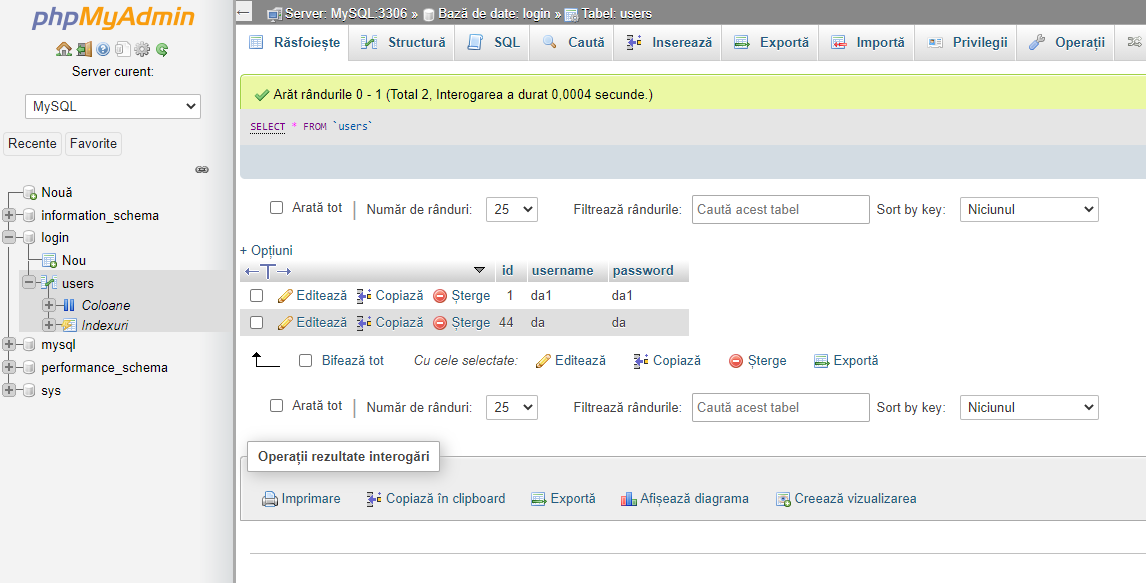
Proiect la disciplina Dezvoltarea aplicațiilor Web

Miricică Robert Vl ăduț

Grupa 40313 C

21/01/2020

Pasul 1.Am creat baza de date “login” si tabelul “users”din aceasta baza de date cu campurile “id”,”username” si “password”,folosind programul WAMP server 3.2.3 in phpMyAdmin.Campul “id” este cel principal,tipul sau este int(11) si are auto\_increment.Campul “username” este de tipul varchar(100) iar in acest camp se vor stoca username-urile iar campul “password” este de tipul varchar(20) iar in acest camp se va stoca parola fiecarui username.



Pasul 2.Am creat pagina de start “index.php” cu ajutorul limbajului html.In aceasta pagina sunt facute cele 2 input-uri username si password unde trebuie sa le introduci pentru a te putea loga,iar in cazul in care nu ai cont,poti apasa pe butonul “register” pentru a fi redirectionat catre pagina “register.php”.Acolo iti poti face cont pentru a reveni catre “index.php” pentru a te loga.Aceasta pagina are inclus “server.php” pentru a se putea face conexiunea cu baza de date “login” si tabelul “users” unde sunt verificare campurile username si password.



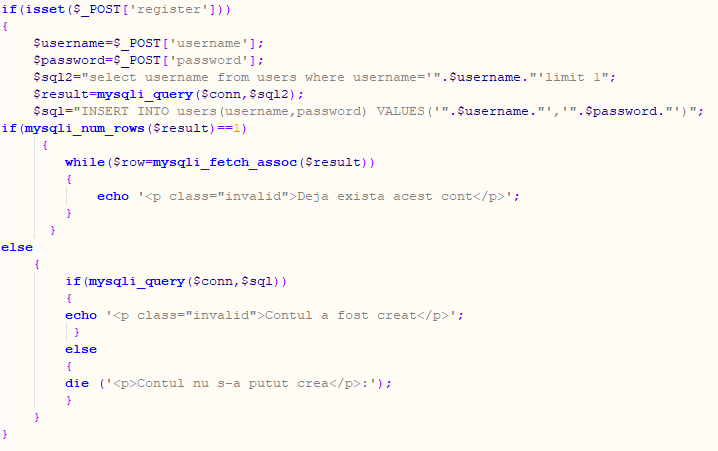
Pasul 3.Am creat conexiunea intre pagina de start “index.php” si baza de date “login”,mai exact sistemul de logare a utilizatorului in pagina “server.php”.In aceasta pagina vom gasi conexiunea facuta cu $conn=mysqli\_connect($servername,$username,$password,$dbname); unde fiecare variala este declarata mai sus.De exemplu $servername="localhost"; $username="root"; $password=""; $dbname="login";

Mai departe gasim if(isset($\_POST['login'])) care verifica daca variabila este declarata sau goala in urma $\_POST care se ocupa sa colecteze datele din form-ul facut in html cu cele 2 input-uri.Mai departe se face query-ul $sql="select \* from users where username='".$username."' AND password='".$password."' limit 1"; unde se selecteaza din tabela user username-ul si password-ul si este verificat prin if(mysqli\_num\_rows($result)>0) si while($row=mysqli\_fetch\_assoc($result)) daca numarul de randuri este mai mare decat 0 si cat timp randul respectiv este egal cu rezultatul array-ul adus de functia mysqli\_fetch\_assoc().In cazul in care este adevarat se va face logarea,sessiunea va incepe si va memora $username fiind redirectionat catre pagina “pagina.php”,iar in cazul in care este fals o sa tipareasca echo '<p class="invalid">Cont invalid </p>';

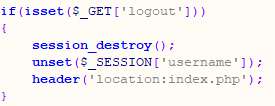


Mai jos gasim if(isset($\_POST['register'])) care tot la fel verifica daca variabila este declarata sau goala in urma $\_POST care se ocupa sa colecteze datele din form-ul facut in html cu cele 2 input-uri. Mai departe se face query-ul $sql2="select username from users where username='".$username."'limit 1"; care selecteaza username-ul din tabela users pentru a putea verifica daca exista deja un cont cu un username similar sau nu.Dupa primul query $sql2 se face alt query $sql="INSERT INTO users (username,password) VALUES ('".$username."','".$password."')"; pentru a se introduce in tabelul users cele 2 input-uri,username si password iar in cazul in care conexiunea s-a facut o sa se tipareasca echo '<p class="invalid">Contul a fost creat</p>';,inca caz contrar

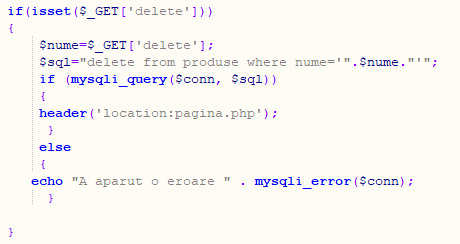
die ('<p>Contul nu s-a putut crea</p>:');



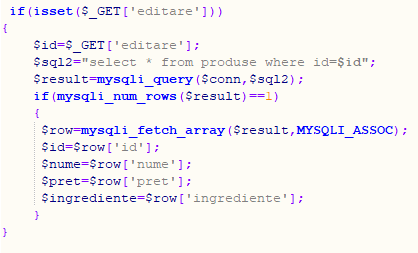
In continuare gasim if(isset($\_GET['logout'])) care colecteaza datele in urma $\_GET.Acesta ajuta sa distrugem sesiunea creata odata ce ne logam in “pagina.php” pentru a nu putea accesa “pagina.php” si “editare.php” fara a fi logat.Odata ce sesiunea este distrusa ne redirectioneaza catre pagina de start “index.php”.



Mai departe avem if(isset($\_GET['delete'])) care care colecteaza datele in urma $\_GET.Aceasta ajuta la stergerea datele din baza de date “login” din tabela “produse” prin query-ul $sql="delete from produse where nume='".$nume."'";



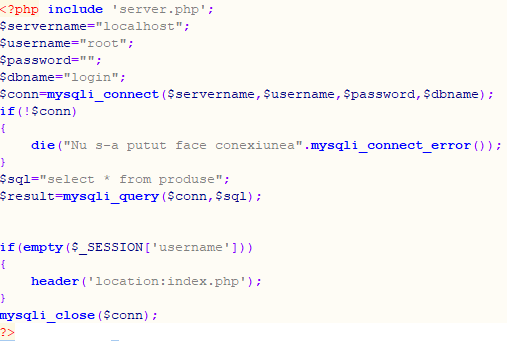
In final avem if(isset($\_GET['editare'])) care colecteaza datele in urma $\_GET.Acesta ajuta la editarea datelor din baza de date ducand mai departe campurile $id,$nume,$pret si $ingrediente la pagina “editare.php”.Selecteaza prin query-ul $sql2="select \* from produse where id=$id"; campul id din tabelul produse.



Pasul 4.Am creat pagina “register.php” cu ajutorul limbajului html,aceasta pagina este aproape similara cu “index.php”,ea este facuta doar cu html si css si are inclus “server.php” unde este facuta conexiunea intre baza de date si cele 2 input-uri din form. <form method="Post"> acest form are metoda post care specifica cum o sa fie trimise datele,el le trimite de forma http.



Pasul 5.Am creat pagina “pagina.php” care este pagina unde este utilizatorul redirectionat dupa ce se logheaza sau isi creaza un cont.Acolo poate sa vada produsele care sunt salvate in baza de date “login”,din tabelul “produse”.Poate adauga un nou produs,poate modifica produsele care sunt deja salvate sau le poate elimina.”pagina”.php este facuta in html,css si in php.

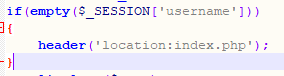


$sql="select \* from produse"; $result=mysqli\_query($conn,$sql); sunt folosite pentru a prelua datele din baza de date si a le afisa pe “pagina.php”.Dupa cum se poate vedea in urmatoarea imagine am folosit in acest div:<div class="container"> <?php if(mysqli\_num\_rows($result)>0):

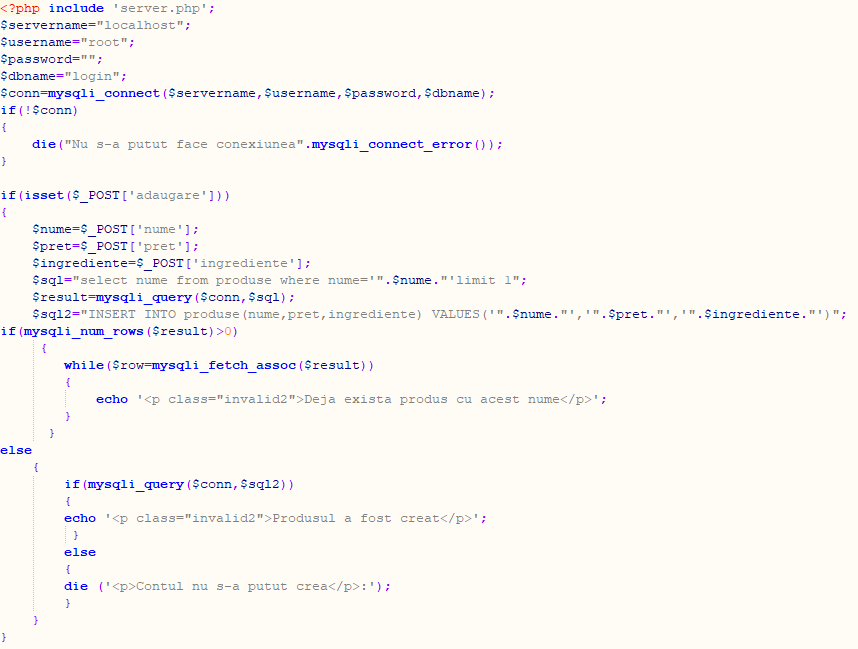
while($row=mysqli\_fetch\_assoc($result)): ?> php verifica daca produsele exista in baza de date,daca ele exista o sa fie tiparite in site in acest div:<div class="unu">

<p> <?php echo $row['nume'].'-'.$row['pret'].' lei';?></p>.

In aceasta pagina mai putem gasi si if(empty($\_SESSION['username'])) care verifica daca sesiunea este pornita si este corecta,in caz contrar ne va redirectiona catre pagina “index.php”.



Pasul 6.Am creat ultima pagina “editare.php” care este pagina dupa ce utilizatorul foloseste butonul “Editeaza reteta” sau “Adauga reteta”.Daca utilizatorul a folosit butonul “Adauga reteta” odata ce este redirectionat o sa aiba campurile nume,pret si ingrediente goale.El le poate completa si prin folosirea butonului “Adauga” o sa inregistreze produsul in baza de date “login” tabelul ”produse”.

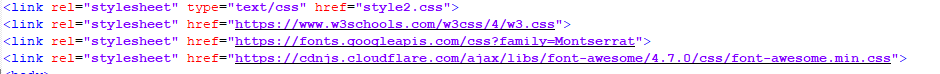


In cazul in care utilizatorul foloseste butonul ”Editeaza reteta” din pagina “pagina.php” este redirectionat cu campurile nume,pret si ingrediente completate din baza de date pentru a putea vedea ce valori au produsele pe care doreste sa le modifice.

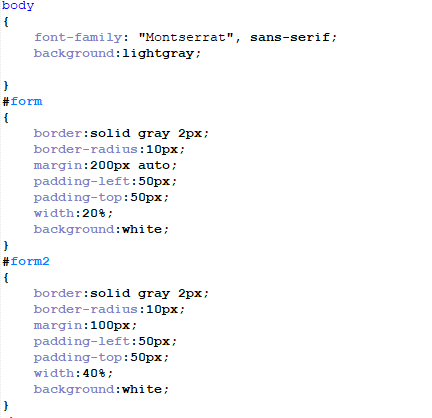




Pasul 7.Am creat foaia “style2.css” care ajuta la stilizarea  elementelor html.Toate paginile enumerate mai sus folosesc aceasta foaie.



De exemplu dupa cum se poate vedea in imaginea de mai jos pentru elemental body setam fontul sa fie “Montserrat”,”sans-serif” si background-ul sa fie de culoarea lightgray.Pentru elementul div cu id-ul “form” ii setam border-ul sa fie de tip “solid”,culoarea sa fie “gray” si grosimea 2px.De asemenea putem sa setam si pozitia acestui element,dimensiunile,culoarea si mai multe.

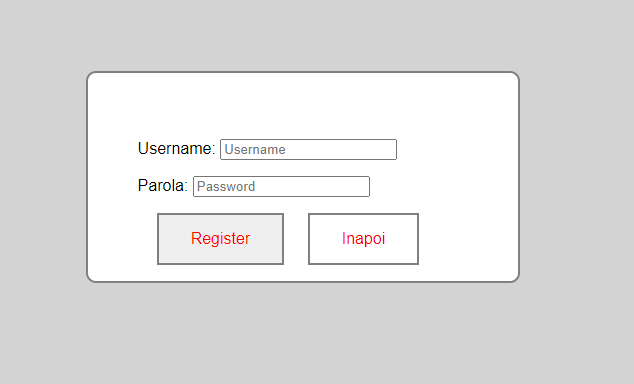


Site-ul arata astfel:

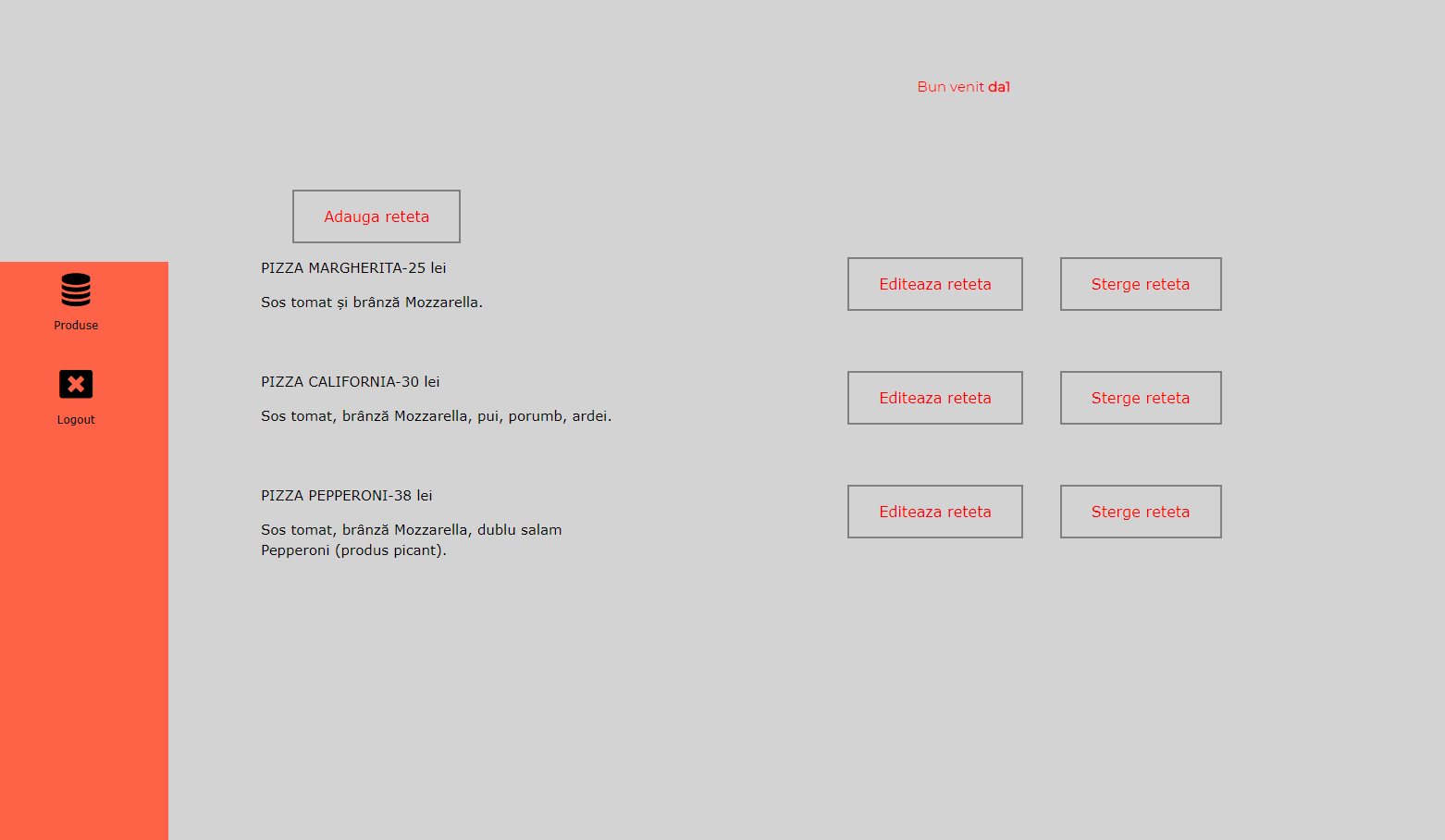
1.Index.php



2.Register.php



3.Pagina.php



4.Editare.php

