

## Aplicatie pentru urmarirea istoricului medical intr-un cabinet veterinar

### 1. Definirea cerintelor

Eu am avut de facut o aplicatie pentru a tine evidenta istoricului medical intr-un cabinet veterinar si am ales sa se numeasca "PetVet". In etapa de proiectare a aplicatiei am identificat cateva componente pe care trebuie sa le contina aplicatia:

- Medicii ce lucreaza la acest cabinet veterinar;
- Medicamentele pe care le pune la dispozitie cabinetul veterinar;
- Pacientii ce viziteaza acest cabinet, impreuna cu proprietarii lor daca este cazul;
- Interventiile pe care le poate face un medic;
- Fisa medicala a unui pacient, ce tine evidenta unei vizite la cabinet si din ce cauza;

Pe baza acestor componente am identificat si niste functionalitati pe care trebuie sa le indeplineasca aplicatia:

- Aplicatia trebuie sa poata lasa un utilizator(medicul) sa se inregistreze in baza de date si dupa sa poata sa se logheze in aplicatie;
- Pe langa utilizatorii obisnuiti(userii) ar trebui sa existe un utilizator admin ce poate accesa paginile de administrare si insera, modifica sau sterge componente din baza de date;
- Odata logat, medicul poate accesa datele despre el, despre pacienti si despre componentele ce tin de medicamente si interventii;
- Medicul sa poata accesa informatii utile cabinetului veterinar, cum ar fi pacientii ce au fost in stare de urgenta ;
- Medicul sa poata introduce date relevante pentru aplicatie cum ar fi date personale sau date despre pacienti;

### 2. Proiectarea bazei de date

In total am folosit 8 tabele: 6 tabele ce reprezinta componentele, 1 tabel ce retine relatia dintre 2 tabele si inca un tabel ce tine cont de utilizatorul logat numit login\_tokens.

- **login\_tokens:** in acest tabel se retin niste token-uri/ cookie-uri unice si sunt active cat timp un utilizator este logat. Un utilizator logat inseamna

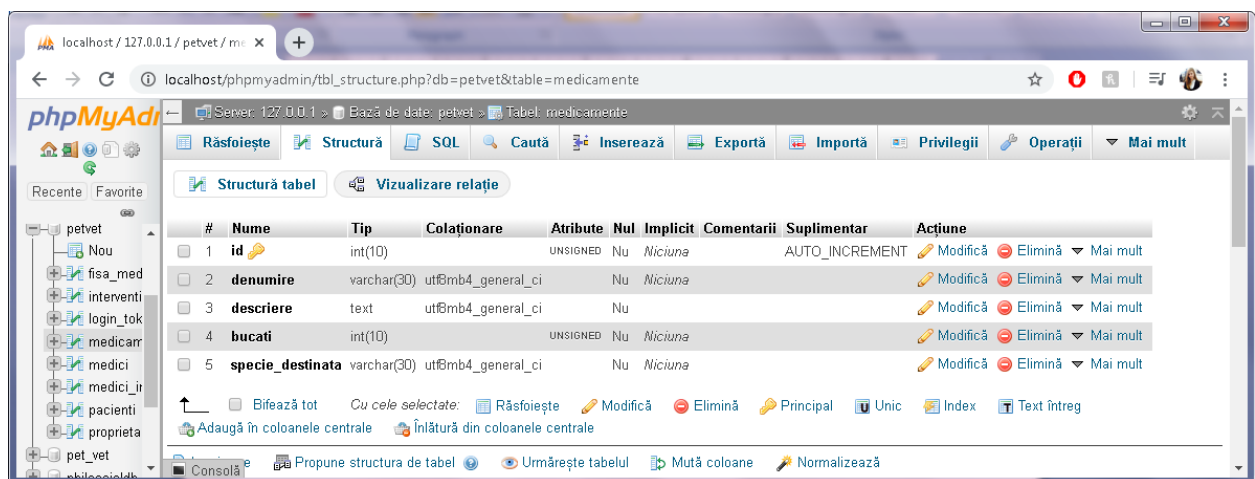
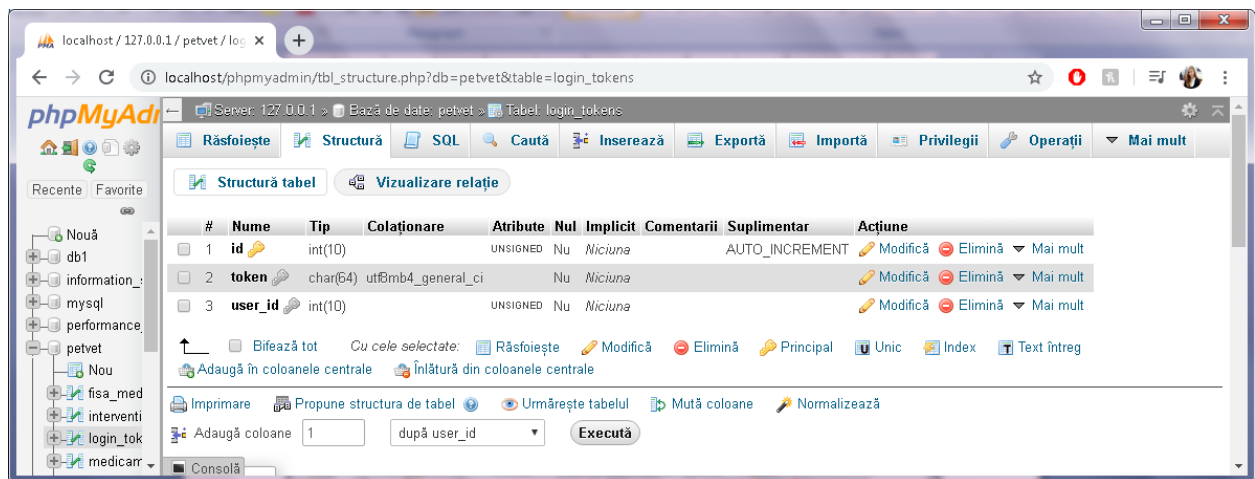
un token, dupa delogare tokenul va fi sters din tabel. Acest tabel este folosit doar pentru a tine minte ce utilizator este logat deoarece acesta poate sta logat timp de 7 zile, dupa care va fi delogat automat.

- **medici:** acest tabel retine datele personale ale medicilor inregistrati: nume, prenume, cnp, username-ul cu care se logheaza si parola si specializarea acestuia;
- **pacienti:** acest tabel retine datele despre pacienti: nume, rasa, specia si id-ul proprietarului corespunzator proprietarului lui;
- **proprietari:** acest tabel retine datele personale despre proprietari: nume, prenume, cnp;
- **medicamente:** retine datele despre fiecare medicament: denumirea, descrierea, cate bucati sunt si specia destinata;
- **interventii:** retine datele despre fiecare interventie ce poate fi efectuata de catre medici: denumire, descriere si pret;
- **medici\_interventii:** tine evidenta interventiilor si a medicilor, mai exact ne spune ce interventii stiu medicii sa realizeze;
- **fisa\_medicala:** reprezinta istoricul medical al pacientilor si retine datele: data la care s-a efectuat fisa, pacientul, ce medicament i s-a administrat, de ce interventie a avut parte, si de ce afectiune sufera

Tabelele au forma:



#	Nume	Tip	Coloanare	Atribute	Nul	Implicit	Comentarii	Suplimentar	Actiune
1	<b>id</b>	int(10)		UNSIGNED	Nu	Niciuna		AUTO_INCREMENT	Modifică Elimină Mai mult
2	<b>afectiune</b>	varchar(100)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
3	<b>data</b>	date			Nu	Niciuna			Modifică Elimină Mai mult
4	<b>id_pacient</b>	int(10)		UNSIGNED	Nu	Niciuna			Modifică Elimină Mai mult
5	<b>id_medicament</b>	int(10)		UNSIGNED	Nu	Niciuna			Modifică Elimină Mai mult
6	<b>id_interventie</b>	int(10)		UNSIGNED	Nu	Niciuna			Modifică Elimină Mai mult



localhost / 127.0.0.1 / petvet / m...  
localhost/phpmyadmin/tbl\_structure.php?db=petvet&table=medici

Server: 127.0.0.1 > Bază de date: petvet > Tabel: medici

Răsfășește Structură SQL Caută Inserează Exportă Importă Privilegii Operații Mai mult

Structură tabel Vizualizare relație

#	Nume	Tip	Colaționare	Atribute	Nul	Implicit	Comentarii	Suplimentar	Acțiune
1	id	int(10)		UNSIGNED	Nu	Niciuna		AUTO_INCREMENT	Modifică Elimină Mai mult
2	nume	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
3	prenume	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
4	cnp	varchar(13)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
5	username	varchar(32)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
6	parola	varchar(64)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
7	specializare	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult

Bifează tot Cu cele selectate: Răsfășește Modifică Elimină Principal Unic Index Text întreg

localhost / 127.0.0.1 / petvet / m...  
localhost/phpmyadmin/tbl\_structure.php?db=petvet&table=medici\_interventii

Server: 127.0.0.1 > Bază de date: petvet > Tabel: medici\_interventii

Răsfășește Structură SQL Caută Inserează Exportă Importă Privilegii Operații Mai mult

Structură tabel Vizualizare relație

#	Nume	Tip	Colaționare	Atribute	Nul	Implicit	Comentarii	Suplimentar	Acțiune
1	id	int(10)		UNSIGNED	Nu	Niciuna		AUTO_INCREMENT	Modifică Elimină Mai mult
2	id_medic	int(10)		UNSIGNED	Nu	Niciuna			Modifică Elimină Mai mult
3	id_interventie	int(10)		UNSIGNED	Nu	Niciuna			Modifică Elimină Mai mult

Bifează tot Cu cele selectate: Răsfășește Modifică Elimină Principal Unic Index Text întreg

Aadaugă în coloanele centrale Înălătură din coloanele centrale

Imprimare Propune structura de tabel Urmărește tabelul Mută coloane Normalizează

Aadaugă coloane 1 după id\_interventie Execută

localhost / 127.0.0.1 / petvet / p...  
localhost/phpmyadmin/tbl\_structure.php?db=petvet&table=pacienti

Server: 127.0.0.1 > Bază de date: petvet > Tabel: pacienti

Răsfășește Structură SQL Caută Inserează Exportă Importă Privilegii Operații Mai mult

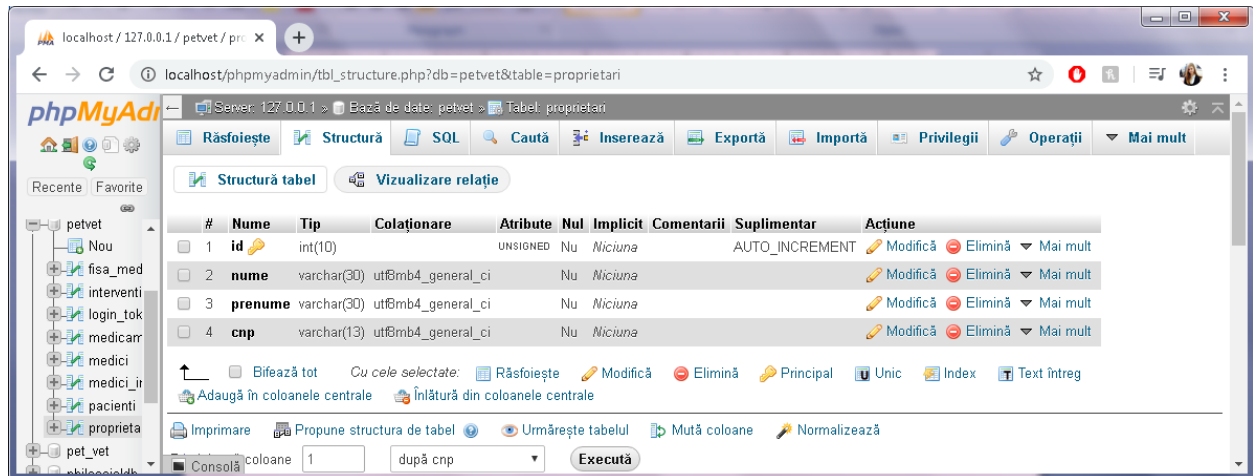
Structură tabel Vizualizare relație

#	Nume	Tip	Colaționare	Atribute	Nul	Implicit	Comentarii	Suplimentar	Acțiune
1	id	int(10)		UNSIGNED	Nu	Niciuna		AUTO_INCREMENT	Modifică Elimină Mai mult
2	nume	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
3	rasa	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
4	specie	varchar(30)	utf8mb4_general_ci		Nu	Niciuna			Modifică Elimină Mai mult
5	id_proprietar	int(10)		UNSIGNED	Da	NULL			Modifică Elimină Mai mult

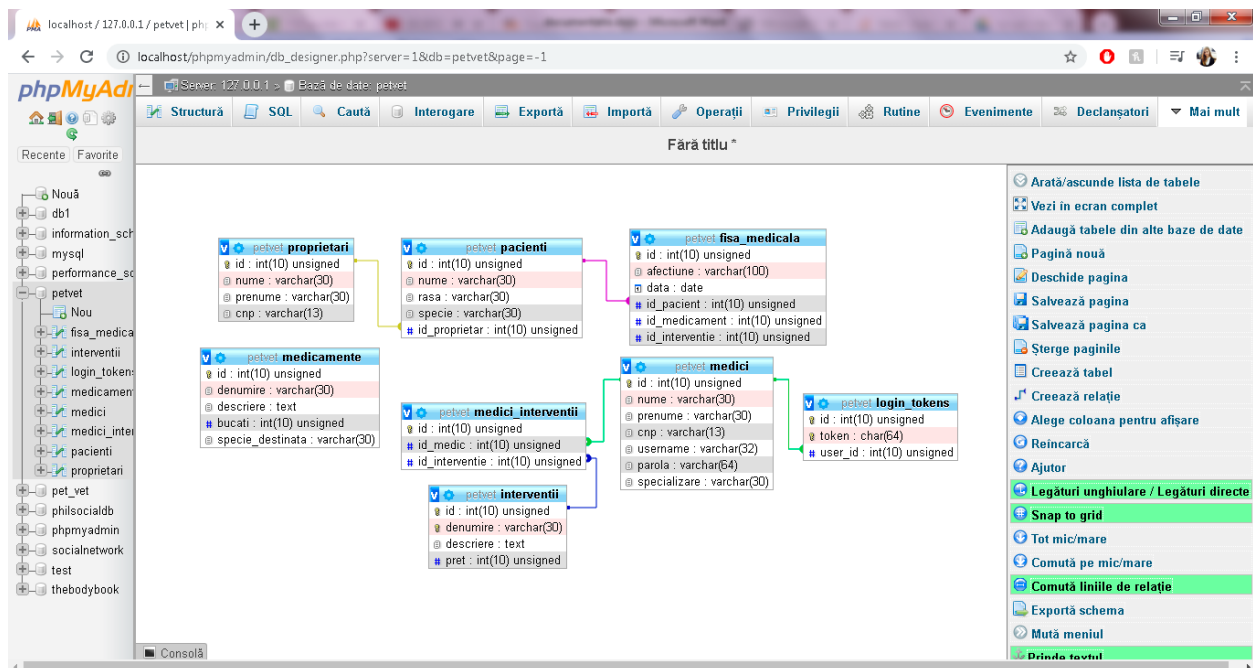
Bifează tot Cu cele selectate: Răsfășește Modifică Elimină Principal Unic Index Text întreg

Aadaugă în coloanele centrale Înălătură din coloanele centrale

Propune structura de tabel Urmărește tabelul Mută coloane Normalizează



Si au urmatoarele relatii:



**One-to-Many:** Proprietar(cheia primara: id) – Pacient(cheia straina: id\_proprietar)

-un pacient(animal de companie) are un singur proprietar, dar un proprietar poate sa detina mai multe animale de companie;

Pacient(cheia primara: id) – Fisa Medicala(cheia straina: id\_pacient)

-fisa medicala este specifica unui singur macient, dar un pacient poate avea mai multe fise medicale deoarece poate vizita de mai multe ori cabinetul medical;

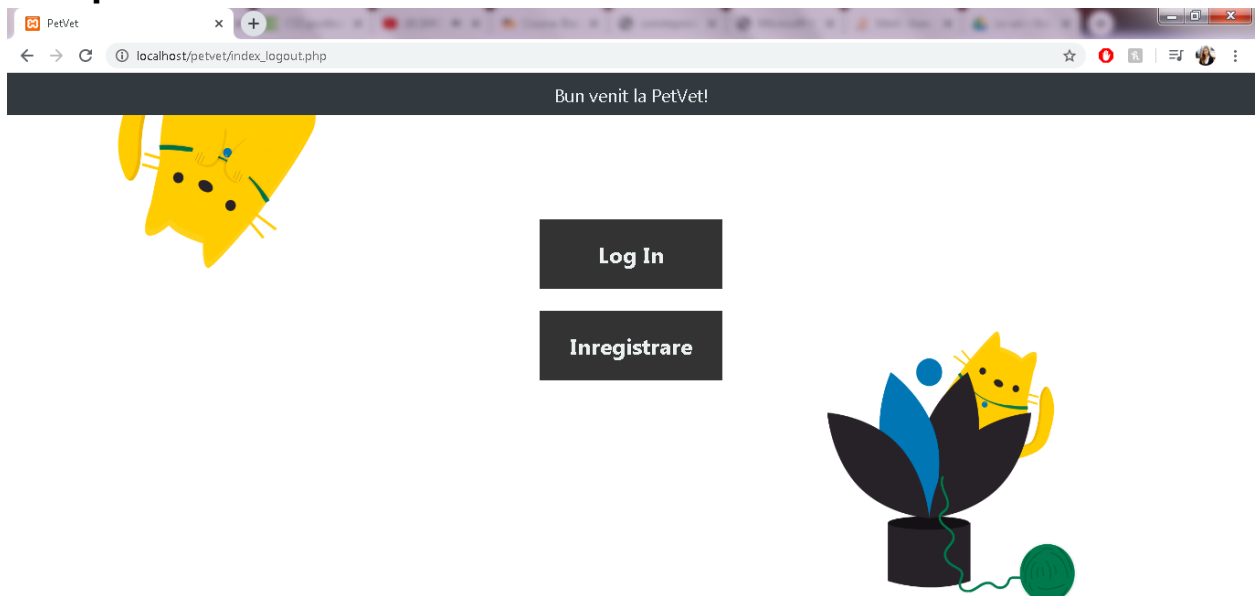
**Many-to-Many:** Medici – Interventii (cheie stăină compusă: id\_medic, id\_interventie)

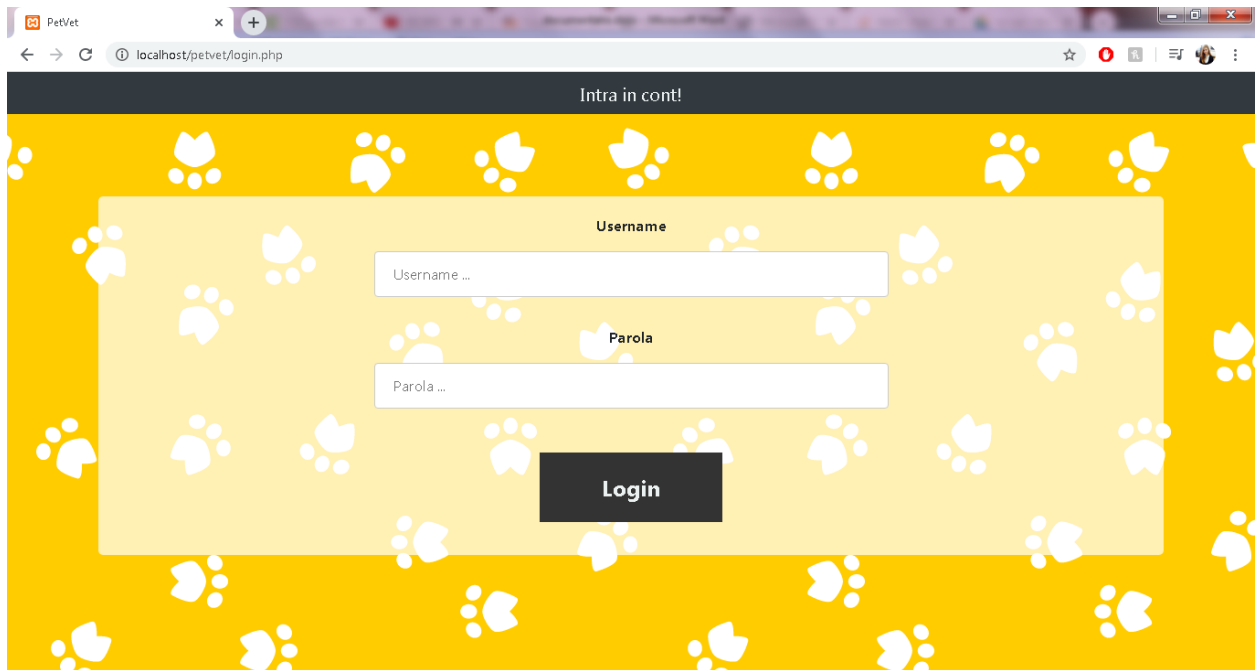
-un medic poate efectua mai multe interventii, dar aceleași interventii pot fi efectuate de mai mulți medici;

### 3. Proiectarea interfeței vizuale și implementarea acesteia într-un mediu de programare

Am ales ca mediu de programare să folosesc php + html + css (pentru interfață). Am 3 fișiere css pe care le folosesc în mai multe documente php, iar unde este nevoie de lucruri în plus cum ar fi: butoane diferite atunci în header între tagurile `<style></style>` introduc componentele css necesare. Legat de partea de programare folosesc în același timp php și html, exceptând baza.php unde fac conexiunea cu baza de date și Login.php unde verific tokenul de logare și returnez userul logat în aplicație. În aceste două fișiere din urmă am doar cod php. Pentru restul fișierelor, chiar dacă pare mai încurcat codul pentru că este php combinat cu html, am găsit o ordine a scrierii codului astfel încât să aibă sens.

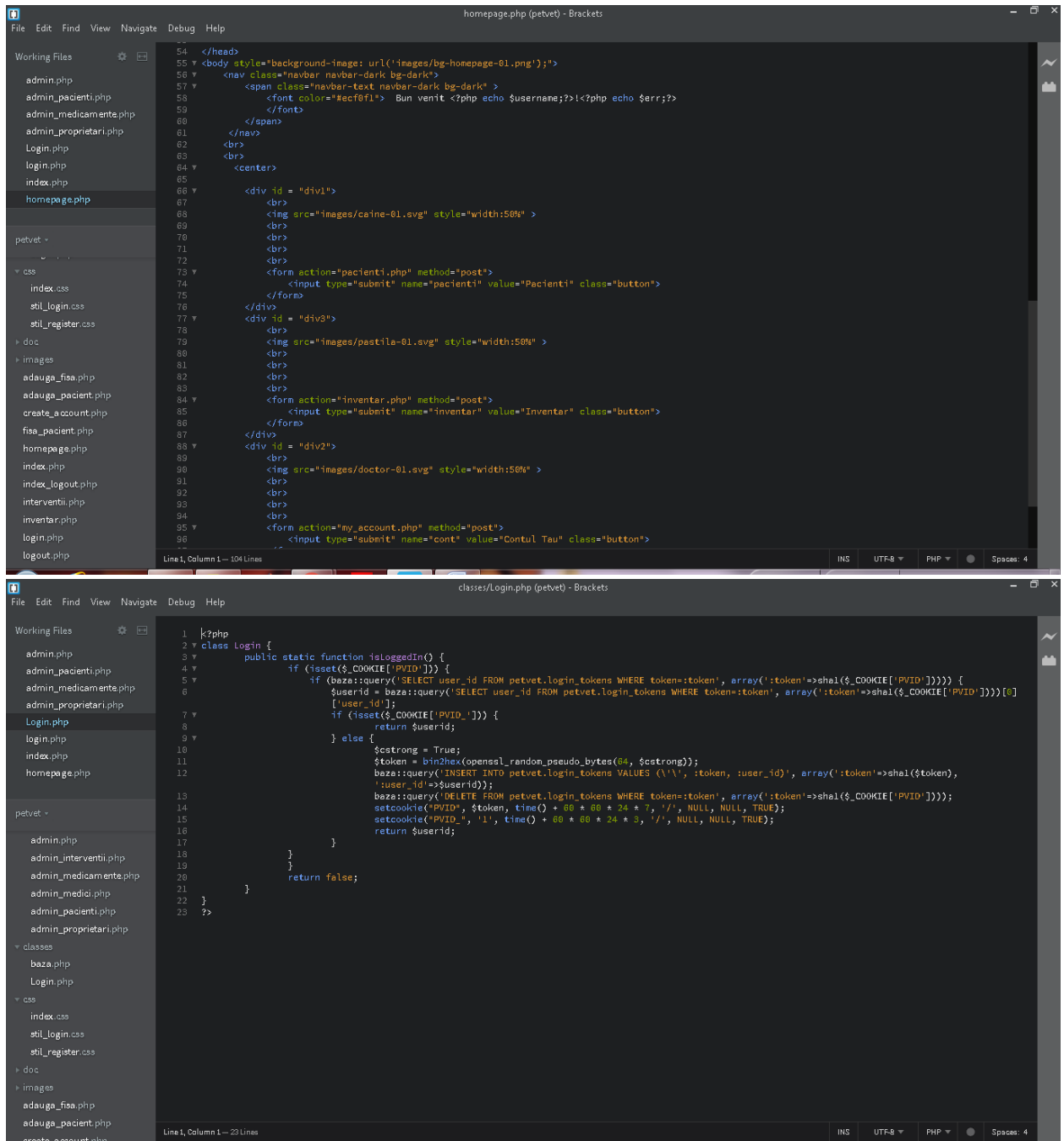
#### Exemple din interfață:





## Exemple din cod:

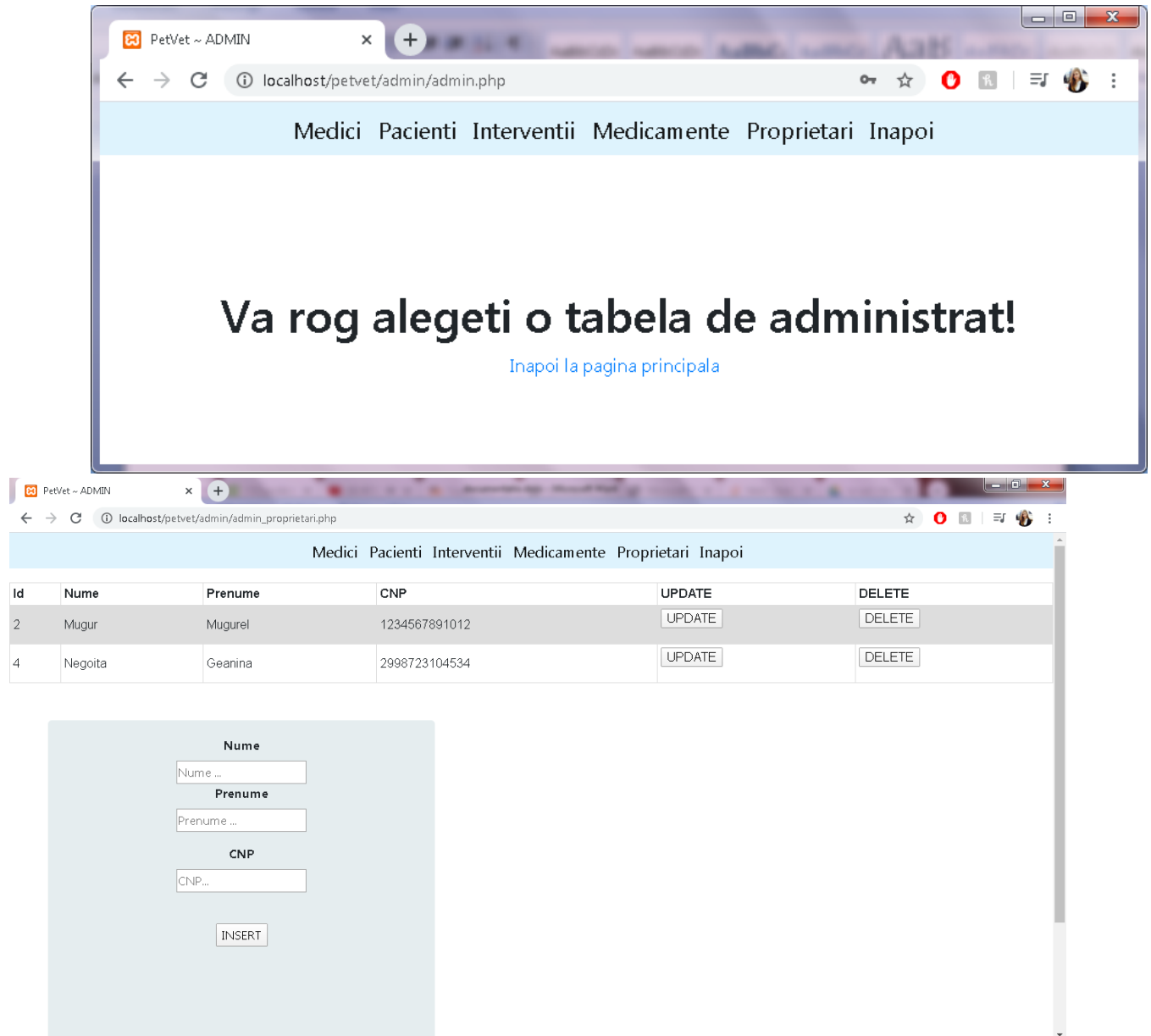
A screenshot of a code editor showing the source code for the login page. The editor has a dark theme. On the left, there is a file explorer showing a list of files including 'admin.php', 'admin\_pacient.php', 'admin\_medicamente.php', 'admin\_proprietari.php', 'login.php', 'index.php', and 'homepage.php'. The main editor area shows PHP code for the 'login.php' file. The code includes a session check, a database query to verify the user's login details, and a redirect to 'index\_logout.php' if the login is successful. Below the PHP code, there is a CSS block for styling the login form, including dimensions, margins, and background colors for different elements.



#### 4. Corelarea interfetei vizuale cu baza de date

In mare parte pentru insert, update, delete se ocupa administratorul(username: admin2020 parola: parola2020) care face aceste operatii pentru 5 tabele:





Dar si medicul poate face insert, update, delete pentru datele proprii sau sa adauge pacienti si fise medicale sau sa adauge in tabelul de legatura cu interventiile:

The first screenshot shows a user account page titled 'Bun venit andreea!'. It features a yellow background with white paw prints. On the left, there are input fields for 'Nume:' (filled with 'Negoita'), 'Prenume:' (filled with 'Andreea'), 'Username:' (filled with 'andreea'), 'CNP:' (filled with '111111111111'), 'Specializare:' (filled with 'andreea'), and 'Parola:'. On the right, there are three dark buttons: 'Sterge Contul', 'Log Out', and 'Inapoi'.

The second screenshot shows a page titled 'Adauga o fisa medicala!'. It has a blue background with white paw prints. It contains several input fields: 'Nume Pacient', 'CNP Proprietar', 'Medicament administrat', 'Afectiune', 'Interventie', and 'Data fisei'.

In componenta aplicatiei am folosit multe interogari, dar le voi mentiona pe cele mai importante:

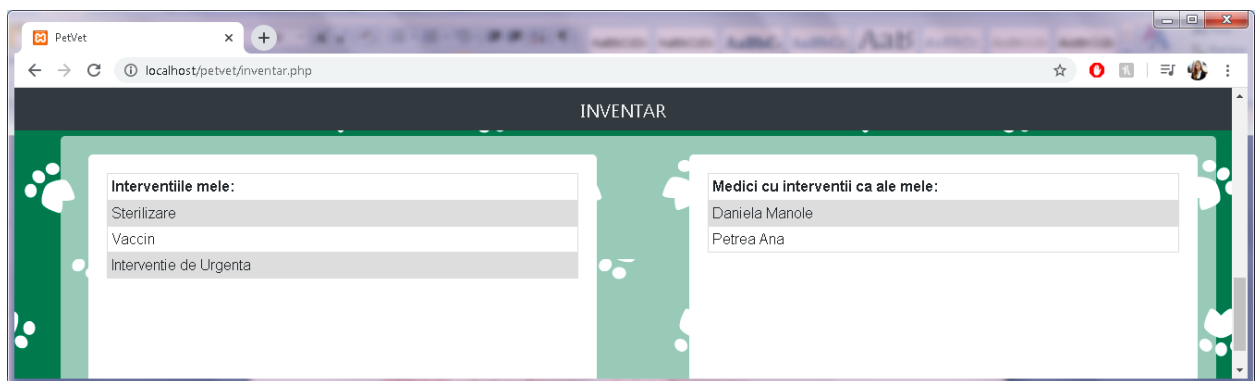
Interogari simple:

1. Afisez pe ecran toate interventiile utilizatorului curent logat  
Se gaseste in fisierul inventar.php la linia 194 iar pe ecran apare atunci cand utilizatorul intra in pagina inventarului.

```
SELECT * FROM petvet.interventii INNER JOIN petvet.medici_interventii ON  
petvet.interventii.id = petvet.medici_interventii.id_interventie WHERE  
petvet.medici_interventii.id_medic = :idmedic ;
```

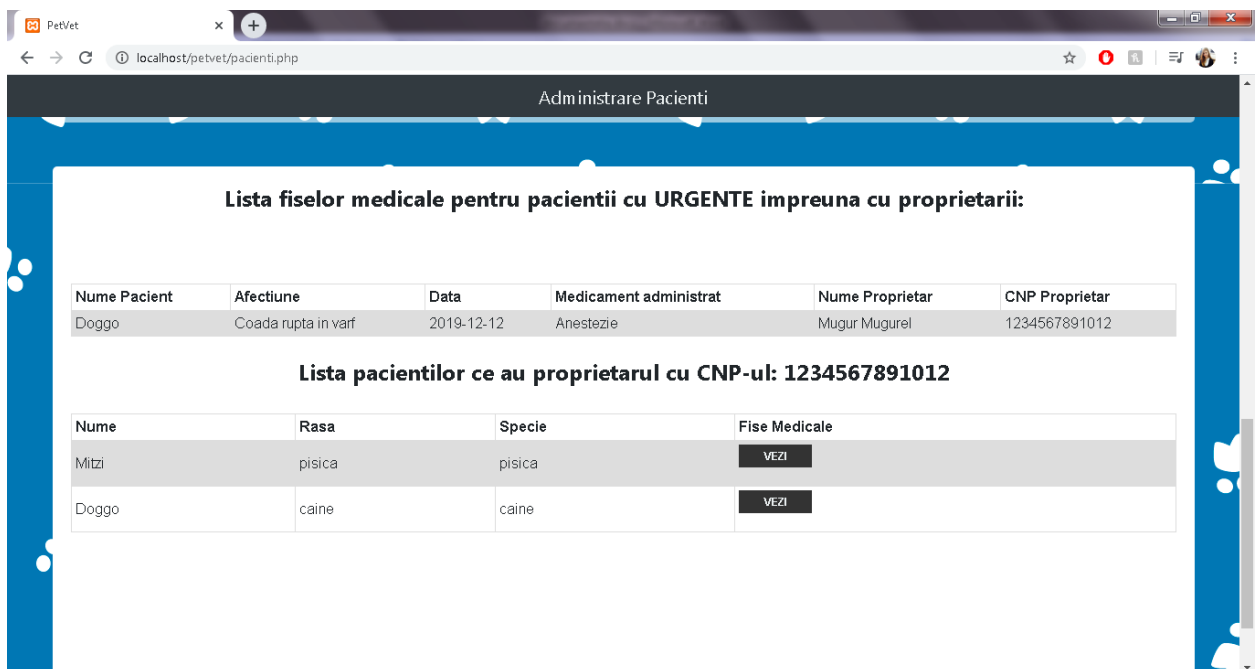
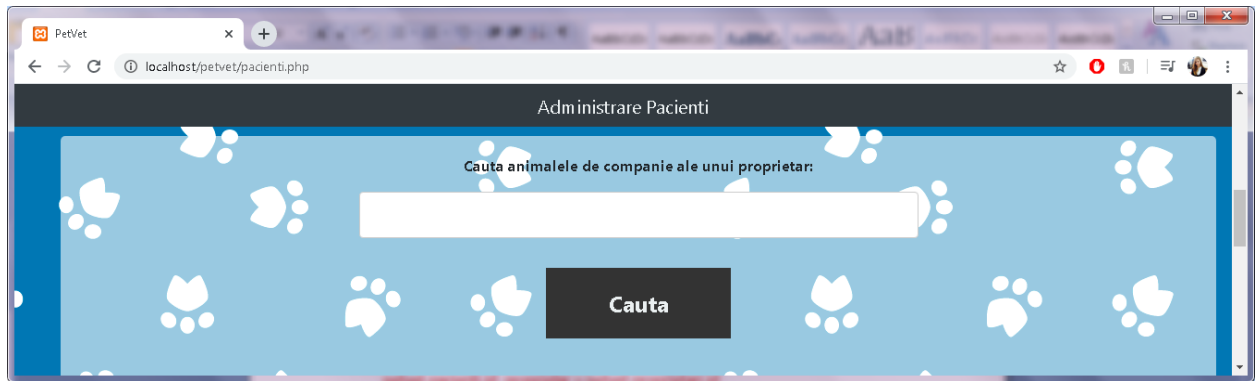
2. Afisez pe ecran medicii care au interventiile ca ale mele  
Se gaseste in fisierul inventar.php la linia 211, iar pe ecran apare atunci cand  
utilizatorul intra in pagina inventarului.

```
SELECT DISTINCT petvet.medici.numa, petvet.medici.prenume FROM  
petvet.medici INNER JOIN petvet.medici_interventii ON petvet.medici.id =  
petvet.medici_interventii.id_medic WHERE petvet.medici.id != :idmedic AND  
petvet.medici_interventii.id_interventie IN (SELECT petvet.interventii.id FROM  
petvet.interventii INNER JOIN petvet.medici_interventii iON petvet.interventii.id  
=petvet.i.id_interventie WHERE petvet.i.id_medic =:idmedic)
```



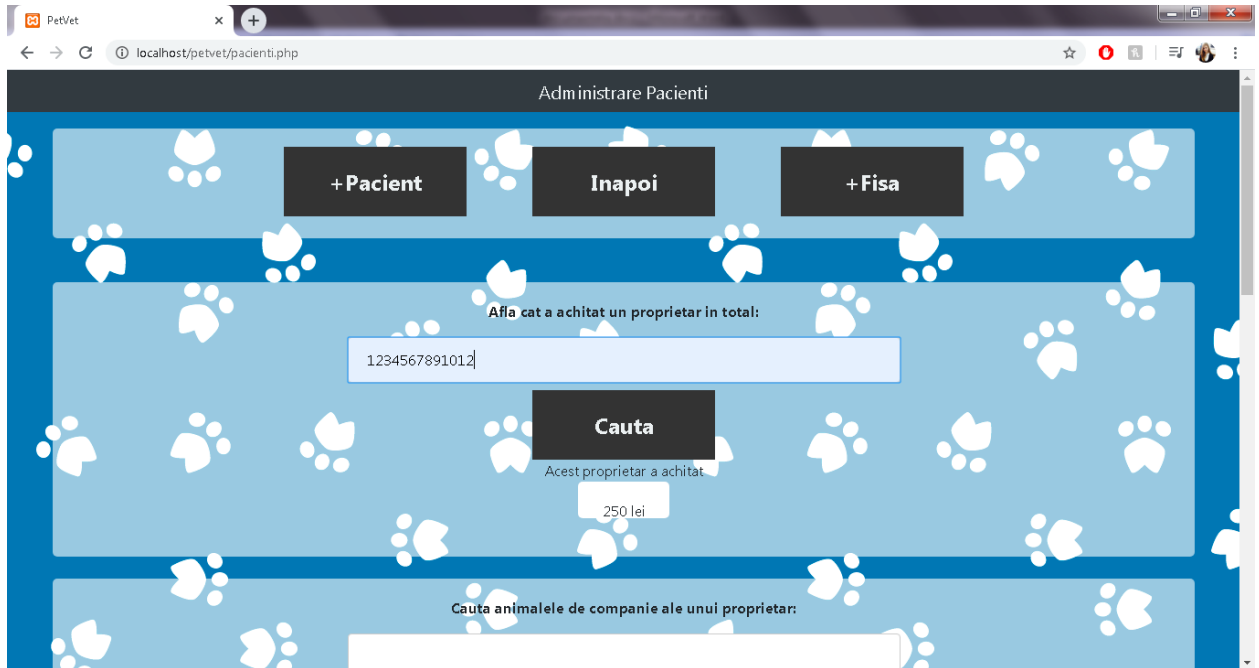
3. In functie de CNP-ul unui proprietar, pot sa afisez ce animale detine  
Se gaseste in fisierul pacienti.php la linia 270 si apare pe ecran formularul pentru  
introducerea cnp-ului proprietarului in pagina despre pacienti.

```
SELECT petvet.pacienti.id, petvet.pacienti.numa, petvet.pacienti.rasa,  
petvet.pacienti.specie FROM petvet.pacienti INNER JOIN petvet.proprietari ON  
petvet.pacienti.id_proprietar = petvet.proprietari.id WHERE petvet.proprietari.id  
= :idproprietar
```



4. Cati bani a achitat un proprietar de-a lungul timpului platind interventiile  
Se gaseste in fisierul pacienti.php la linia 27, iar formularul pentru introducerea proprietarului se gaseste in pagina despre pacienti.

```
SELECT SUM(petvet.interventii.pret) FROM petvet.interventii INNER JOIN
petvet.fisa_medicala ON petvet.interventii.id =
petvet.fisa_medicala.id_interventie
INNER JOIN petvet.pacienti ON petvet.fisa_medicala.id_pacient =
petvet.pacienti.id
INNER JOIN petvet.proprietari ON petvet.pacienti.id_proprietar =
petvet.proprietari.id
WHERE petvet.proprietari.id = :idproprietar
```

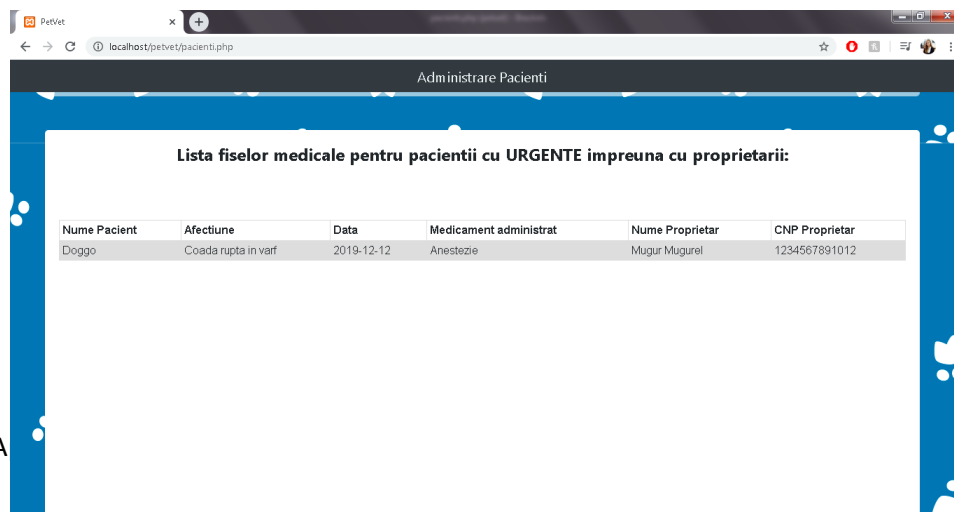


5. Ce pacienti au fost in stare de urgenta, sa se afiseze fisele lor medicale  
Se gaseste in fisierul pacienti.php la linia 215, iar datele rezultate se afla in pagina despre pacienti.

```
SELECT * FROM petvet.fisa_medicala INNER JOIN petvet.interventii ON  
petvet.fisa_medicala.id_interventie = petvet.interventii.id WHERE  
petvet.interventii.denumire=:denumire
```

6. Afisarea proprietarilor pacientilor cu urgente  
Se gaseste in fisierul pacienti.php la linia 236, iar datele rezultate se afla in pagina despre pacienti.

```
SELECT petvet.proprietari.ume FROM petvet.proprietari INNER JOIN  
petvet.pacienti ON petvet.proprietari.id = petvet.pacienti.id_proprietar WHERE  
petvet.pacienti.id=:id
```



### Interogari complexe

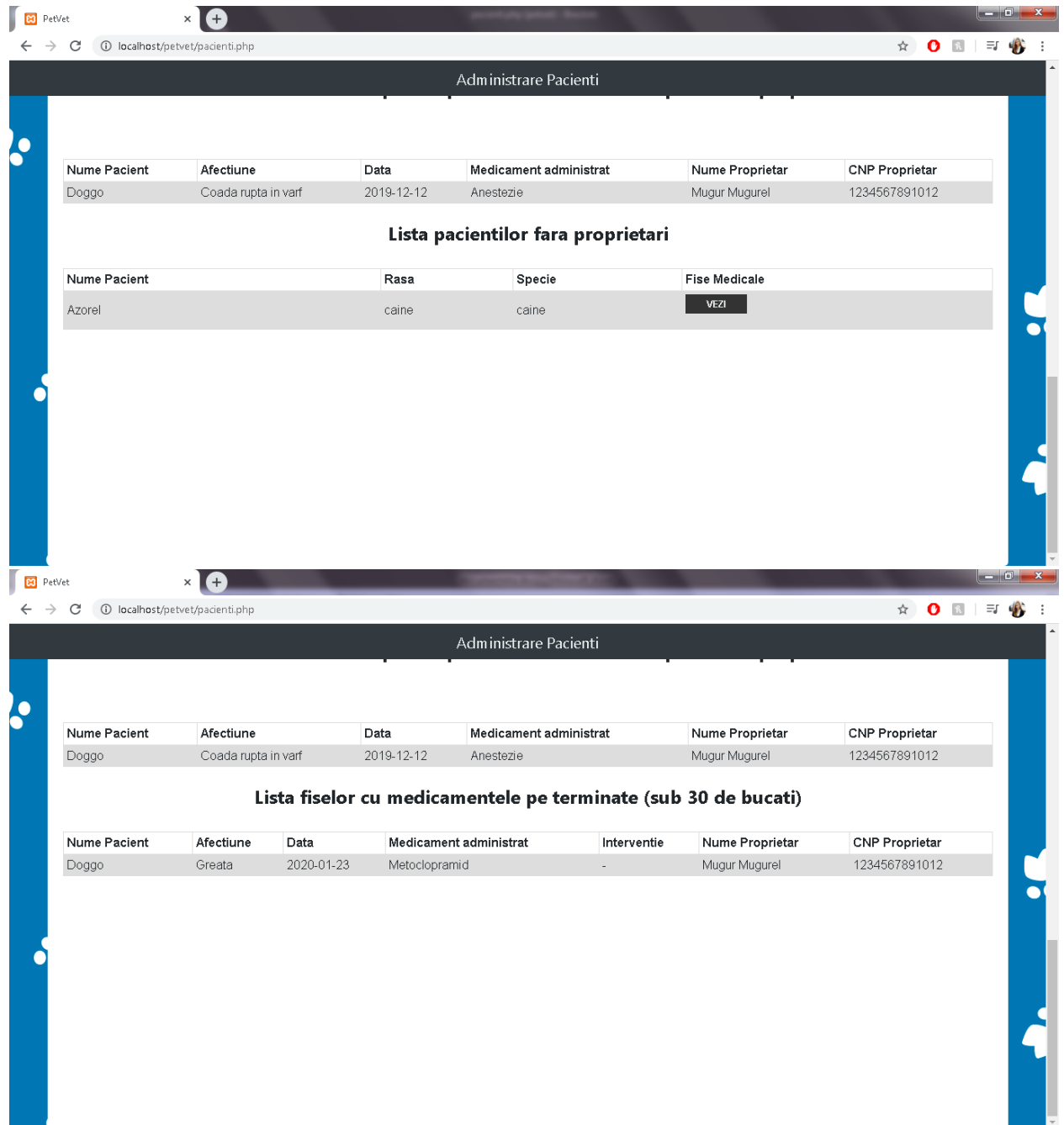
1. Afisarea fiselor medicale unde pastilele sunt pe terminate(sub 30 de bucati)  
Se gaseste in fisierul pacienti.php la linia 304. In urma apasarii butonului asociat din pagina despre pacienti, mai jos se va afisa lista fiselor medicale rezultate interogarii.

```
SELECT * FROM petvet.fisa_medicala  
WHERE 30 > (SELECT bucati FROM petvet.medicamente WHERE  
petvet.medicamente.id = petvet.fisa_medicala.id_medicament)
```

2. Afisarea pacientilor fara proprietar  
Se gaseste in fisierul pacienti.php la linia 357. In urma apasarii butonului asociat din pagina despre pacienti, mai jos se va afisa lista fiselor medicale rezultate interogarii.

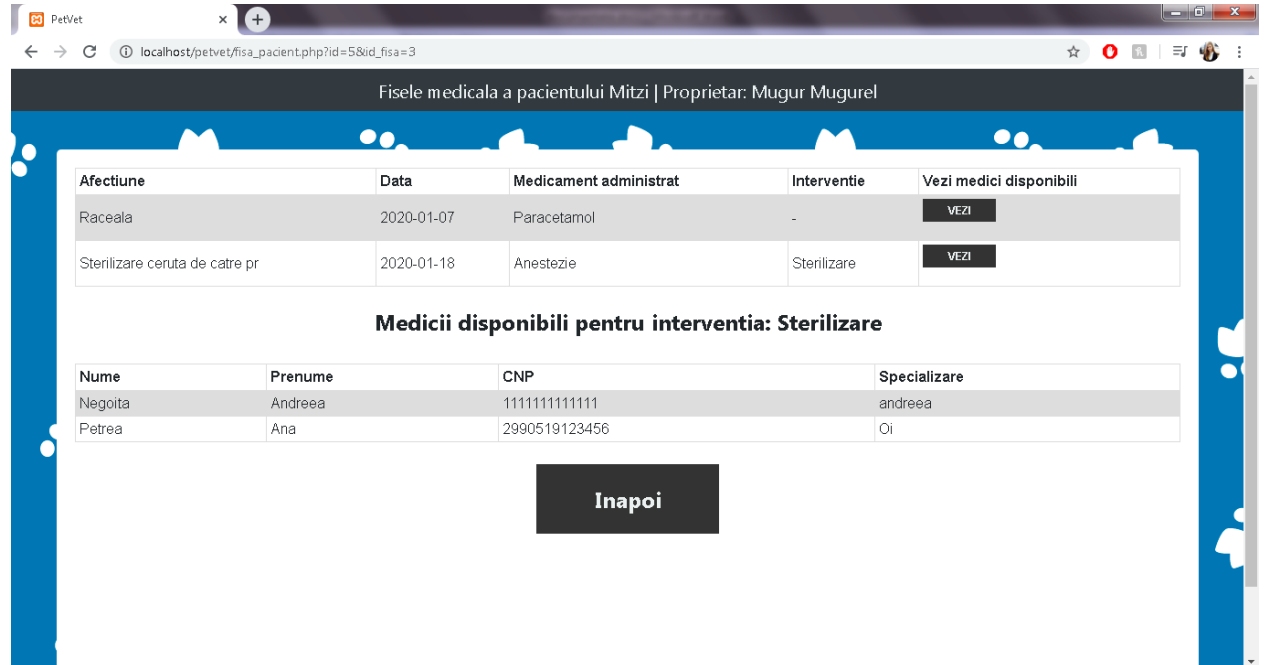
```
SELECT * FROM petvet.pacienti WHERE id NOT IN (SELECT petvet.pacienti.id  
FROM petvet.pacienti INNER JOIN petvet.proprietari ON  
petvet.pacienti.id_proprietar = petvet.proprietari.id
```





3. Afisarea medicilor responsabili de a face interventia dintr-o fisa medicala  
Se gaseste in pagina fisa\_pacient.php la linia 193. Pentru a afisa medicii responsabili de o interventie de procedeaza astfel: in pagina pacienti se cauta animalele de companie ale unui proprietar. Pentru fiecare animal putem intra in pagina despre fisele lui medicale. In dreptul unei fise medicale avem un buton care ne arata ce medici se ocupa de interventia respectiva fisei acestui pacient.

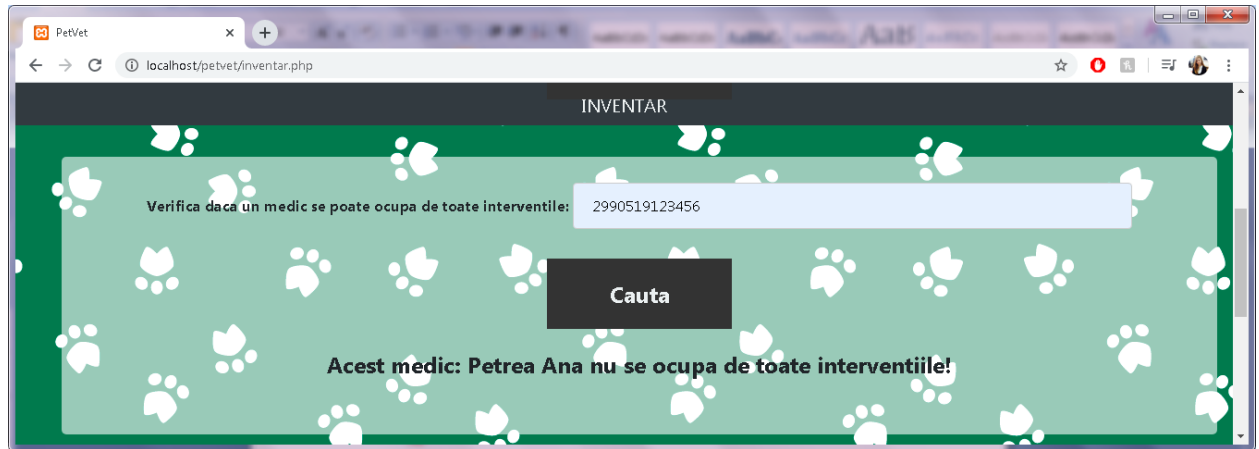
```
SELECT * FROM petvet.medici WHERE id IN (SELECT id_medic FROM  
petvet.medici_interventii WHERE id_interventie = (SELECT id_interventie  
FROM petvet.fisa_medicala WHERE id = :id))
```



4. In functie de CNP-ul unui medic putem afisa daca acesta este responsabil de toate interventiile sau nu
- Se gaseste in fisierul `inventar.php` la linia 169, iar pe ecran se gaseste in pagina inventarului. Se introduce cnp-ul medicului iar butonul de cautare ne arata daca medicul respectiv se ocupa de toate interventiile sau nu.

```
SELECT * FROM petvet.medici WHERE (SELECT COUNT(*) FROM  
petvet.interventii) = (SELECT COUNT(petvet.medici_interventii.id_interventie)  
FROM petvet.medici_interventii WHERE petvet.medici_interventii.id_medic =  
:id)
```





Fisierele responsabile de partea de administrare se gasesc in folderul admin, iar fisierele responsabile de conectarea la baza de date si partea de login se afla in folderul classes. In folderul css se gasesc documentele css importate in cele php.

Automat aplicatia deschide pagina index.php, daca utilizatorul este deja logat suntem redirectionati in homepage.php, iar daca nu este logat ajungem in pagina index\_nelogat.php. In paginile de administratie se ajunge doar daca ne logam cu datele: username = admin2020 si parola = parola2020. In celelalte pagini se ajunge dupa logare prin homepage: pacienti.php, inventar.php si my\_account.php . De aici in functie de preferinte se trece in restul de pagini. In fiecare pagina se verifica daca utilizatorul este logat, pentru ca in cazul unei erori sa afiseze paginile pentru persoane nelogate.

Daca doriti sa intrati in conturile utilizatorilor deja inregistrati de mine puteti scrie atat la username cat si la parola valoarea username-ului din tabele. Acest lucru nu este o regula, doar ca am ales sa introduc astfel datele pentru a nu tine minte parolele. Daca doriti sa va inregistrati ca medic puteti sa puneti orice parola doriti, acesta va fi criptata pentru a fi mai sigura.