**VISOKA TEHNIČKA ŠKOLA**

**STRUKOVNIH STUDIJA**

**S** **U B O T I C A**

**PROJEKAT**

**iz predmeta Web programiranje**

***PORUČIVANJE HRANE***

**KANDIDAT MENTOR**

**Vladimir Vorkapić dr Zlatko Čović**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SUBOTICA 2023.god.**

**SADRŽAJ**

[OPIS ZADATKA 3](#_Toc32766)

[REALIZACIJA ZADATKA 5](#_Toc28862)

[OPIS FUNKCIONALNOSTI 13](#_Toc358)

[KORIŠĆENA LITERATURA 20](#_Toc30177)

**OPIS ZADATKA**

Cilj projekta je da se kreira web aplikacija-sajt za jednostavno poručivanje hrane iz jednog

restorana. Aplikacija razlikuje tri nivoa pristupa: gost, registrovan/prijavljen korisnik i

administrator.

**Gost**

• može da pogleda ponuđene informacije na sajtu

• može da izvrši registraciju na sajtu

**Registrovan/prijavljen korisnik**

• može da naruči hranu iz restorana

• može da promeni svoje profilne podatke (lozinka, ime, prezime, broj telefona, adresu)

• može da pregleda svoje narudžbine

• može da postavi i obriše jelo iz liste omiljenih jela (brži odabir jela)

• može da zahteva promenu lozinke (zaboravljena lozinka)

**Administrator**

• može da unosi, menja i briše kategorije hrane

• može da unosi, menja i briše informacije o jelima (fotografija, opis, cena, kategorija)

• može da pregleda sve narudžbine

• može da na osnovu odabira datum pregleda narudžbine za taj datum. Sve narudžbine ostaju u bazi podataka.

Registraciju korisnika mora biti urađena na bezbedan način i obaveznim slanjem aktivacionog

linka putem e-maila. Slanje linka koristiti i kod zahteva za promenu lozinke. Ne sme se dozvoliti

registracija korisničkih naloga sa istom e-mail adresom. E-mail adresa mora biti jedinstvena i

ona predstavlja korisničko ime.

Prijavljeni korisnik vrši pregled hrane na osnovu kategorija. Svako jelo treba da ima informaciju

o nazivu, ceni, veličini porcije, kategoriji i da poseduje fotografiju. Korisnik ima mogućnost da

naruči više vrsta jela u okviru jedne narudžbine i to u količinama od 1-30. Ukoliko mu je neko

jelo omiljeno može da ga doda u listu omiljenih jela. Iz liste omiljenih jela korisnik može brže

da naruči jelo. Pored svakog jela u listi omiljenih treba da stoji dugme za uklanjanje iz liste

omiljenih jela. Pre potvrđivanja narudžine korisnik mora da unese adresu dostave koja može da

se razlikuje od registrovane adrese. Nakon uspešne narudžbine korisnik treba da dobije e-mail

o uspešnom naručivanju.

Potrebno je kreirati opciju Pretraga preko koje prijavljeni korisnik na osnovu ključnih reči-

pojmova vrši pretragu u ponudi hrane.

Administratorski deo zaštiti upotrebom sesija (PHP). Sve korisničke lozinke „hešovati“ bcrypt

algoritmom.

Kreirati bazu podataka pod imenom food i unutar nje tabele koje će zadovoljiti sve

funkcionalnosti projekta.

# REALIZACIJA ZADATKA

**Prvo što sam uradio je povezao tabelu korišćenjem PDO-a:**

const PARAMS = [

"HOST" => 'localhost',

"USER" => 'root',

"PASS" => '',

"DBNAME" => 'it',

"CHARSET" => 'utf8mb4'

];

const SITE = 'http://localhost/flopi\_shop/'; // enter your path on localhost

$dsn = "mysql:host=" . PARAMS['HOST'] . ";dbname=" . PARAMS['DBNAME'] . ";charset=" . PARAMS['CHARSET'];

$pdoOptions = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

PDO::ATTR\_EMULATE\_PREPARES => false

];

try {

$pdo = new PDO($dsn, PARAMS['USER'], PARAMS['PASS'], $pdoOptions);

} catch (\PDOException $e) {

var\_dump($e->getCode());

throw new \PDOException($e->getMessage());

}

return $pdo;

**U slucaju da je korisnik izabrao login proveravaju se uneti podaci:**

$username = trim($\_POST["username"]);

$password = trim($\_POST["password"]);

if (!empty($username) and !empty($password)) {

$data = checkUserLogin($pdo, $username, $password);

if ($data and is\_int($data['id\_user'])) {

$\_SESSION['username'] = $username;

$\_SESSION['id\_user'] = $data['id\_user'];

redirection('index.php');

} else {

redirection('login.php?l=1');

}

} else {

redirection('login.php?l=1');

}

**Prilikom registrovanja , sledeci kod sprečava korisnika da unese slabu lozinku**

if (!preg\_match("#.\*^(?=.{8,20})(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[0-9])(?=.\*\W).\*$#", $password)) {

redirection('login.php?l=10');

}

**Ako ne postoji korisnik sa datim emailom on ce ga registrovati:**

if (!existsUser($pdo, $email)) {

$token = createToken(20);

if ($token) {

$id\_user = registerUser($pdo, $password, $firstname, $lastname, $email, $token,$ukupnaCena,$adresa);

try {

$body = "Your username is $email. To activate your account click on the <a href=" . SITE . "active.php?token=$token>link</a>";

sendEmail($pdo, $email, $emailMessages['register'], $body, $id\_user);

redirection("login.php?r=3");

} catch (Exception $e) {

error\_log("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

error\_log($e->getMessage());

error\_log("file:" . $e->getFile() . " line:" . $e->getLine());

redirection("login.php?r=11");

}

}

} else {

redirection('login.php?r=2');

}

function existsUser(PDO $pdo, string $email): bool

{

$sql = "SELECT id\_user FROM users WHERE email=:email AND (registration\_expires>now() OR active ='1') LIMIT 0,1";

$stmt = $pdo->prepare($sql);

$stmt->bindParam(':email', $email, PDO::PARAM\_STR);

$stmt->execute();

$stmt->fetch(PDO::FETCH\_ASSOC);

if ($stmt->rowCount() > 0) {

return true;

} else {

return false;

}

}

**Kada smo se ulogovali prvo sto nam program radi je to da proverava da li smo se ulogovali :**

if (isset($\_SESSION['username']) OR isset($\_SESSION['id\_user']))

{

}

**Ako korisnik nije ulogovan on može samo da lista , a ne i da poručuje !**

**U product delu search radimo pomocu sledeceg koda :**

if (isset($\_GET['search']) && !empty($\_GET['search'])) {

$search = $connect->real\_escape\_string($\_GET['search']);

$search\_arr = explode(' ', $search);

foreach ($search\_arr as $key => $value) {

if(mb\_strlen($value) <= 2) {

unset($search\_arr[$key]);

}

}

if (isset($search\_arr) && !empty($search\_arr)) {

$where = array();

foreach ($search\_arr as $key => $value) {

$where[] = "WHERE proizvodjac LIKE UPPER('%$value%') OR model LIKE UPPER('%$value%')";

}

$sql\_str .= implode(' OR ', $where);

}

}

**Slicno tako je I unos podatak u bazu!**

if (isset($\_GET['broj']) && !empty($\_GET['broj'])) {

$searchDva = $connect->real\_escape\_string($\_GET['broj']);

$search\_arrDva = explode(' ', $searchDva);

if (isset($search\_arrDva) && !empty($search\_arrDva)) {

$whereDva = array();

foreach ($search\_arrDva as $keyDva => $valueDva) {

$whereDva = $valueDva;

}

$vimac=$\_SESSION['id\_user'];

$query= "INSERT INTO `narudzbine`(`id\_user`, `email`,`narudzbina`, `date\_time`) VALUES ( '$vimac', '{$\_SESSION['username']}', '$whereDva','$today')";

$resultnova = $connect->query($query);

if($resultnova){

$last\_id = $connect->insert\_id;

}

}

}

**Kada odemo na products možemo primetiti srce za favorit!**

function myFunction(x) {

x.classList.toggle("fas");

}

**Možemo proveriti da li je korisnik vec uneo hranu u favorit:**

$sql = "SELECT \* from omiljeno WHERE id\_vozila = '" . $id . "' AND id\_user = '". $\_SESSION['id\_user']. "'" ;

if ($result = mysqli\_query($connect, $sql)) {

// Return the number of rows in result set

$rowcount = mysqli\_num\_rows( $result );

if ($rowcount>=1) {

echo ' <i onclick="myFunction(this)" class="far fa-heart fas bottom-left"></i>';

}

else

{

echo ' <i onclick="myFunction(this)" class="far fa-heart bottom-left"></i>';

}

}

**Sada prelazimo na korpu , ukoliko je prazna ispisace se da je prazna:**

<?php

if(empty($products))

{

?>

<h1>Prazna korpa</h1>

<h1>Kupi Nešto :)</h1>

<a href="products.php" class="cta"><button>Buy</button></a>

<?php

}

else{

?>

<h1>Vaša korpa</h1>

<?php

}

?>

**Ako korpa nije prazna redom ce ispisati sve elente u korpi**

**Ako porucimo hranu prvo ce nam izaci prozor sa pitanjem da potvrdimo adresu . Kada se adresa potvrdi preko ajaxa unosimo podatke u bought tabelu I brisemo iz karte sve elemente I refreshujemo stranicu**

var defaultName = '<?php echo $\_SESSION["adress"]; ?>';

var userInput = prompt('Potvrdi svoju adresu:',defaultName);

if (userInput !== null) {

var inputElement = document.getElementById("output");

var value = inputElement.innerHTML;

console.log(value);

var total = document.getElementsByClassName("total-number")[0];

var totalValue = total.innerHTML;

console.log(totalValue);

console.log(userInput);

$.ajax({

url: "insert.php",

type: "POST",

data: { value: value , userInput:userInput, totalValue:totalValue },

success: function(response) {

console.log(response);

// Handle success response

},

error: function(xhr, status, error) {

console.log(xhr.responseText);

// Handle error response

}

});

$.ajax({

url: 'refresh.php',

method: 'GET',

success: function() {

// Reload the page

location.reload();

}

});

}

// Get the values from the input fields

});

});

**Ako zelimo da promenimo svoje podatke ulazimo u acount.php**

**Tu uzimamo sve podatke iz usersa I stavljamo ih u sesije**

$sql = "SELECT firstname , lastname , phoneNum, adress from users where id\_user = {$\_SESSION['id\_user']}";

if ($result = $connect -> query($sql)) {

while ($row = $result -> fetch\_assoc()) {

$\_SESSION['firstname']=$row['firstname'];

$\_SESSION['lastname']=$row['lastname'];

$\_SESSION['phoneNum']=$row['phoneNum'];

$\_SESSION['adress']=$row['adress'];

//sad uraditi da bude lokacija a ne dirstname

}

}

**Kada promenimo podatkle kliknemo da insert I on nam update-a podatke**

if(isset($\_POST['submitBtn']) )

{

$fname=$\_POST['fname'];

$lname=$\_POST['lname'];

$email=$\_POST['email'];

$position=$\_POST['position'];

if(!empty($fname) && !empty($lname) && !empty($email) && !empty($position))

{

$query= "UPDATE users SET firstname = '$fname', lastname= '$lname',phoneNum = '$email',adress = '$position' WHERE id\_user = {$\_SESSION['id\_user']}";

$resultnova = $connect->query($query);

if($resultnova){

$last\_id = $connect->insert\_id;

// echo "Svaka , čast znaš da uneseš sve podatka , kako treba (unešeni su podaci u sql)";

header("Location:acount.php");

}

}

else

{

header("Location:index.php");

die;

}

Ako zelimo da se izlogujemo sledeci kod nam omogucuje upravo to :

<?php

session\_start();

$\_SESSION = [];

if (ini\_get("session.use\_cookies")) {

$params = session\_get\_cookie\_params();

setcookie(session\_name(), '', time() - 42000,

$params["path"], $params["domain"],

$params["secure"], $params["httponly"]

);

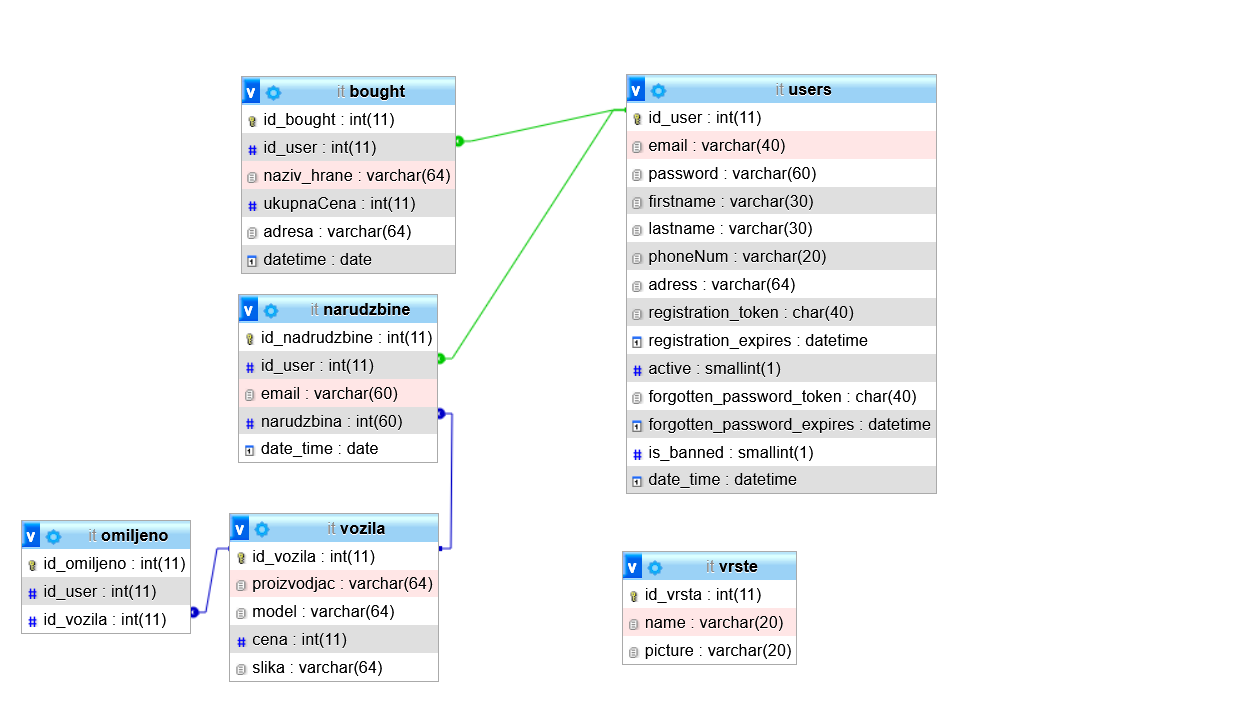
}

session\_destroy();

header("Location:login.php?l=5");

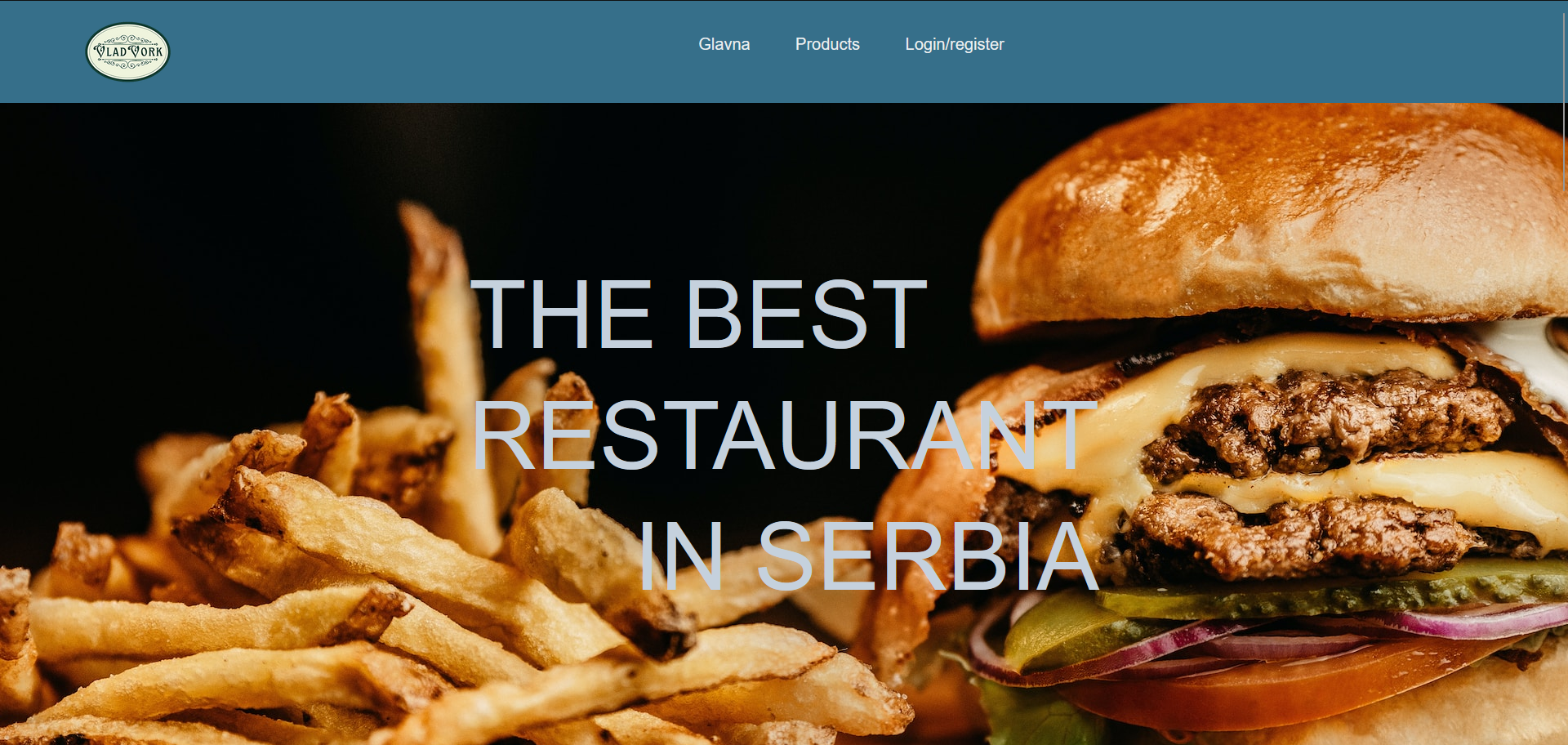
exit();

**STRUKTURA BAZE PODATAKA (MySQL Workbench)**



**OPIS FUNKCIONALNOSTI**

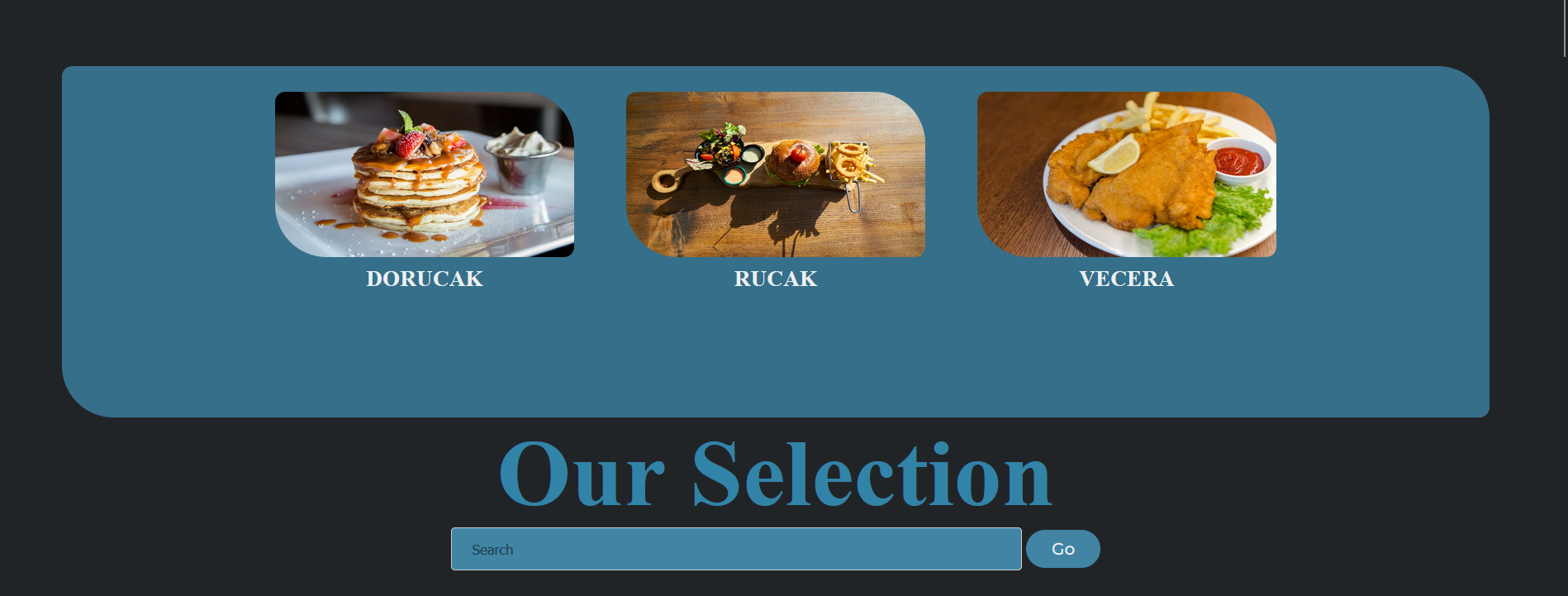
Kada se otvori web-site otvori se index.php



Kada smo došli na index.php možemo da se ulogujemo ili da prelistamo sva jela

Imamo opciju da izaberemo po kategorijama (ako izaberemo doručak samo će nam doručak izbaciti)

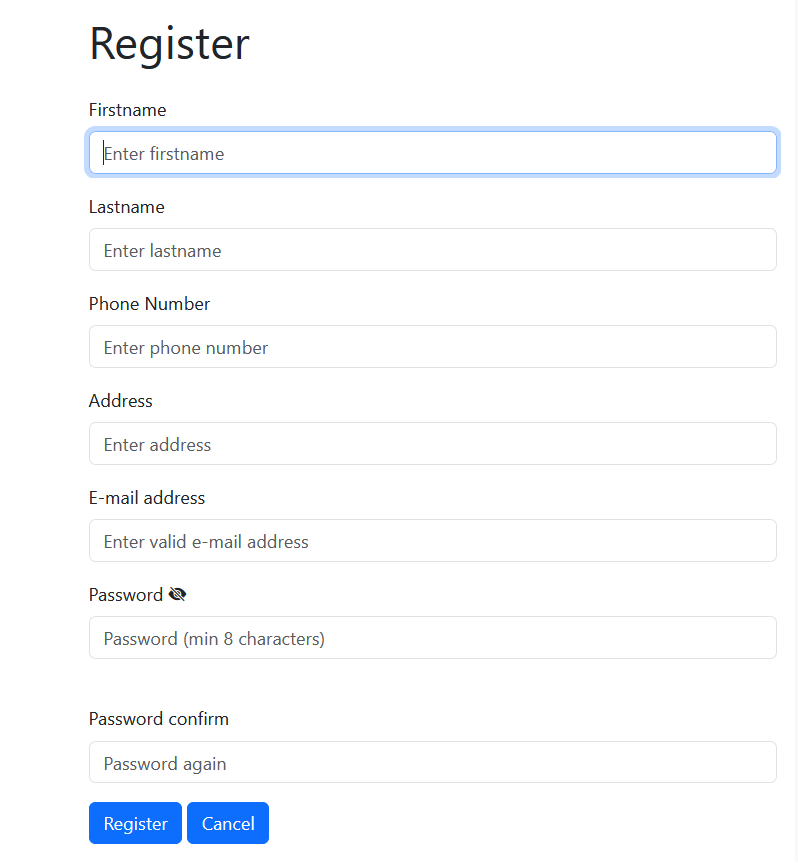
Imamo mogućnost i detaljne pretrage





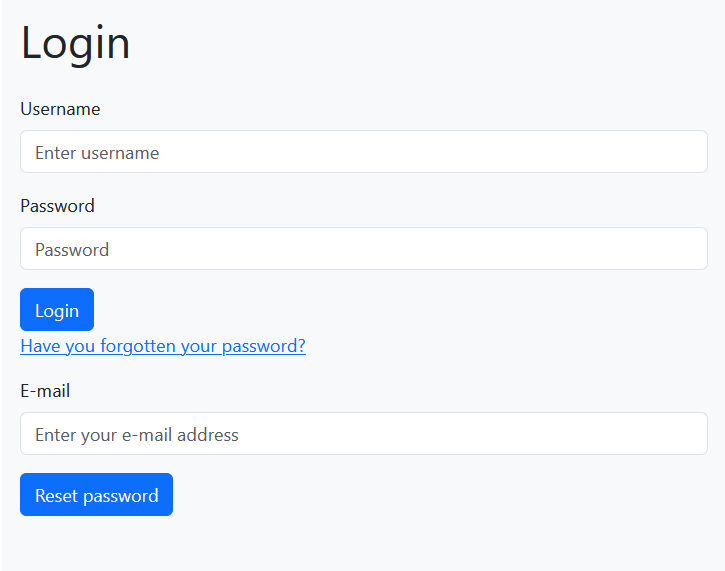
Ako se klikne na buy odmah nas prebaci na logIn register.

Mozemo se registrovati , radi se validacija (strong password , email etc)

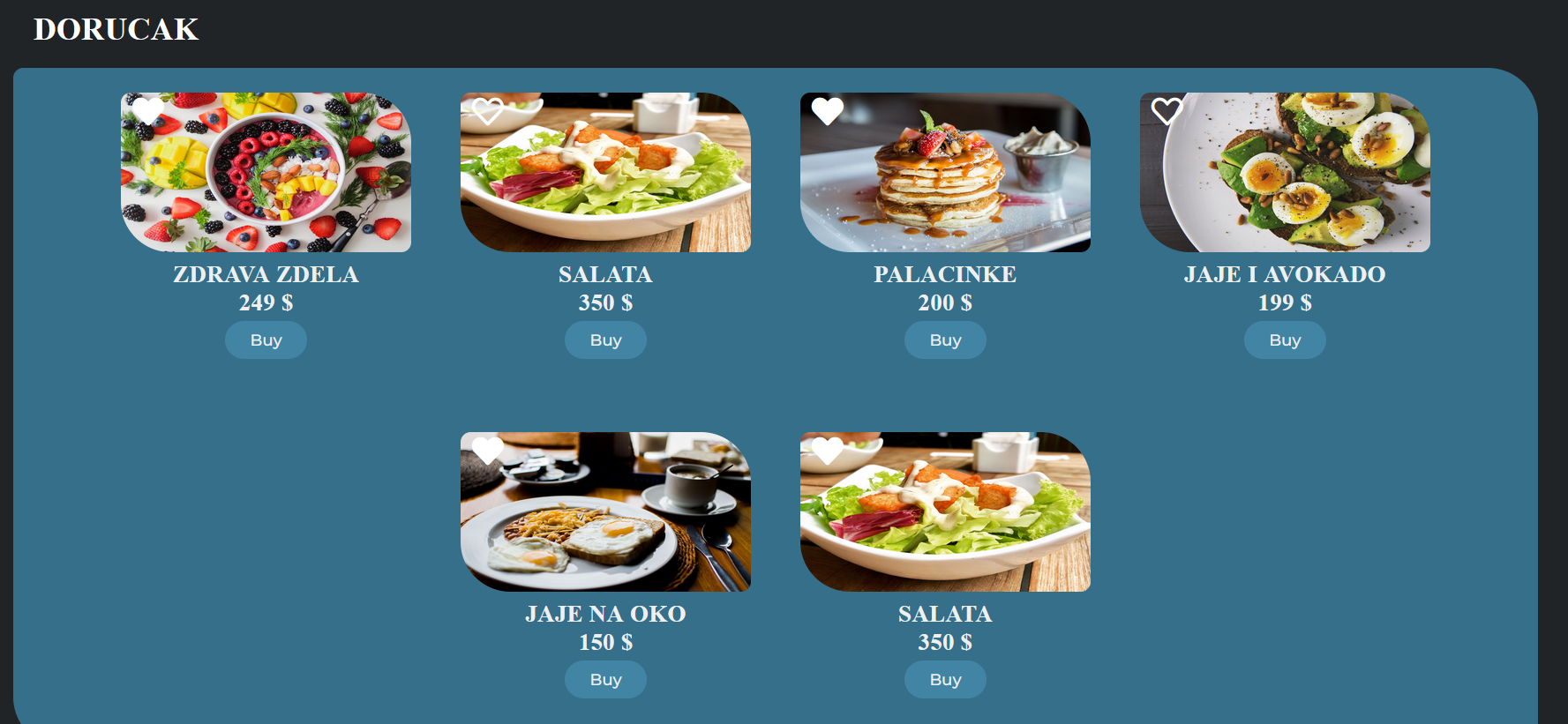


Kada se registrujemo bez problema , salje nam se na mailtrap email sa verifikacijonim kodom

Nakon toga možemo se ulogovati



Kada se ulogujemo dobijamo pregršt opcija kao što su favorite meal



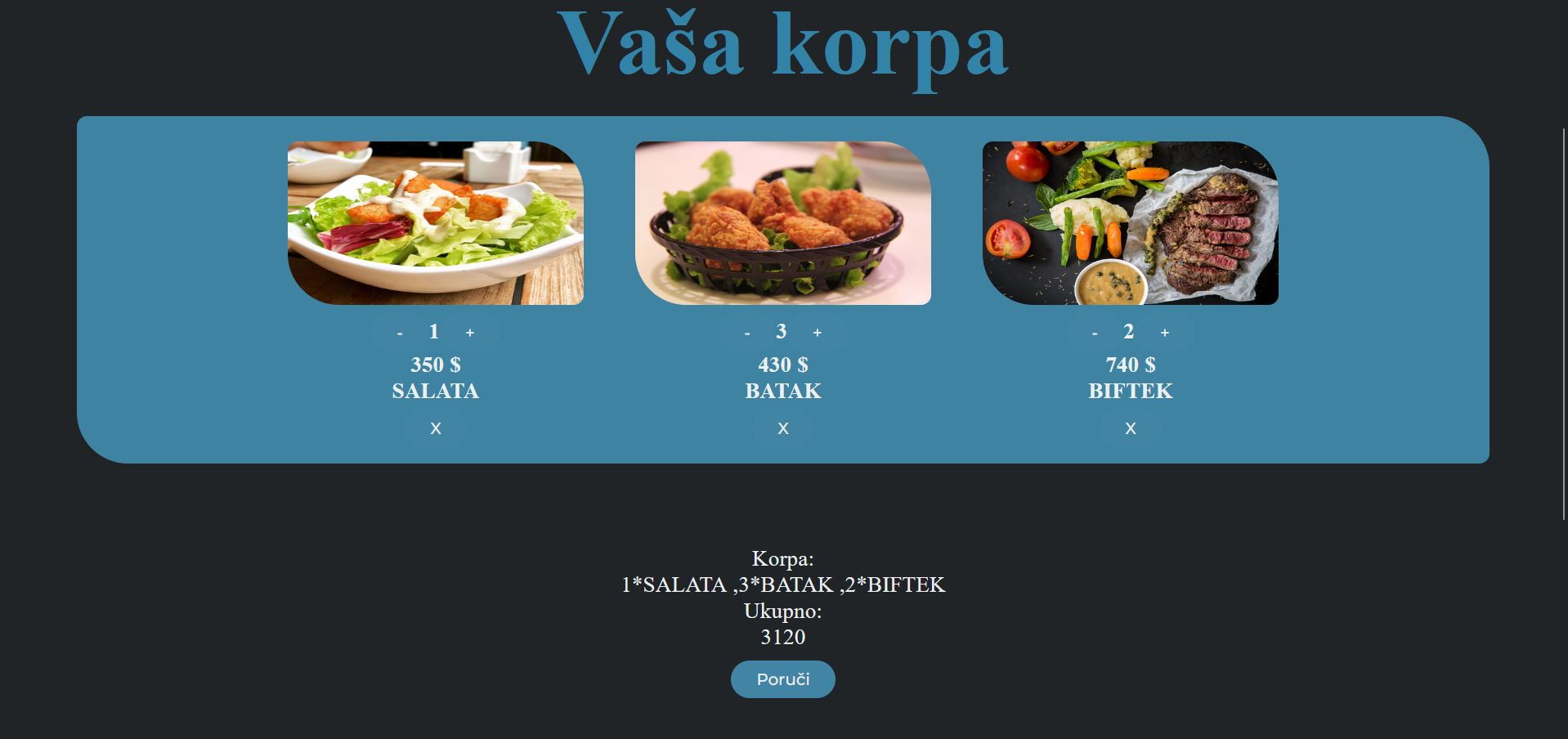
Naravno možemo i konačno kupiti jela

Kada kliknemo na buy , program unosi u Cart

Ukoliko je Cart prazan :

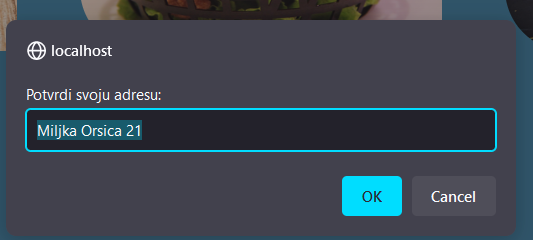


Ako nam nije prazna:



Možemo povećati količinu , ili izbaciti iz korpe

Kada kliknemo na poruči iskače nam prozor gde moramo potvrditi adresu!



Kada potvrdimo iz Cart-a nam se brišu svi elementi i ubavuju u bought tabelu

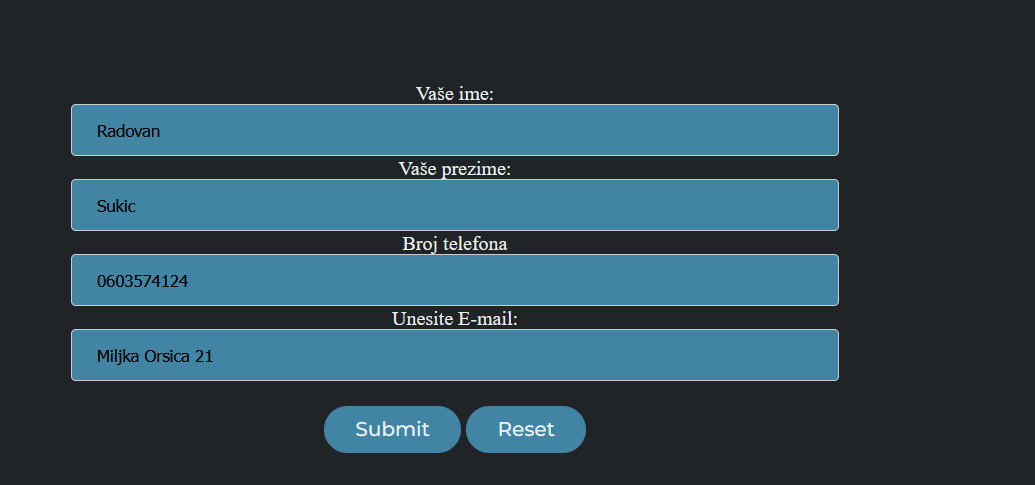


Imamo i Favourite meals , to su nam omiljena jela koja smo dodali pre koju čas .

Kao ulogovan korisnik imamo još mogućnost da izlistamo sve nape narudžbine



Korisnik može i da izmeni svoje podatke

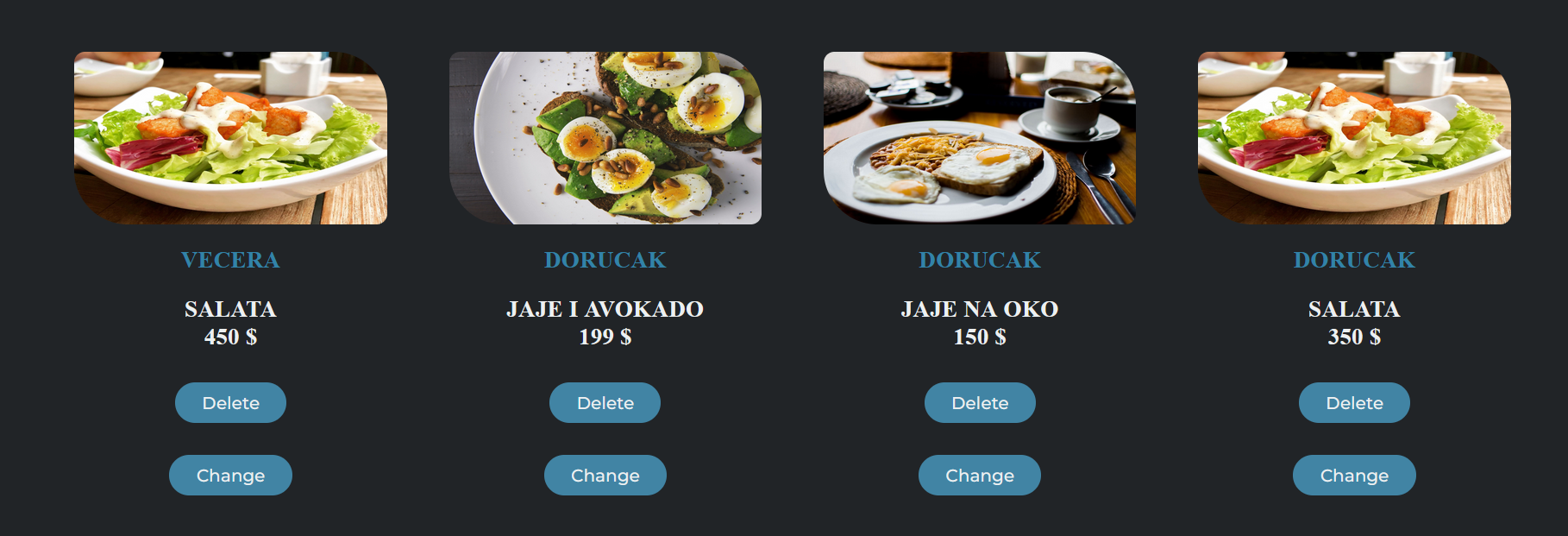


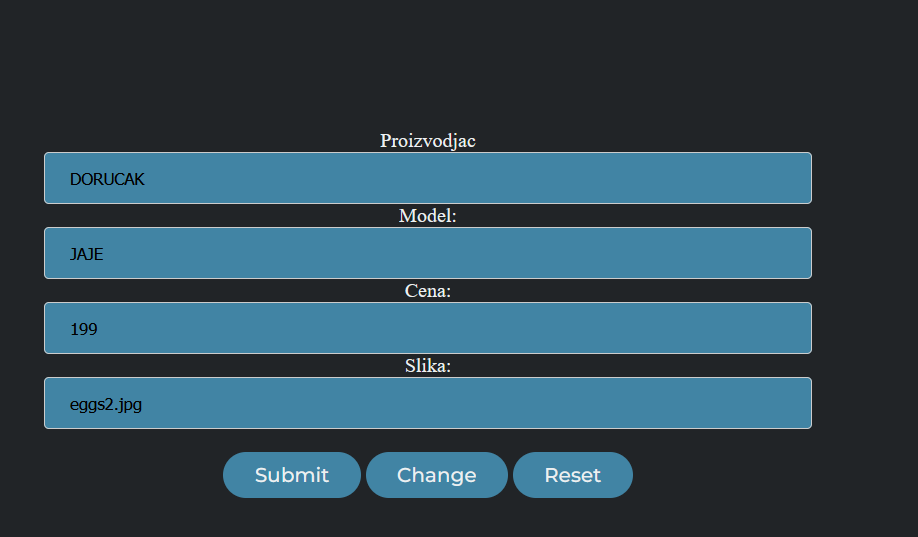
**Admin** ima mogućnost da izlista sve narudžbine

Takođe može po datumu da izlista narudžbine

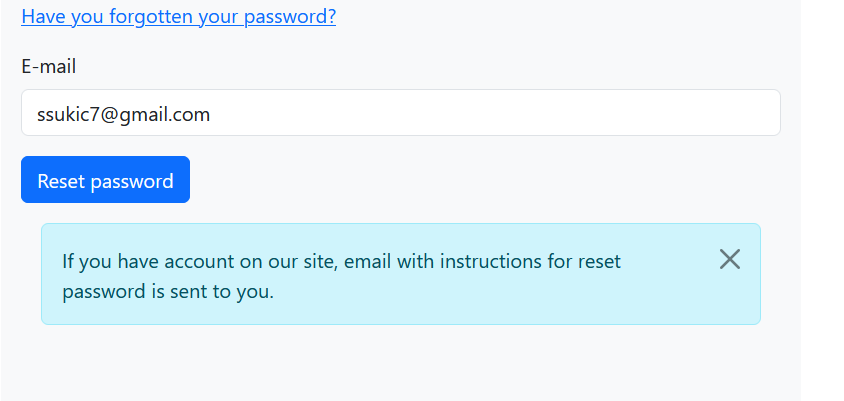


Admin takođe može da izmeni podatke ili da ubaci nove

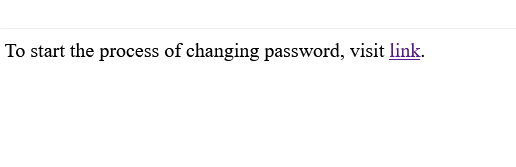




U slučaju da smo zaboravili šifru imamo mogućnost da je povratimo



Dobijamo poruku na Mailtrap



# KORIŠĆENA LITERATURA

W1. <https://mailtrap.io/home>

W2. <https://www.w3schools.com/bootstrap/default.asp>

W3. <https://www.w3schools.com/js/default.asp>

W4. <https://www.w3schools.com/sql/default.asp>

W5. <https://www.w3schools.com/php/default.asp>

W6. <https://www.w3schools.com/html/default.asp>

W7. <http://www.jquery.com>

W8. <https://www.w3schools.com/js/js_ajax_intro.asp>

W9. <https://people.vts.su.ac.rs/~chole/wp/primeri-peldak/>