. Diagrama Entitate-Legătură

O dată ce au fost stabilite cerințele pentru aplicația cu baze de date, urmează etapa de proiectare a bazei de date conceptuale, în care informațiile adunate în faza de analiză se prezintă, în mod abstract, cu ajutorul unei diagrame entite-legătură, o ilustrare grafică a descrierii bazei de date conceptuale.(figura 2.1).

Pentru a facilita înțelegerea diagramei entitate legătură , unele dintre atributele diverselor mulțimi entitate au fost ignorate, nefiind ilustrate în desen.

Se găsesc următoarele mulțimi entitate: DEPARTAMENTE, ANGAJATI, MANAGERI, PRODUSE, ANIMALE și FURNIZORI. Fiecare dintre acestea are atribute specifice care se deduc prin propriul nume.

Mulțimea entitate ANGAJAȚI prezintă cheia primară: id\_ang și reușește să prezinte toate informațiile unui angajat precum: vârstă, vechimea, experiența, salariul pe care îl are sau rolul său în magazin.

Mulțimea entitate DEPARTAMENTE figurează cu cheia primară: id\_dep și cheia candidat: nume\_dep. De asemenea, se pot afla informații în legătură cu numărul de angajați prezentat de un departament sau numărul de telefon al respectivului departament.

Mulțimea entitate MANAGERI prezintă numele managerului specific unui departament și bonusul aferent pe care îl căștiga având acest rol în magazin. Aceasta are ca cheie primară: id\_man.

Mulțimea entitate PRODUSE este caracterizată de o cheie primară: cod care poate face referire și la codul de bare al produsului respectiv. Aceasta oferă informații cu privire la numărul de produse, descrierea produsului, marca sau prețul unui anumit produs.

Mulțimea entitate ANIMALE prezintă informațiile unui anumit animal specific unui anumit produs, având ca cheie primară: id\_animal și alte atribute precum: tipul animalului sau descrierea acestuia.

Mulțimea entitate FURNIZORI face referire la informațiile fiecărui furnizor al magazinului, având că cheie primară: id\_f și atributele: adresa, nume companie sau număr de telefon.

Între aceste mulțimi entitate există diverse legături pe care le vom prezenta în continuare. Mai întăi avem legătura lucreaza\_in dintre ANGAJATI și DEPARTAMENTE, care ne arată că fiecare angajat poate fi încadrat într-un singur departament și fiecare departament poate avea mai mulți angajați, deci este o legătură unu-multi.

Legătură conduse\_de este o legătură unu-unu pentru că un departament poate avea un manager iar un manager poate avea acest rol pentru un singur departament.

Legătură prezintă dintre DEPARTAMENTE și PRODUSE, ne arată că fiecare produs aparține unui singur departament dar un departament poate avea mai multe produse având o legătură unu-multi.

Legătură folosesc este o legătură multi-multi deoarece mai multe animale pot folosi un produs dar și un produs poate fi folosit de mai multe animale.

Legătura dintre PRODUSE și FURNIZORI ne arată că un furnizor poate furniza mai multe produse dar și un produs poate fi redat de către mai mulți furnizori. De asemenea, legătură furnizate\_de prezintă și două atribute proprii: data care face referire data la care s-au pus în stoc anumite produse și nr\_de\_prod care reprezintă numărul produselor furnizate specifice unui produs.

Toate legăturile au gradul binar iar tipul de apartenență este obligatoriu.

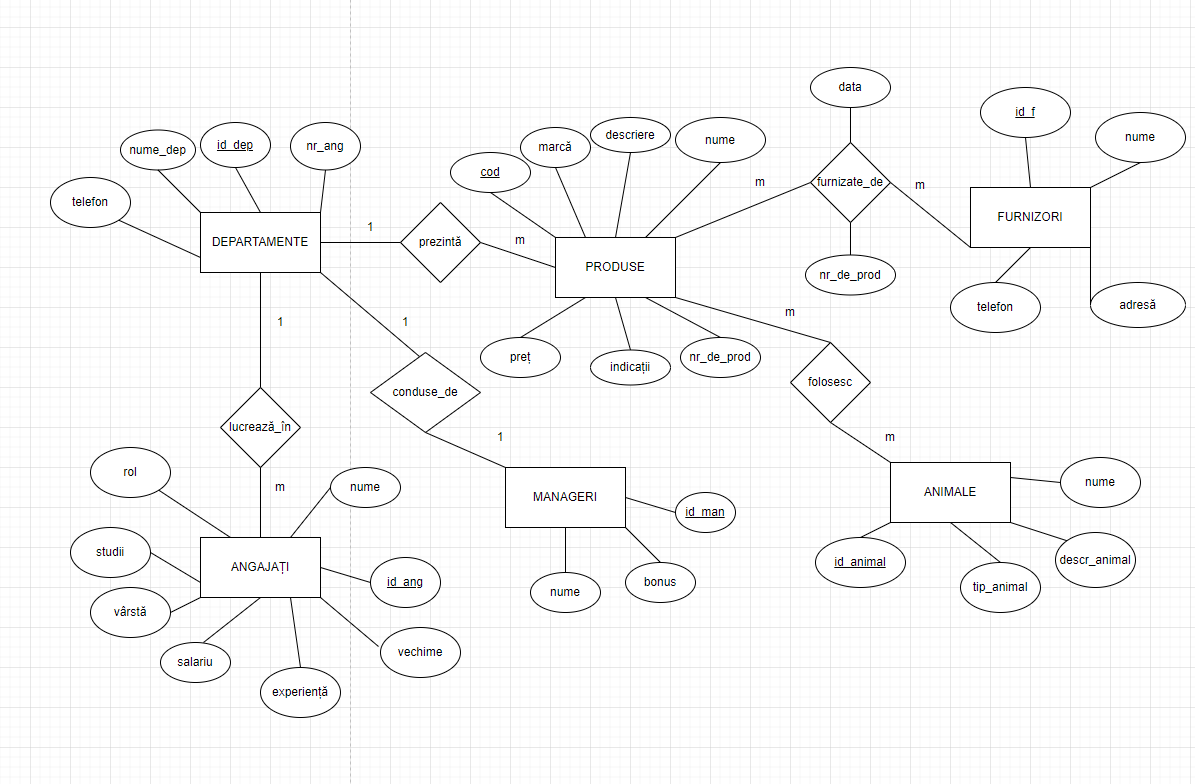


Figura 2.1

3. Implementarea bazei de date în MySQL

Diagrama entitate legătură obținută în etapa precedentă urmează să fie convertită într-o bază de date în format MySQL.

Pentru a face această translatare am folosit regulile prezentate în capitolul al treilea, care descriu modelul relațional. Pentru fiecare mulțime entitate se creează o relație, care va conține în schema sa toate atributele mulțimii entitate respective, iar cheia mulțimii entitate va deveni cheia relației.

Pentru fiecare entitate care se găsește la capătul ,,multi” al unei legături (sau al mai multora) unu-multi, se adaugă în tabela sa corespunzătoare, cheia primară a fiecărei entități de la capătul ,,unu” al legăturii respective, pe post de cheie străină. Fiecare legătură unu-multi a fost reprezentată prin includerea datelor despre ea într-una dintre relațiile corespunzătoare mulțimilor entitate care stau asociate în legătura respectivă.

În cazul multi-multi legătura a fost reprezentată în baza de date relațională ca relație separată. Formandu-se tabele corespunzătoare fiecărei legături.

În cazul unu-unu s-a ales, că în legătură multi-multi, reprezentarea prin includerea datelor despre ea într-una dintre relațiile corespunzătoare mulțimilor entitate care sunt asociate în legătură respectiva. În cazul de față, s-a pus cheia străină în tabelul manageri.

Respectând recomandările de mai sus, am obținut schema relațională a bazei de date care va conține informațiile funcționării eficiente a unui magazin de produse pentru animale de companie. În continuare vom prezenta schema relațională(în coloana stangă) pentru fiecare relație de mai sus și codul MySQL(în coloana dreapta) corespunzător fiecărei scheme în parte.

* DEPARTAMENTE(id\_dep, nume\_dep, nr\_ang, telefon) O imagine care conține text

  Descriere generată automat
* ANGAJATI(id\_ang, nume, vechime, experienta, salariu, varsta, studii, rol, *id\_dep*) O imagine care conține text

  Descriere generată automat
* PRODUSE(cod, nume, marca, descriere,pret, nr\_de\_prod, *id\_dep*) O imagine care conține text

  Descriere generată automat
* MANAGERI(id\_man,bonus, nume, *id\_dep*) O imagine care conține text

  Descriere generată automat
* ANIMALE(id\_animal, nume, tip\_animal, descr\_animal) O imagine care conține text, captură de ecran, persoană

  Descriere generată automat
* Folosesc(*cod*, *id\_animal*) O imagine care conține text

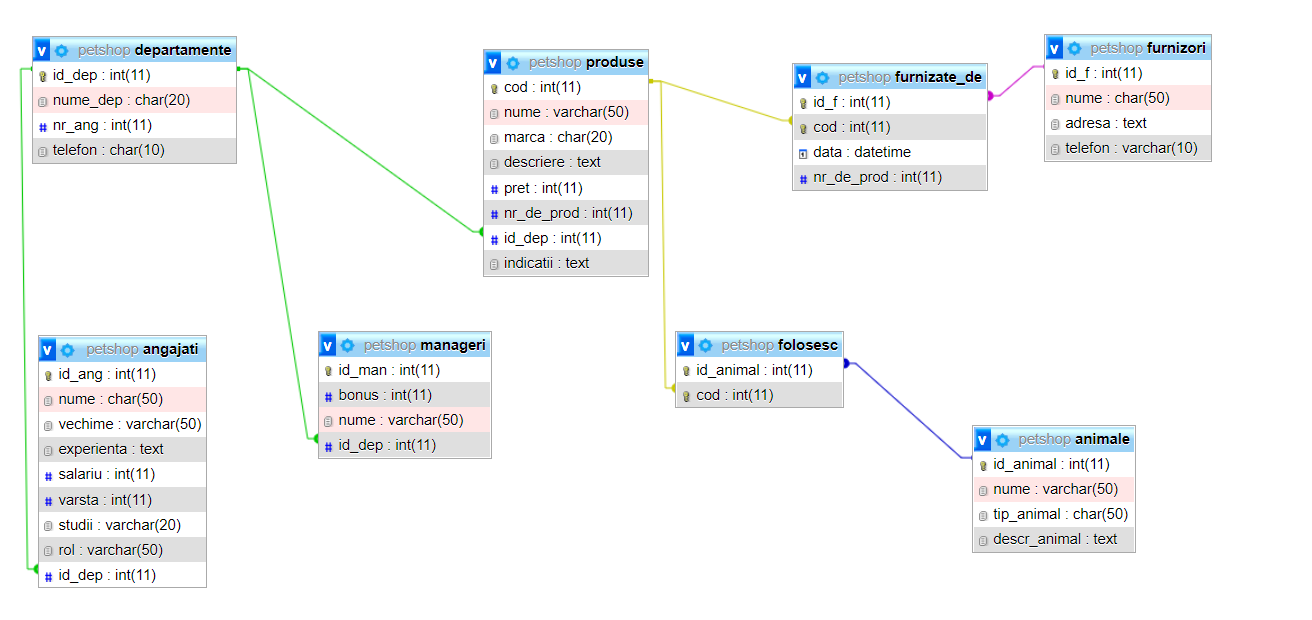
  Descriere generată automat
* FURNIZORI(id\_f, nume, adresa, telefon) O imagine care conține text

  Descriere generată automat
* Furnizate\_de(*cod*, *id\_f*, nr\_de\_prod, data) O imagine care conține text

  Descriere generată automat

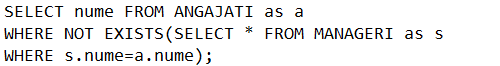
\*\*S-au subliniat cheile primare si s-au redat in format italic cheile straine

Schema relațională totală cu legăturile corespunzătoare:



4.Operații asupra bazei de date

a) Să se afișeze angajații care nu sunt manageri(diferență)



O imagine care conține text

Descriere generată automat

b) Să se afișeze numele și marca produselor cu valoare între 100 și 999lei.(Selecția)

O imagine care conține text

Descriere generată automat

c) Să se afișeze numele produselor, numărul rămas de produse în stoc, id-ul furnizorului de produs și numărul de produse distribuit de către acesta. În cazul în care produsul are mulți furnizori vor apărea detalii pentru fiecare furnizor. (Joncțiune)

O imagine care conține text

Descriere generată automatO imagine care conține text

Descriere generată automat

d)Să se verifice dacă toți managerii sunt și angajați. Se va folosi uniunea, verificându-se dacă prin unirea celor două tabele: angajați și manageri vom primi toate numele de angajați.

O imagine care conține text

Descriere generată automat

O imagine care conține text

Descriere generată automat

e)Să se afișeze tipurile de animale distincte. (PROIECȚIA)



O imagine care conține text

Descriere generată automat