Faculatea de Automatica si Calculatoare

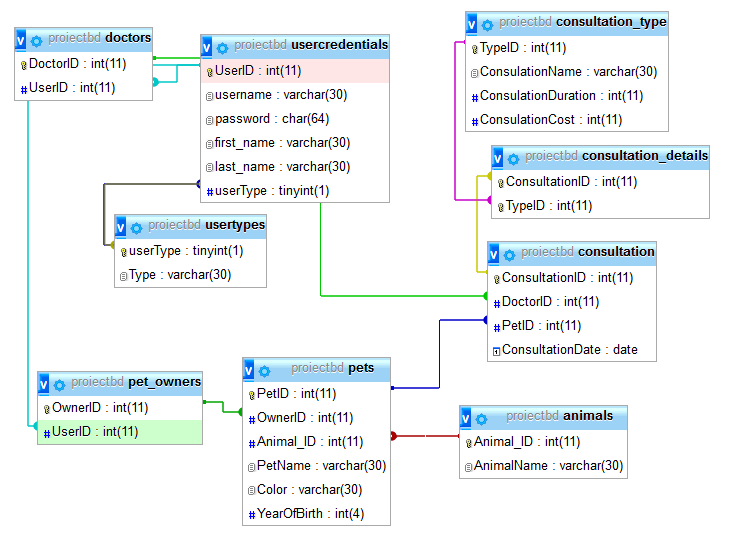
Proiect Baze de Date

Panaite Andrei – Silviu

323AC

2016

1. Diagrama structura tabele



1. Descriere

Pentru crearea acestui proiect am folosit JavaScript si jQuery pentru partea de frontend , PHP si MySql pentru partea de backend si Bootstrap pentru design-ul site-ului.

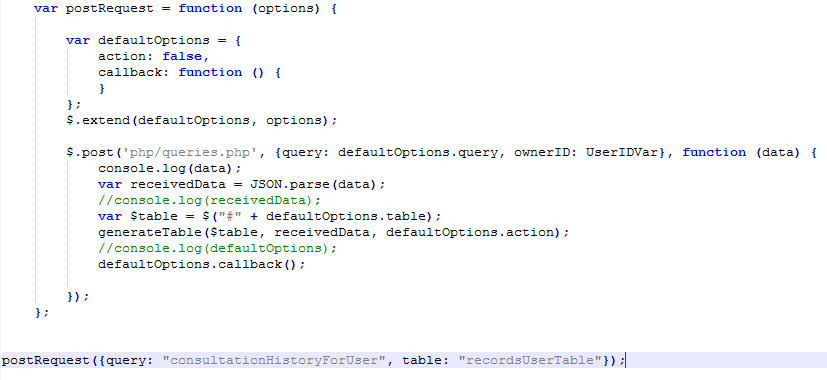
La deschiderea site-ului utilizatorul se poate loga sau isi poate crea un nou cont.

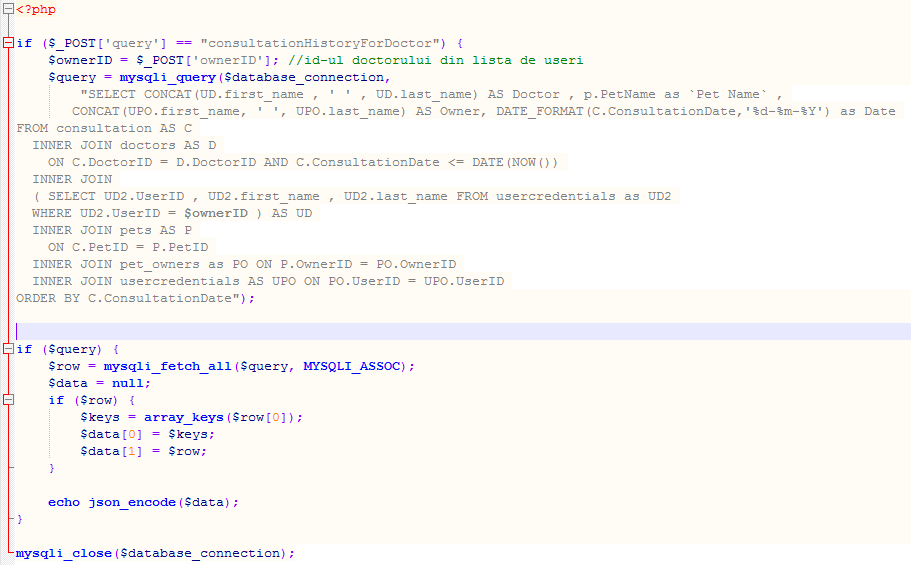
Daca logarea s-a efectuat cu success, sectiunea de login va fi inlocuita printr-un load asincron cu un panel corespunzator tipului de utilizator (userType din tablela usercredentials).

Ulterior logarii site-ul va face o serie de requesturi catre server care va raspunde cu datele corespunzatoare requestului facut. Datele sunt apoi prelucrate si prezentate in forma tabelara.

Exemplu:

Trimiterea requestului:



Rezolvarea reqestului pe server :

Pe site se pot loga 3 tipuri de utilizatori : user , doctor sau administrator.

Un utilizator de tip user isi poate vedea istoricul consultatiilor si pe cele viitoare, poate adauga, sterge sau modifica o consultatie si poate vedea informatii legate de animale sale.

Doctorul poate accesa istoricul consultatiilor , consultatiile sale viitoare si pe cele cuprinse intre doua date si poate vedea o lista cu clienti si animalele lor.

Administratorul poate modifica sau sterge un user (adaugarea unui user se face la user register) si poate accesa liste cu doctorii, consultatiile , clientii si toate animalele.

1. Selecturi folosite

Toti utilizatorii

SELECT U.username AS Username,U.first\_name AS `First Name` , U.last\_name AS `Last Name`,UT.type AS `User Type` FROM usercredentials U

INNER JOIN usertypes UT ON U.userType = UT.userType

Toti Doctorii

SELECT CONCAT(U.first\_name , ' ' , U.last\_name )AS `Name`,U.username AS `Username` ,COUNT(C.ConsultationID) AS Consultations

FROM usercredentials AS U INNER JOIN doctors AS D

ON U.UserID = D.UserID

LEFT JOIN consultation AS C

ON D.DoctorID = C.DoctorID AND c.ConsultationDate >= DATE(NOW())

GROUP BY D.DoctorID

Toate animalele

SELECT P.PetName AS `Pet Name`, CONCAT(U.first\_name ,' ', U.last\_name) AS Owner , a.AnimalName AS Animal , P.YearOfBirth AS `Birth Year` , P.Color

FROM usercredentials AS U INNER JOIN pet\_owners AS PO

ON U.UserID = PO.UserID

INNER JOIN pets AS P

ON P.OwnerID = PO.OwnerID

INNER JOIN animals AS A ON P.Animal\_ID = A.Animal\_ID

Toate animalele pentru un utilizator specificat

SELECT P.PetName as `Pet Name` ,A.AnimalName AS Animal ,P.YearOfBirth as `Birth Year`, P.Color FROM pets P

INNER JOIN pet\_owners PO

ON P.OwnerID = PO.OwnerID AND PO.UserID = $ownerID

INNER JOIN animals as A ON P.Animal\_ID = A.Animal\_ID

Toti clientii si numarul lor de animale

SELECT CONCAT(U.first\_name , ' ' , U.last\_name) AS Name ,U.username AS Username ,COUNT(p.PetID) AS Pets

FROM usercredentials AS U INNER JOIN pet\_owners AS PO

ON U.UserID = PO.UserID

INNER JOIN pets AS P

ON P.OwnerID = PO.OwnerID

GROUP BY U.UserID

Toate consultatiile

SELECT CONCAT(U.first\_name , ' ', U.last\_name) AS Doctor, P.PetName AS `Pet Name`, DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') AS `Date` FROM consultation AS C

INNER JOIN doctors AS D

ON c.DoctorID = d.DoctorID

INNER JOIN usercredentials AS U

ON D.UserID = u.UserID

INNER JOIN pets AS P

ON C.PetID = P.PetID

INNER JOIN ( SELECT CONCAT(U2.first\_name , U2.last\_name) , PO2.OwnerID AS PetOwner FROM usercredentials U2

INNER JOIN pet\_owners PO2 ON U2.UserID = PO2.UserID) AS PO

ON P.OwnerID = PO.PetOwner

ORDER BY C.ConsultationDate

Istoricul consultatiilor pentru un doctor precizat

SELECT CONCAT(UD.first\_name , ' ' , UD.last\_name) AS Doctor , p.PetName as `Pet Name` ,

CONCAT(UPO.first\_name, ' ', UPO.last\_name) AS Owner, DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') as Date

FROM consultation AS C

INNER JOIN doctors AS D

ON C.DoctorID = D.DoctorID AND C.ConsultationDate <= DATE(NOW())

INNER JOIN

( SELECT UD2.UserID , UD2.first\_name , UD2.last\_name FROM usercredentials as UD2

WHERE UD2.UserID = $ownerID ) AS UD

INNER JOIN pets AS P

ON C.PetID = P.PetID

INNER JOIN pet\_owners as PO ON P.OwnerID = PO.OwnerID

INNER JOIN usercredentials AS UPO ON PO.UserID = UPO.UserID

ORDER BY C.ConsultationDate

Consultatiile viitoare pentru un doctor precizat

SELECT CONCAT(UD.first\_name , ' ' , UD.last\_name) AS Doctor , UPO.PetName as `Pet Name` ,

CONCAT(UPO.first\_name, ' ', UPO.last\_name) AS Owner, DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') as Date

FROM consultation AS C

INNER JOIN doctors AS D

ON C.DoctorID = D.DoctorID AND C.ConsultationDate >= DATE(NOW())

INNER JOIN usercredentials as UD

ON D.UserID = UD.UserID AND UD.UserID = $ownerID

INNER JOIN ( SELECT P.PetName , P.PetID , UPO2.first\_name , UPO2.last\_name FROM pets AS P

INNER JOIN pet\_owners as PO ON P.OwnerID = PO.OwnerID

INNER JOIN usercredentials AS UPO2 ON PO.UserID = UPO2.UserID

) AS UPO

WHERE C.PetID = UPO.PetID

ORDER BY C.ConsultationDate

Consultatiile dintre doua date precizate

SELECT CONCAT(UD.first\_name , ' ' , UD.last\_name) AS Doctor , p.PetName ,

CONCAT(UPO.first\_name, ' ', UPO.last\_name) AS Owner, DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') as Date

FROM consultation AS C

INNER JOIN doctors AS D

ON C.DoctorID = D.DoctorID AND C.ConsultationDate BETWEEN CAST('" . $date1 . "' AS DATE) AND CAST('" . $date2 . "' AS DATE)

INNER JOIN ( SELECT UD2.UserID , UD2.first\_name , UD2.last\_name FROM usercredentials as UD2

WHERE UD2.UserID = $ownerID ) AS UD

INNER JOIN pets AS P

ON C.PetID = P.PetID

INNER JOIN pet\_owners as PO ON P.OwnerID = PO.OwnerID

INNER JOIN usercredentials AS UPO ON PO.UserID = UPO.UserID

ORDER BY C.ConsultationDate

Istoricul consultatiilor pentru un utilizator precizat

SELECT P.PetName , CONCAT(U.first\_name, ' ',U.last\_name) as Doctor , DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') AS Date

FROM pets as P INNER JOIN

( SELECT C1.ConsultationDate , C1.PetID , C1.DoctorID from consultation as C1

HAVING C1.ConsultationDate <= DATE(NOW()) ) as C

ON C.PetID = P.PetID

INNER JOIN doctors as D

ON C.DoctorID = D.DoctorID

INNER JOIN usercredentials as U

ON U.UserID = D.UserID

INNER JOIN pet\_owners as PO on P.OwnerID = PO.OwnerID

INNER JOIN usercredentials U2 on PO.UserID = U2.UserID AND U2.UserID = $ownerID

ORDER BY c.ConsultationDate

Consultatii viitoare pentru utilizator

SELECT C.ConsultationID as ID,P.PetName , CONCAT(U.first\_name,' ',U.last\_name) as Doctor , DATE\_FORMAT(C.ConsultationDate,'%d-%m-%Y') AS Date

FROM pets as P INNER JOIN

( SELECT C1.ConsultationDate , C1.PetID , C1.DoctorID, C1.ConsultationID from consultation as C1

HAVING C1.ConsultationDate >= DATE(NOW()) ) as C

ON C.PetID = P.PetID

INNER JOIN doctors as D

ON C.DoctorID = D.DoctorID

INNER JOIN usercredentials as U

ON U.UserID = D.UserID

INNER JOIN pet\_owners as PO on P.OwnerID = PO.OwnerID

INNER JOIN usercredentials U2 on PO.UserID = U2.UserID AND U2.UserID = $ownerID

ORDER BY C.ConsultationDate