**DOKUMENTACIJA**

Igor Jovanovic 127/18

UVOD

* 1. Korišćceni programski jezici:

HTML, CSS, native Javascript, Jquery, PHP, Sajt je pisan u programu Visual Studio Code

* 1. Funcionalnosti sajt su:

Sajt je namenjen za online shopping raznih proizvoda. Na svakoj stranici je učitan navigacioni meni iz baze. Na stranici index.php?page=home su prikazana 3 nasumično izabrana proizvoda iz baze. Na stranici index.php?page=products su prikazani svi proizvodi kao i njihova podela u paginaciji, takođe postoji i mogućnost filtriranja po kategorijama, sortiranja po ceni, i pretraga proizvoda po nazivu. Na stranici index.php?page=about se nalazi statički tekst o autoru. Na stranici index.php?page=register se nalazi forma za registraciju korisnika.

Na svakoj stranici se takođe prikazuje forma za logovanje korisnika, kao i link ka registraciji . Ako korisnik unese dobar username,ali ne password, dobija poruku na email adresu o grešci za logovanje. Kada se korisnik/admin uloguje u navigacionom meniju se pojavljuje link za logout kao i link ka user/admin stranici.

Na stranici index.php?page=user se nalaze sve informacije o ulogovanom korisniku. Na ovoj stranici korisnik može da promeni bilo koji deo svog naloga pod uslovom da unese validne podatke.

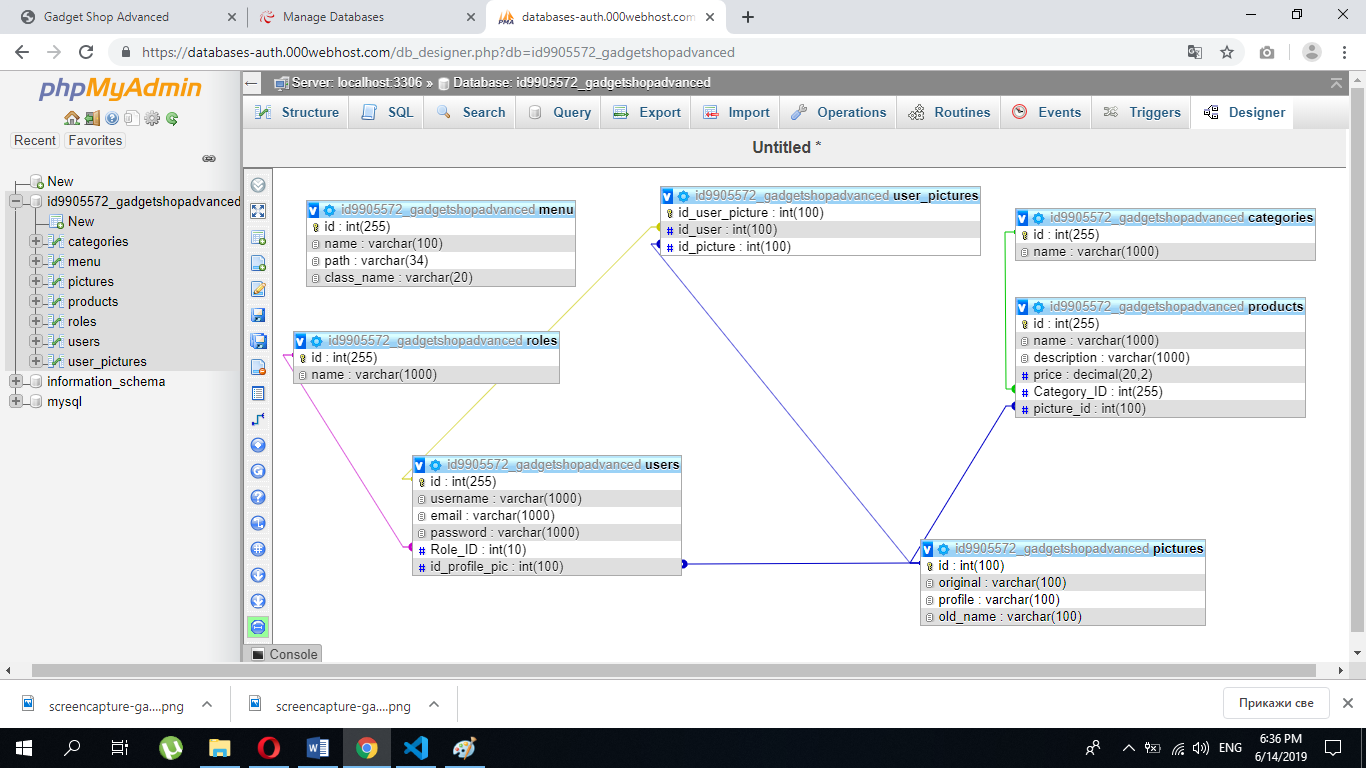
Na stranici index.php?page=admin se nalaze forme za unos novih proizvoda i korisnika, kao i mogućnost promene vec postojećih. Takođe je prikazana posećenost svakoj stranici u poslednjih 24h.

* 1. Tamplate:

Ovaj Tamplate je korišćen za izradu svake stranice izuzev saržaja koji se razlikuje.Detaljniji kod će biti prikazan u stavci KODOVI

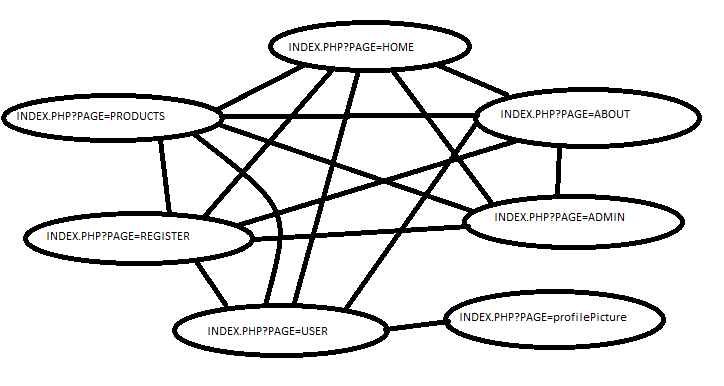


BAZA PODATAKA



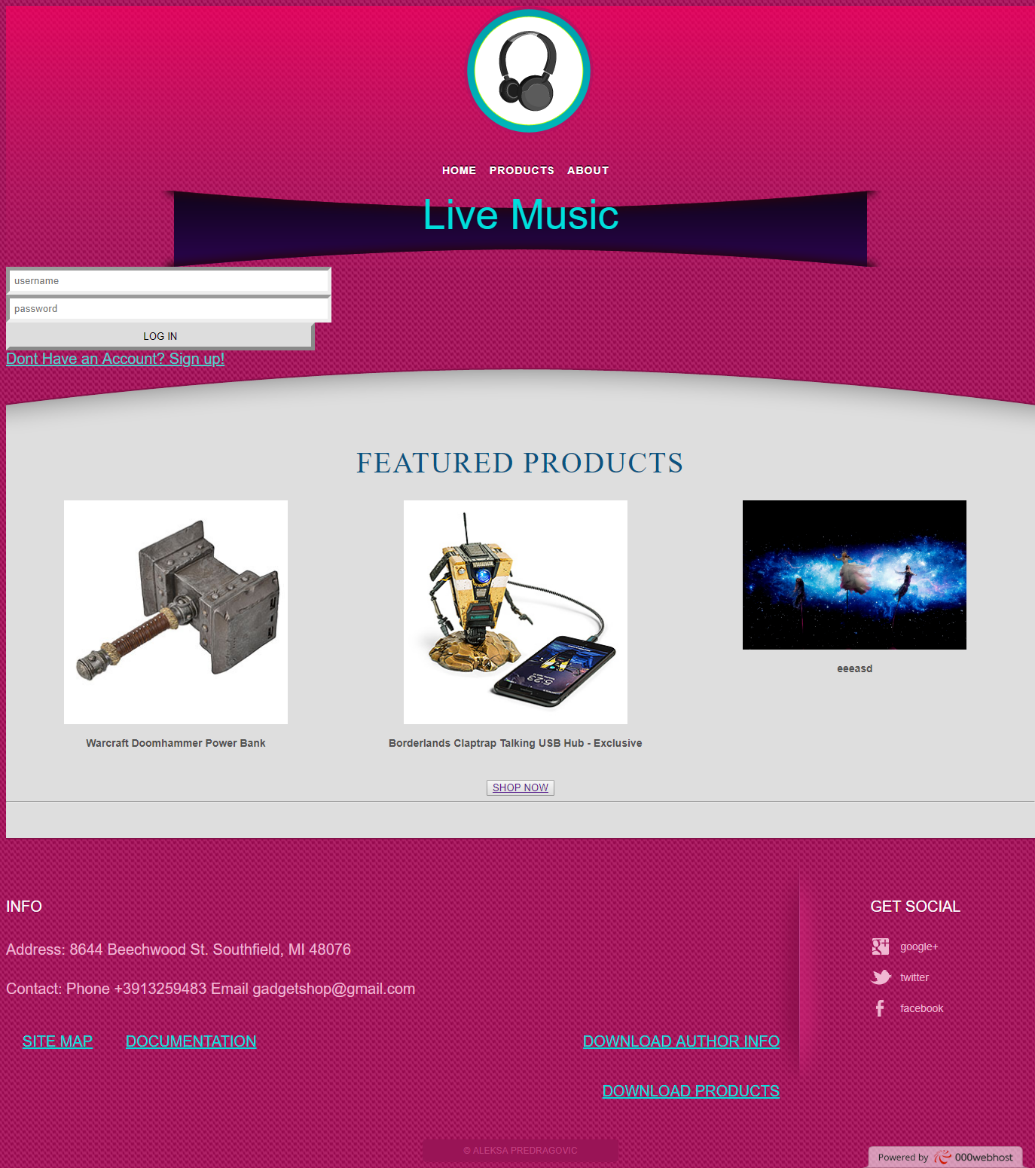
ORGANIZACIJA

2.1 Organizaciona šema



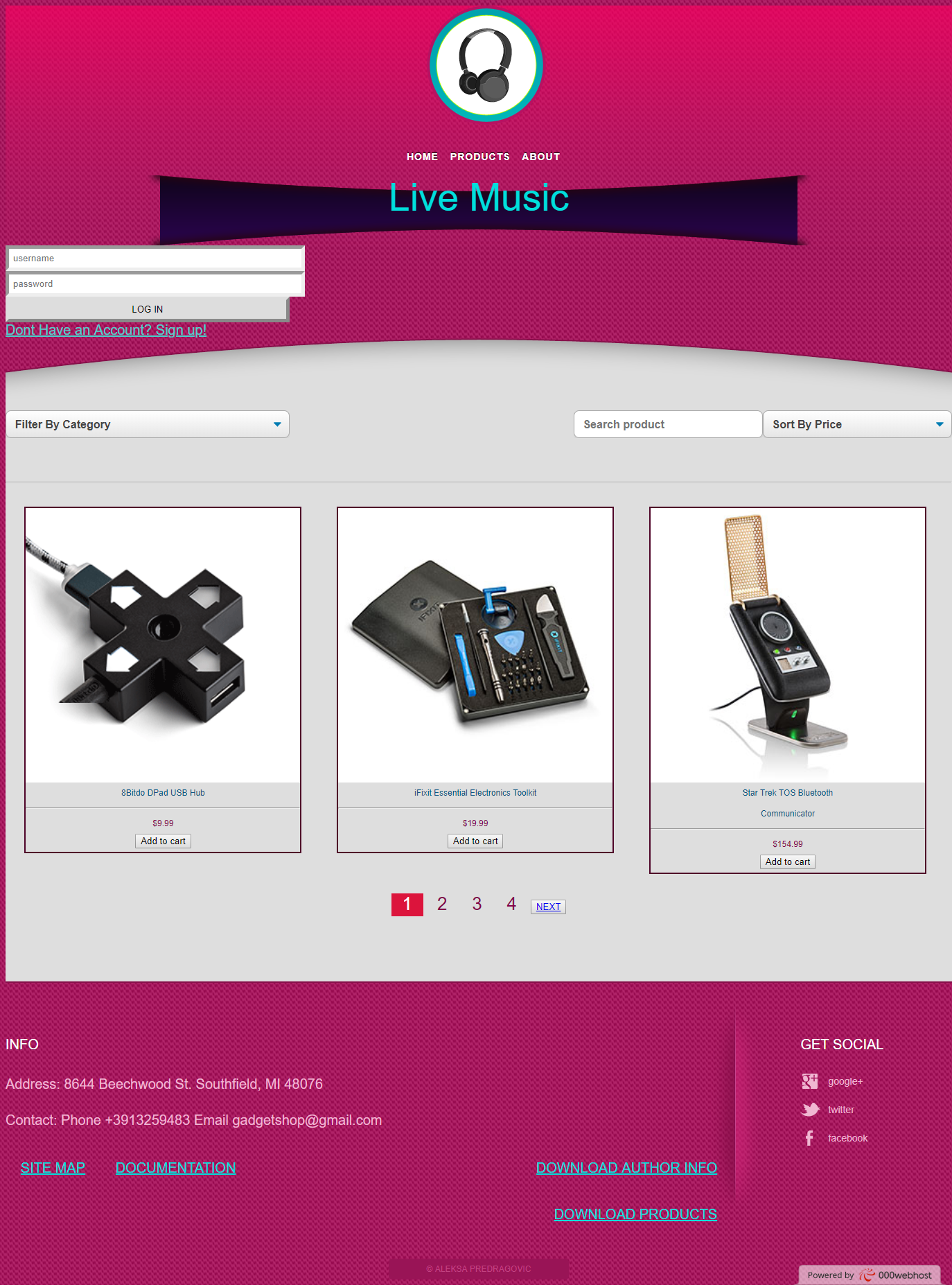
2.3 Funcionalnosti stranica

**Index.php?page=home**

**

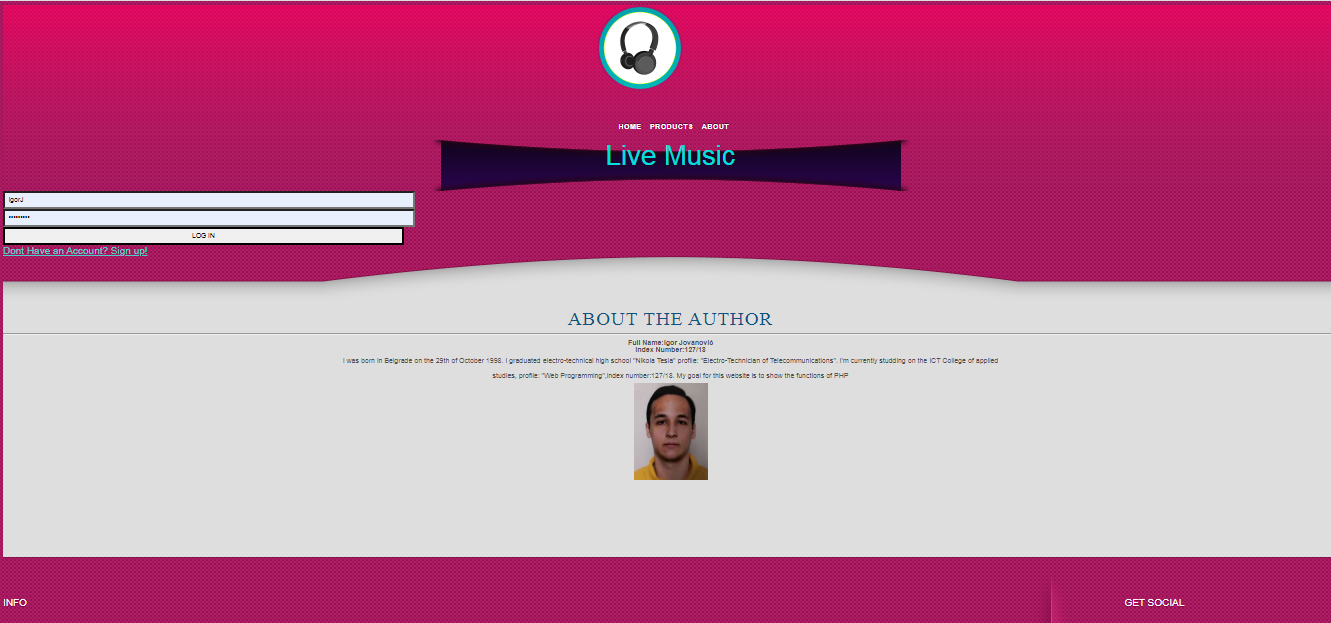
Na ovoj stranici se nalaze 3 nasumično izabrana proizvoda iz baze podataka. Takođe postoji i forma za logovanje korisnika, i link ka registraciji

**Index.php?page=products**

****

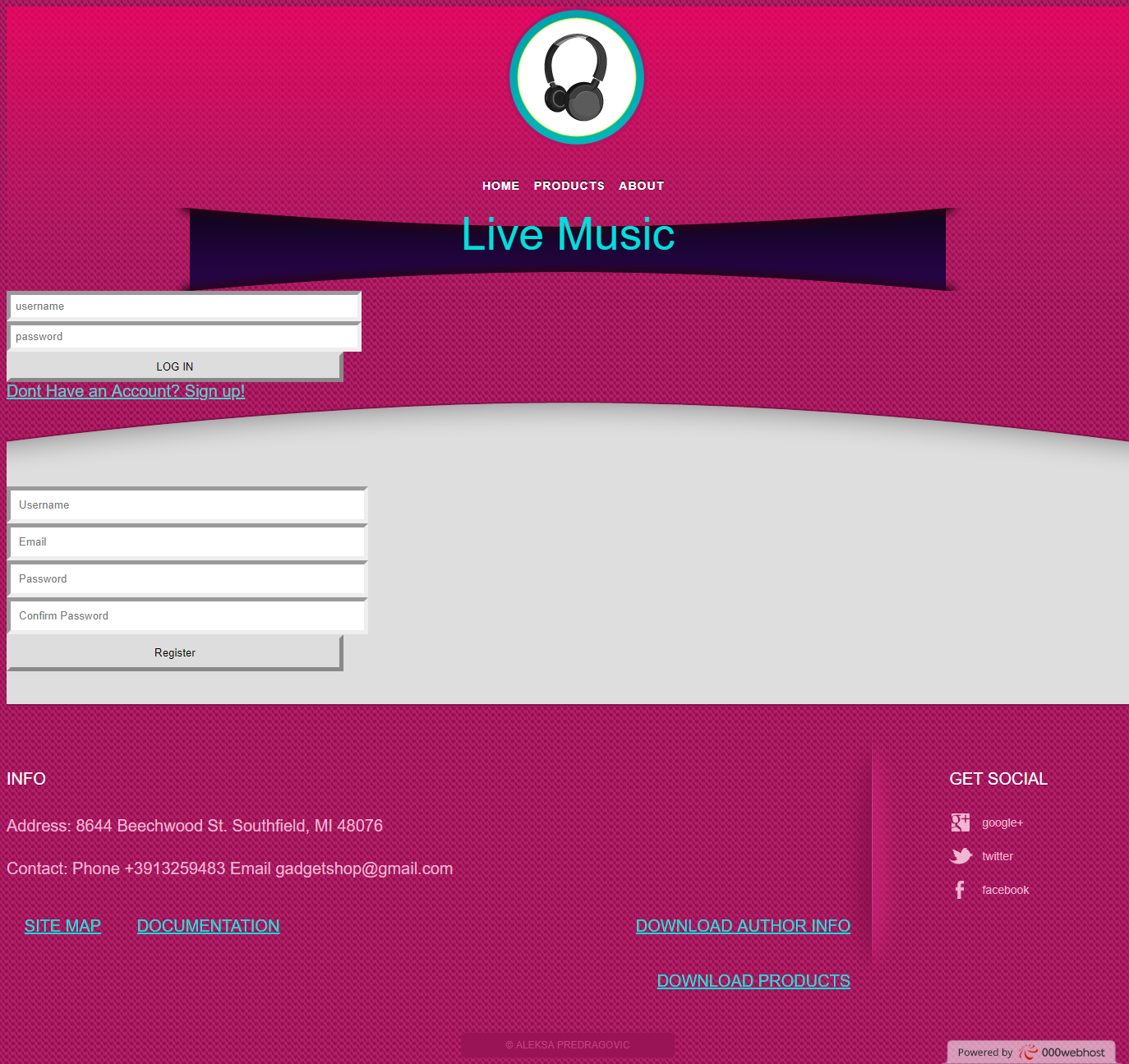
Na ovoj stranici se nalaze svi proizvoid podeljeni po paginaciji – 3 po strani, kao i polja za filtriranje proizvoda po kategoriji i nazivu i sortiranje po ceni. Takođe postoji i forma za logovanje korisnika, i link ka registraciji

**Index.php?page=about**

****

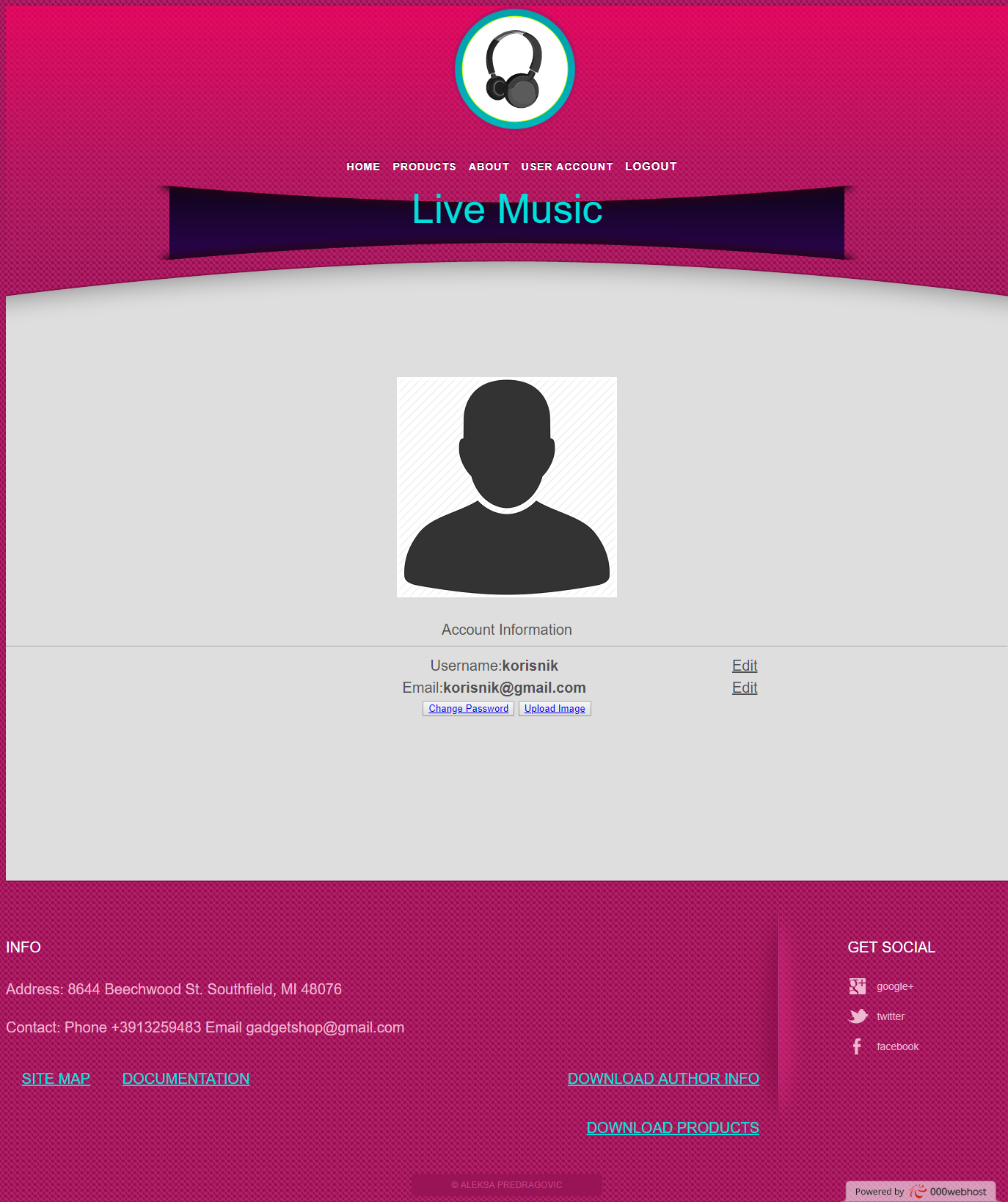
Na ovoj stranici se nalaze sve informacije o autoru sajta. Takođe postoji i forma za logovanje korisnika, i link ka registraciji

**Index.php?page=register**

****

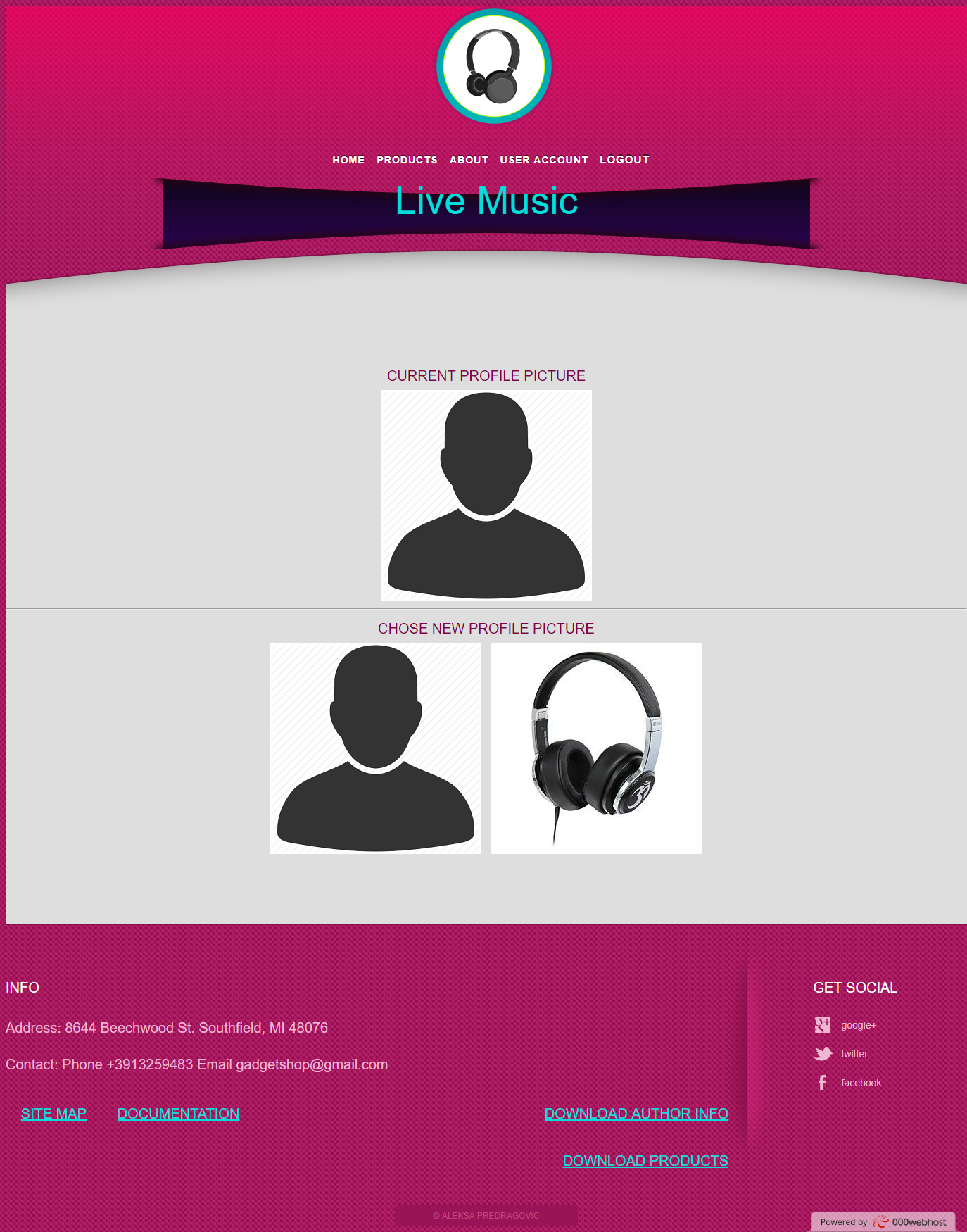
Na ovoj stranici se nalazi forma za registraciju korisnika. Forma sadrži i klijentsku i serversku validaciju podataka. Kada su svi podaci uneti ispravno, tada se zapisuju u bazu i korisnik moze da se uloguje.

**Index.php?page=user**

****

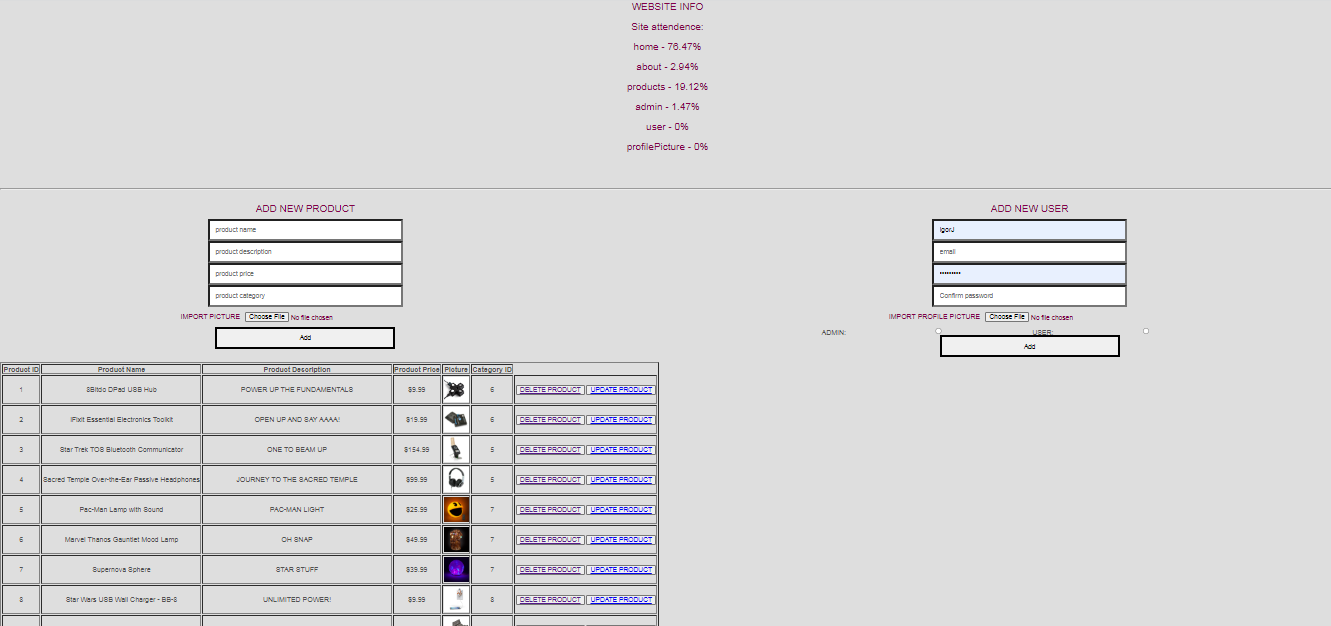
Na ovoj stranici se nalaze sve informacije o ulogovanom korisniku. Korisnik ovde može da promeni svoj username, email, i šifru. Može i da postavi svoju sliku koja će se sačuvati na serveru. Klikom na sliku ga vodi ka stranici za promenu profilne slike.

**Index.php?page=profilePicture**

****

Na ovoj stranici se nalaze sve slike ulogovanog korisnika. Ovde korisnik može da promeni svoju profilnu sliku odabirom željene slike iz kolkecije.

**Index.php?page=admin**

****

Na ovoj stranici se nalaze forme za dodavanje novih proizvoda i korisnika. U tabelama se nalaze dugmići za brisanje i ažuriranje odredjenog proizvoda ili korisnika. Prilikom ažuriranja se proveravaju uneti podaci, i ako je administrator slučajno napravio gresku, ona ce mu se ispisati na vrhu stranice. Na vrhu stranice se takođe ispisuju posećenosti stranicama u proteklih 24h.

**KODOVI**

SVI KODOVI KOJI SU PREUZETI SA NETA SU OBOJENI CRVENO

Index.php

<?php

session\_start();

include "config/connection.php";

include "models/products/functionsProducts.php";

include "models/user/functionsUser.php";

include "models/categories/functionsCategories.php";

include "views/fixed/header.php";

if(!isset($\_SESSION["account"])):

include "views/fixed/loginForm.php";

endif;

include "views/fixed/beginBodyDiv.php";

if(isset($\_GET["page"])):

switch($\_GET["page"]):

case "products":

include "views/pages/products.php";

break;

case "about":

include "views/pages/about.php";

break;

case "user":

if(isset($\_SESSION["account"])):

if($\_SESSION["account"]->Role\_ID == 2)

include "views/pages/user.php";

else

http\_response\_code(404);

endif;

break;

case "admin":

if(isset($\_SESSION["account"])):

if($\_SESSION["account"]->Role\_ID ==1)

include "views/pages/admin.php";

else

http\_response\_code(404);

endif;

break;

case "register":

include "views/fixed/registerForm.php";

break;

case "profilePicture":

if(isset($\_SESSION["account"])):

if($\_SESSION["account"]->Role\_ID == 2)

include "views/pages/profilePicture.php";

break;

else:

http\_response\_code(404);

endif;

default:

include "views/pages/home.php";

break;

endswitch;

else:

include "views/pages/home.php";

endif;

include "views/fixed/endBodyDiv.php";

include "views/fixed/footer.php";

?>

Views/fixed/beginBodyDiv.php

<div id="body">

<div>

<div>

<div>

<div>

Views/fixed/endBodyDiv.php

</div>

</div>

</div>

</div>

</div>

Views/fixed/footer.php

<div id="footer">

<div>

<div>

<h3>INFO</h3>

<p class= "info">Address: 8644 Beechwood St. Southfield, MI 48076</p>

<p class="info">Contact: Phone +3913259483 Email gadgetshop@gmail.com</p>

<a class="a-info" href = "sitemap.xml">SITE MAP</a>

<a class="a-info" href = "documentation.pdf">DOCUMENTATION</a>

<a class ="a-infoR" href = "models/user/getMe.php">DOWNLOAD AUTHOR INFO</a>

<a class ="a-infoR" href = "models/products/getAllExcel.php">DOWNLOAD PRODUCTS</a>

</div>

<div>

<h3>Get Social</h3>

<ul>

<li>

<a href="http://freewebsitetemplates.com/go/googleplus/" target="\_blank" id="googleplus">Google&#43;</a>

</li>

<li>

<a href="http://freewebsitetemplates.com/go/twitter/" target="\_blank" id="twitter">Twitter</a>

</li>

<li>

<a href="http://freewebsitetemplates.com/go/facebook/" target="\_blank" id="facebook">Facebook</a>

</li>

</ul>

</div>

</div>

<p class="footnote">

&copy; Igor Jovanovic

</p>

</div>

<script type ="text/javascript" src = "assets/js/jq.js"></script>

<script type ="text/javascript" src = "assets/js/index.js"></script>

</body>

</html>

Views/fixed/header.php

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gadget Shop Advanced</title>

<link rel="stylesheet" type="text/css" href="assets/css/style.css"/>

<meta name = "viewport" content = "width=device-width, initial-scale=1.0">

</head>

<body>

<div id="header">

<div>

<div id="logo">

<a href="index.php"><img src="assets/images/logo.png" alt="Logo"></a>

</div>

<ul id ="meni">

<?php

include "models/menu/showMenu.php";

?>

</ul>

</div>

<div>

<div id="figure">

<div>

<span id="background">

<h1>Live Music</h1>

</span>

</div>

</div>

</div>

</div>

Views/fixed/loginForm.php

<div id = "login">

<form action = "models/login/login.php" method = "post" class = "login">

<input type = "text" id = "username" name = "username" placeholder = "username"/></br>

<input type ="password" id = "password" name = "password" placeholder = "password"></br>

<input type = "submit" id = "posalji" name = "send" value = "LOG IN"/>

</form>

<?php if(isset($\_GET["error"])):?>

<p class ="error"><?= $\_GET["error"]; ?></p>

<?php endif;?>

<a class ="signUp" href = "index.php?page=register">Dont Have an Account? Sign up!</a>

</div>

Views/fixed/registerForm.php

<form id = "forma" action = "models/login/register.php" method = "post" class = "register">

<input type = "text" id = "usernameReg" name = "username" placeholder = "Username"/> </br>

<input type = "text" id = "email" name = "email" placeholder = "Email"/></br>

<input type = "password" id = "passwordReg" name = "password" placeholder = "Password"/></br>

<input type = "password" id = "Cpassword" name = "Cpassword" placeholder = "Confirm Password"/></br>

<input type = "submit" id = "send" name = "send" value = "Register"/>

</form>

Views/pages/about.php

<div id = "omeni"><h1>About the author</h1><hr>

<div id="tekst"> <b>Full Name:Igor Jovanović</b></br>

<b>Index Number:127/18</b>

<p>I was born in Belgrade on the 29th of October 1998. I graduated electro-technical high school "Nikola Tesla" profile: "Electro-Technician of Telecommunications".

I'm currently studding on the ICT College of applied studies, profile: "Web Programming",index number:127/18.

My goal for this website is to show the functions of PHP</p>

<img id = "ja" src = "assets/images/ja.png"/>

</div>

</div>

Views/pages/admin.php

<div id="article">

<?php

include "models/attendances/functions.php";

$attendance = pageAttendencePercent();

if(isset($\_GET["message"])):?>

<span class = "spanGreske"><?=$\_GET["message"]?></span>

<?php endif;?>

<div class = "siteInfo">

<p class ="pText">WEBSITE INFO</p>

<p class = "pText">Site attendance:</p>

<?php foreach($attendance as $key => $value):?>

<p class = "pText"><?=$key . " - ". $value."%"?></p>

<?php endforeach;?>

</div><hr>

<form action = "models/products/insertProduct.php" method = "post" class = "registerP" enctype="multipart/form-data">

<p class ="pText">ADD NEW PRODUCT</p>

<input type = "text" name = "productName" id = "productName" placeholder = "product name"/></br>

<input type = "text" name = "productDesc" id = "productDesc" placeholder = "product description"/></br>

<input type = "text" name = "productPrice" id = "productPrice" placeholder = "product price"/></br>

<input type = "text" name = "productCat" id = "productCat" placeholder = "product category"/></br>

<span class ="spanText">IMPORT PICTURE<input type = "file" name = "productPicture"/></span></br>

<input type = "submit" name = "send" id = "send" value ="Add" />

</form>

<form action = "models/user/insertUser.php" method = "post" class = "registerU" enctype="multipart/form-data">

<p class ="pText">ADD NEW USER</p>

<input type = "text" name = "username" id = "userName" placeholder = "username"/></br>

<input type = "text" name = "email" id = "email" placeholder = "email"/></br>

<input type = "password" name = "password" id = "password" placeholder = "password"/></br>

<input type = "password" name = "passwordC" id = "passwordC" placeholder = "Confirm password"/></br>

<span class = "spanText">IMPORT PROFILE PICTURE<input type = "file" name = "fileUser"/></span></br>

ADMIN:<input class ="ne" type = "radio" name = "role" value= "1"/>

USER:<input class = "ne" type = "radio" name = "role" value= "2"/><br>

<input type = "submit" name = "send" value ="Add" />

</form>

<div class = "updateForma">

<p class = "pText">UPDATE PRODUCT</p>

<form action = "models/products/updateProduct.php" method = "post" class = "register" enctype="multipart/form-data">

<input type = "text" name = "name" placeholder = "product name"/></br>

<input type = "text" name = "description" placeholder = "product description"/></br>

<input type = "text" name = "price" placeholder = "product price"/></br>

<input type = "text" name = "category" placeholder = "product category"/></br>

<input type = "hidden" name = "idProduct" id = "idProduct"/>

<span class = "spanText">CHANGE PICTURE<input type = "file" name = "file"/></span>

<input type = "submit" name = "sendU" id = "sendU" value ="Update" />

</form>

</div>

<div class = "updateFormaU">

<p class = "pText">UPDATE USER</p>

<form action = "models/user/updateUser.php" method = "post" class = "register" enctype="multipart/form-data">

<input type = "text" name = "username" id = "usernameU" placeholder = "username"/></br>

<input type = "text" name = "email" id = "emailU" placeholder = "email"/></br>

<input type = "text" name = "password" id = "passwordU" placeholder = "password"/></br>

<input type = "text" name = "passwordC" id = "passwordUC" placeholder = "Confirm password"/></br>

<span class = "spanText">CHANGE PROFILE PICTURE<input type = "file" name = "fileUser"/></span></br>

<input type = "hidden" name = "idUser" id = "idUser"/>

ADMIN:<input class ="ne" type = "radio" name = "role" value= "1"/></br>

USER:<input class = "ne" type = "radio" name = "role" value= "2" /></br>

<input type = "submit" name = "adminUpdate" id = "sendUserUpdate" value ="Update" />

</form>

</div>

<div id="sidebar">

<table class = "tabela" border = "1px solid black" >

<tr>

<th>Product ID</td>

<th>Product Name</th>

<th>Product Description</th>

<th>Product Price</th>

<th>Picture</th>

<th>Category ID</th>

</tr>

<tbody id = "products">

<?php

$products = getAllProducts();

foreach($products as $product):

?>

<tr>

<td><?=$product->id?></td>

<td><?=$product->name?></td>

<td><?=$product->description?></td>

<td>$<?=$product->price?></td>

<td><img src = "<?=$product->original?>"/></td>

<td><?=$product->Category\_ID?></td>

<td>

<button><a data-id = "<?=$product->id?>" class = "delete" name = "delete" href = "" >DELETE PRODUCT</a></button>

<button><a data-id = "<?=$product->id?>" class = "update" name = "update" href = "#" >UPDATE PRODUCT</a></button>

</td>

</tr>

<?php endforeach;?>

</tbody>

</table>

<hr>

<table class = "tabela" border = "1px solid black" id = "users" >

<tr>

<th>User ID</th>

<th>Username</th>

<th>Email</th>

<th>Profile Picture</th>

<th>Role</th>

</tr>

<tbody id = "users">

<?php

$users = getAllUsers();

foreach($users as $user):

?>

<tr>

<td><?=$user->id?></td>

<td><?=$user->username?></td>

<td><?=$user->email?></td>

<td><img src = "<?=$user->profile?>"/></td>

<td><?=$user->name?></td>

<td>

<button><a data-id = "<?=$user->id?>" class = "deleteU" name = "deleteU" href = "" >DELETE USER</a></button>

<button><a data-id = "<?=$user->id?>" class = "updateU" name = "update" href = "#" >UPDATE USER</a></button>

</td>

</tr>

<?php endforeach;?>

</tbody>

</table>

</div>

</div>

</div>

Views/pages/home.php

<h1>Featured Products</h1>

<ul id = "featured">

<?php

$meni = getFeatured();

foreach($meni as $item):

?>

<li><img class ="slider" src = "<?=$item->original?>"/><p><b><?=$item->name?></b></p></li>

<?php endforeach; ?>

</ul>

<button class = "shopNow"><a href = "index.php?page=products">SHOP NOW</a></button><hr>

Views/pages/products.php

<?php

if(isset($\_GET["id"])):

$product = getOneProduct($\_GET["id"]);

?>

<div class = "oneProduct">

<img class = "productImg" src = "<?= $product->original?>"/>

<p class = "name"><?=$product->name?></p>

<p class = "description"><?=$product->description?></p>

<p class = "price">$<?=$product->price?></p>

</div>

<?php else:?>

<div class ="filter-sort">

<div class="filter-categories">

<select id = "filter" class = "select-css">

<option value="0">Filter By Category</option>

<?php

$categories = getAllCategories();

foreach($categories as $category):

?>

<option value = "<?=$category->id?>"><?=$category->name?></option>

<?php endforeach;?>

</select>

</div>

<div class="sort-price">

<select id="sortPrice" class = "select-css">

<option value="0">Sort By Price</option>

<option value="1">Ascending</option>

<option value="2">Descending</option>

</select>

</div>

<div class="sort-name">

<input type = "text" id = "sortName" placeholder = "Search product" class = "select-css-name" />

</div>

</div>

<hr>

<ul id = "products">

<?php

$limit = 0;

$products = getAllProducts($limit);

foreach($products as $product):

?>

<li>

<div class = product>

<a href = "index.php?page=products&id=<?=$product->id?>"><img src = "<?= $product->original?>"/></a>

<p class="name"><?= $product->name?></p><hr>

<p class ="price">$<?= $product->price?></p>

<input class = "id" type ="hidden" value = "<?=$product->id?>">

<input class = "idC" type ="hidden" value = "<?=$product->Category\_ID?>">

<input type = "button" value = "Add to cart"/>

</div>

</li>

<?php endforeach;?>

</ul>

<?php endif;?>

<div class = "pages">

<?php

$numOfPages = getPaginationNum();

for($i = 0;$i<$numOfPages;$i++):

if($i == 0):

?>

<a href ="#" class = "page active" data-limit = "<?=$i?>"><?=$i + 1?></a>

<?php else:?>

<a href = "#" class = "page" data-limit = "<?=$i?>"><?=$i+1?></a>

<?php endif; endfor;?>

<button><a class = "next" href = "#">NEXT</a></button>

</div>

Views/pages/profilePicture.php

<?php

$user = getOne($\_SESSION["account"]->id);

$userPictures = getPictures($\_SESSION["account"]->id);

?>

<div id="article">

<div id = "message">

<p><?=@ $\_GET["message"];?></p>

</div>

<p class = "pText">CURRENT PROFILE PICTURE</P>

<img src = "<?=$user->profile?>"/>

<hr>

<p class = "pText">CHOSE NEW PROFILE PICTURE</p>

<?php foreach($userPictures as $picture):?>

<a href = "models/user/updateUser.php?idUser=<?=$\_SESSION["account"]->id?>&idPicture=<?=$picture->id?>"><img class = "profilePicture" src = "<?=$picture->profile?>"/></a>

<?php endforeach;?>

</div>

Views/pages/user.php

<?php

$user = getOne($\_SESSION["account"]->id);

?>

<div id="article">

<div id = "message">

<p><?=@ $\_GET["message"];?></p>

</div>

<a href="index.php?page=profilePicture"><img src = "<?=$user->profile?>" id = "profilePic"/></a>

<div class ="userInfo">Account Information<hr>

<p>Username:<b><?=$user->username?></b><a class= "editLink" id = "editUsername" href = "#">Edit</a></p>

<p>Email:<b><?=$user->email?></b><a class ="editLink" id = "editEmail" href = "#">Edit</a></p>

<button><a id = "editPass" href = "#">Change Password</a></button>

<button><a id = "uploadImage" href = "#">Upload Image</a></button>

<form class = "editForme" id = "formPass" action = "models/user/updateUser.php" method = "post">

<input type = "password" name = "editPass" placeholder = "New Password"/>

<input type = "password" name = "editPass2" placeholder = "Repeat new Password"/>

<input type = "submit" name ="sendPassword"/>

</form>

</div>

</div>

<div id="sidebar2">

<form class = "editForme" id = "formUsername" action = "models/user/updateUser.php" method = "post">

<input type = "text" name = "editUsername" placeholder = "New Username"/>

<input type = "submit" name ="sendUsername"/>

</form></br>

<form class = "editForme" id = "formEmail" action = "models/user/updateUser.php" method = "post">

<input type = "text" name = "editEmail" placeholder = "New Email"/>

<input type = "submit" name ="sendEmail"/>

</form>

<form action = "models/user/updateUser.php" method = "post" class = "editForme" id = "formPicture" enctype="multipart/form-data">

<p class ="pText">UPLOAD IMAGE TO PROFILE</p>

<input type = "file" name = "file"/>

<input type = "submit" name = "sendPicture" value ="Upload Picture" />

</form>

</div>

models/attendances /functions.php

<?php

function pageAttendencePercent(){

$array = ["home"=>"","about"=>"","products"=>"","register"=>"","admin"=>"","user"=>"","profilePicture"=>""];

$sum = 0;

$home = 0;

$about = 0;

$products = 0;

$admin = 0;

$profilePicture = 0;

$user = 0;

$register = 0;

$oneDayAgo = strtotime("1 day ago");

$file = file(VISITLOG);

if(count($file)):

foreach($file as $row):

$pieces = explode("\t",$row);

$url = explode(".php",$pieces[0]);

$page = explode("&",@ $url[1]);

if($pieces[1] >= $oneDayAgo):

switch($page[0]):

case "":$home++;$sum++;break;

case "?page=home":$home++;$sum++;break;

case "?page=about":$about++;$sum++;break;

case "?page=products":$products++;$sum++;break;

case "?page=admin":$admin++;$sum++;break;

case "?page=user":$user++;$sum++;break;

case "?page=profilePicture":$profilePicture++;$sum++;break;

case "?page=register":$register++;$sum++;break;

default:$home++;$sum++;break;

endswitch;

endif;

endforeach;

if($sum>0):

$array ["home"] = round($home\*100/$sum,2);

$array ["about"] = round($about\*100/$sum,2);

$array ["products"] = round($products\*100/$sum,2);

$array ["admin"] = round($admin\*100/$sum,2);

$array ["user"] = round($user\*100/$sum,2);

$array ["profilePicture"] = round($profilePicture\*100/$sum,2);

$array ["register"] = round($register\*100/$sum,2);

endif;

endif;

return $array;

}

?>

models/categories /functionsCategories.php

<?php

function getAllCategories(){

return executeQuery("select \* from categories");

}

?>

models/login/login.php

<?php

use PHPMailer\PHPMailer\PHPMailer;

use PHPMailer\PHPMailer\Exception;

if(isset($\_POST["send"])):

include "../../config/connection.php";

include "../../phpmailer/vendor/autoload.php";

session\_start();

$username = $\_POST["username"];

$password = md5($\_POST["password"]);

try{

$upit = "SELECT \* FROM users

WHERE username = ? AND password = ?";

$prepared = $conn->prepare($upit);

$prepared->execute([$username,$password]);

$user = $prepared->fetch();

}

catch(Exception $e){

$error = $e->getMessage();

logUserError("database",$error);

header("location:../../index.php?error=".$error);

}

if($user):

$\_SESSION["account"] = $user;

header("location:../../index.php");

else:

$query = "SELECT \* FROM users where username = ?";

$prep= $conn->prepare($query);

$prep->execute([$username]);

$found = $prep->fetch();

if($found){

try{

$mail = new PHPMailer(TRUE);

$mail->setFrom("gadget.shop.advanced@gmail.com","Admin");

$mail->addAddress($found->email,"User");

$mail->Subject = "Incorrect login";

$mail->Body = "Failed to login to gadget shop advanced,incorrect password";

$mail->isSMTP();

$mail->Host = "smtp.gmail.com";

$mail->Port = 465;

$mail->SMTPAuth = true;

$mail->Username = "gadget.shop.advanced@gmail.com";

$mail->Password = "predragovic98";

$mail->SMTPSecure = "ssl";

if(!$mail->send()){

$error = "username or password did not match";

header("location:../../index.php?error=".$error.",".$mail->ErrorInfo);

}

}

catch(Exception $e){

$error = $e->getMessage();

logUserError("mail",$error);

header("location:../../index.php?error=".$error);

}

}

$error = "username or password did not match";

logUserError("login",$error);

header("location:../../index.php?error=$error");

endif;

else:

http\_response\_code(400);

endif;

?>

models/login/logout.php

<?php

session\_start();

if(isset($\_SESSION["account"])):

$\_SESSION["account"] = null;

header("location:../../index.php");

endif;

?>

models/login/register.php

<?php

if(isset($\_POST["username"])):

include "../user/functionsUser.php";

include "../../config/connection.php";

header("content-type:application/json");

$username = $\_POST["username"];

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

$password = $\_POST["password"];

$Cpassword = $\_POST["Cpassword"];

$register = userRegister($username,$email,$password,$Cpassword);

if($register):

echo json\_encode($register);

http\_response\_code(200);

else:

echo json\_encode($register);

http\_response\_code(422);

endif;

else:

http\_response\_code(400);

endif;

?>

models/menu/showMenu.php

<?php

$meni = executeQuery("SELECT \* FROM menu");

foreach($meni as $item):

if($item->class\_name == "admin" || $item->class\_name == "user"):

if(!empty($\_SESSION["account"])):

if($\_SESSION["account"]->Role\_ID == 1 && $item->class\_name == "admin"):

?>

<li class = "<?=$item->class\_name?>" ><a href = "<?=$item->path?>"><?=$item->name?></a></li>

<li class = "logoutli" >

<a class ="logout" href = "models/login/logout.php">LOGOUT</a>

</li>

<?php

elseif($\_SESSION["account"]->Role\_ID == 2 && $item->class\_name == "user"):

?>

<li class = "<?=$item->class\_name?>" ><a href = "<?=$item->path?>"><?=$item->name?></a></li>

<li class = "logoutli" >

<a class ="logout" href = "models/login/logout.php">LOGOUT</a>

</li>

<?php endif; endif; else: ?>

<li class ="<?=$item->class\_name?>"><a href ="<?=$item->path ?>"><?=$item->name?></a></li>

<?php endif; endforeach;?>

models/products/deleteProduct.php

<?php

if(isset($\_POST["id"])):

header("content-type:application/json");

include "functionsProducts.php";

include "../../config/connection.php";

$message = deleteProduct($\_POST["id"]);

echo json\_encode($message);

else:

http\_response\_code(400);

endif;

?>

models/products/filterCategory.php

<?php

if(isset($\_POST["limit"])):

header("content-type:application/json");

include "functionsProducts.php";

include "../../config/connection.php";

$id = $\_POST["id"];

$limit = $\_POST["limit"];

$value = $\_POST["value"] != "" ? $\_POST["value"] : null;

if($id == 0):

$filteredProducts = getAllProducts($limit);

$pages = getPaginationNum();

echo json\_encode([

"products"=>$filteredProducts,

"pages"=>$pages

]);

http\_response\_code(200);

else:

$filteredProducts = filterProducts($id,$value,$limit);

$pages = countProducts($id,$value);

echo json\_encode([

"products" => $filteredProducts,

"pages" => $pages

]);

http\_response\_code(200);

endif;

else:

http\_response\_code(400);

endif;

?>

models/products/filterName.php

<?php

if(isset($\_POST["value"])):

header("content-type:application/json");

include "../../config/connection.php";

include "functionsProducts.php";

$value = $\_POST["value"] != "" ? $\_POST["value"] : null;

$id = $\_POST["id"] > 0 ? $\_POST["id"] : null;

$limit = $\_POST["limit"];

$products = filterProducts($id,$value,$limit);

$pages = countProducts($id,$value);

echo json\_encode([

"products" => $products,

"pages" =>$pages

]);

http\_response\_code(200);

else:

http\_response\_code(400);

endif;

?>

models/products/functionsProducts.php

<?php

DEFINE("PRODUCT\_OFFSET",3);

function getFeatured(){

$products = getAllProducts();

$num = count($products);

$random = rand(0,$num-3);

return executeQuery("SELECT \* FROM products pr inner join pictures pic on pr.picture\_id = pic.id LIMIT $random,3");

}

function getAllProducts($limit = -1){

if($limit < 0):

return executeQuery("SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id");

else:

global $conn;

try{

$query = "SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id LIMIT :limit,:offset";

$prep = $conn->prepare($query);

$limit = (int)$limit \* PRODUCT\_OFFSET;

$offset = PRODUCT\_OFFSET;

$prep->bindParam(":limit",$limit,PDO::PARAM\_INT);

$prep->bindParam(":offset",$offset,PDO::PARAM\_INT);

$prep->execute();

$products = $prep->fetchAll();

return $products;

}

catch(Exception $e){

return $e->getMessage();

}

endif;

}

function getNumOfProducts(){

return executeQueryOneRow("SELECT COUNT(\*) as numOfProducts FROM products");

}

function getPaginationNum(){

$numOfProducts = getNumOfProducts();

$paginationNum = $numOfProducts->numOfProducts;

return ceil($paginationNum / PRODUCT\_OFFSET);

}

function getOneProduct($id){

global $conn;

try{

$query = "SELECT \*,pr.id FROM products pr INNER JOIN pictures pic on pr.picture\_id = pic.id WHERE pr.id = ?";

$prep = $conn->prepare($query);

$prep->execute([$id]);

$product = $prep->fetch();

return $product;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function countProducts($id=null,$value=null){

global $conn;

$offset = PRODUCT\_OFFSET;

if($id != null && $value == null ):

try{

$prep = $conn->prepare("SELECT COUNT(\*) as numOfProducts FROM products pr inner join pictures pic on pr.picture\_id = pic.id where pr.Category\_ID = ?");

$prep->bindValue(1,$id);

$prep->execute();

$numOfProducts = $prep->fetch();

$pagination = $numOfProducts->numOfProducts;

return ceil($pagination / $offset);

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

elseif($id == null && $value != null):

try{

$prep = $conn->prepare("SELECT COUNT(\*) as numOfProducts FROM products pr inner join pictures pic on pr.picture\_id = pic.id where name like ?");

$prep->bindValue(1,"%$value%");

$prep->execute();

$numOfProducts = $prep->fetch();

$pagination = $numOfProducts->numOfProducts;

return ceil($pagination / $offset);

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

elseif($id == null && $value == null):

try{

$prep = $conn->prepare("SELECT COUNT(\*) as numOfProducts FROM products pr inner join pictures pic on pr.picture\_id = pic.id");

$prep->execute();

$numOfProducts = $prep->fetch();

$pagination = $numOfProducts->numOfProducts;

return ceil($pagination / $offset);

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

try{

$prep = $conn->prepare("SELECT COUNT(\*) as numOfProducts FROM products pr inner join pictures pic on pr.picture\_id = pic.id where pr.Category\_ID = ? AND name LIKE ?");

$prep->bindValue(1,"%$value%");

$prep->bindValue(2,$id);

$prep->execute();

$numOfProducts = $prep->fetch();

$pagination = $numOfProducts->numOfProducts;

return ceil($pagination / $offset);

}

catch(Exception $e){

return $e->getMessage();

}

endif;

}

function filterProducts($id = null ,$value = null,$limit){

global $conn;

$limit = (int)$limit \* PRODUCT\_OFFSET;

$offset = PRODUCT\_OFFSET;

if($id != null && $value == null):

try{

global $conn;

$prep = $conn->prepare("SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id where pr.Category\_ID = ? LIMIT ?,?");

$prep->bindValue(1,$id);

$prep->bindValue(2,$limit,PDO::PARAM\_INT);

$prep->bindValue(3,$offset,PDO::PARAM\_INT);

$prep->execute();

return $prep->fetchAll();

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

elseif($id == null && $value != null):

try{

$prep = $conn->prepare("SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id where name like ? limit ?,?");

$prep->bindValue(1,"%".$value."%");

$prep->bindValue(2,$limit,PDO::PARAM\_INT);

$prep->bindValue(3,$offset,PDO::PARAM\_INT);

$prep->execute();

return $prep->fetchAll();

}

catch(Exception $e){

return $e->getMessage();

}

elseif($id == null && $value == null):

return getAllProducts($limit);

else:

try{

$query = "SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id where pr.Category\_ID = ? AND name like ? LIMIT ?,?";

$prep = $conn->prepare($query);

$prep->bindValue(1,$id);

$prep->bindValue(2,"%".$value."%");

$prep->bindValue(3,$limit,PDO::PARAM\_INT);

$prep->bindValue(4,$offset,PDO::PARAM\_INT);

$prep->execute();

return $prep->fetchAll();

}

catch(Exception $e){

return $e->getMessage();

}

endif;

}

function sortPrice($value,$ids){

try{

global $conn;

if($ids != null):

$query = "SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id where (pr.id = ?";

if(count($ids) == 1)

$query.=")";

for($i = 1;$i<count($ids);$i++):

if($i+1 == count($ids)):

$query.="OR pr.id = ?)";

else:

$query .= " OR pr.id = ?";

endif;

endfor;

if($value == 1)

$query.= " ORDER BY price ASC";

else

$query.= " ORDER BY price DESC";

$prep = $conn->prepare($query);

for($i = 0; $i<count($ids);$i++):

$prep->bindValue($i+1,$ids[$i]);

endfor;

$prep->execute();

return $prep->fetchAll();

else:

if($value == 1)

return executeQuery("SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id ORDER BY price ASC");

else

return executeQuery("SELECT \*,pr.id FROM products pr inner join pictures pic on pr.picture\_id = pic.id ORDER BY price DESC");

endif;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function deleteProduct($id){

try{

global $conn;

$query = "delete from products where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$id]);

return "Successfully deleted product";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updateName($name,$id){

global $conn;

try{

$query = "update products set name = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$name,$id]);

return "Successfully updated name";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updateDescription($description,$id){

global $conn;

try{

$query = "update products set description = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$description,$id]);

return "Successfully updated description";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updatePrice($price,$id){

global $conn;

try{

$query = "update products set price = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$price,$id]);

return "Successfully updated price";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updateCategory($categoryId,$id){

global $conn;

try{

$query = "update products set Category\_ID = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$categoryId,$id]);

return "Successfully updated category";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updatePicture($file,$id){

$lastId = uploadPictureProduct($file);

if($lastId > 0):

try{

global $conn;

$queryUpdate = "update products set picture\_id = ? where id = ?";

$prepareUpdate = $conn->prepare($queryUpdate);

$prepareUpdate->execute([$lastId,$id]);

var\_dump($prepareUpdate);

return "Successfully updated picture";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

return "Picture could not be uploaded";

endif;

}

function uploadPictureProduct($file){

$tmpPath = $file["tmp\_name"];

$type = $file["type"];

if($type != "image/jpeg" && $type != "image/png")

return $message = "File must be jpg or png";

if($file["size"] > 10\*1024\*1024)

return $message = "File is too big,Max 10MB";

$fileName = $file["name"];

global $conn;

try{

$insertPic = "insert into pictures values(null,null,null,?)";

$prep = $conn->prepare($insertPic);

$prep->execute([$fileName]);

$lastId = $conn->lastInsertId();

$namePieces = explode(".",$fileName);

$extension = end($namePieces);

$originalName = $lastId . "." . $extension;

$uploadDir = "/assets/images/product\_images/";

$uploadOriginal = $uploadDir.$originalName;

if(move\_uploaded\_file($tmpPath,BASE\_URL.$uploadOriginal)):

$query = "update pictures set original = ? where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$uploadOriginal,$lastId]);

return $lastId;

else:

return 0;

endif;

}

catch(Exception $e){

return "There was a poblem with the database:".$e->getMessage();

}

}

function updateProduct($name,$description,$price,$categoryId,$picture,$productId){

$message = ["name"=>"","description"=>"","price"=>"","category"=>"","picture"=>""];

if(!empty($name))

$message ["name"] = updateName($name,$productId);

if(!empty($description))

$message ["description"] = updateDescription($description,$productId);

if(!empty($price))

$message ["price"] = updatePrice($price,$productId);

if(!empty($categoryId))

$message ["category"] = updateCategory($categoryId,$productId);

if(!empty($picture["name"]))

$message ["picture"] = updatePicture($picture,$productId);

return $message;

}

function exportExcel(){

$excel = new COM("Excel.Application");

$excel->Visible = 1;

$excel->DisplayAlerts = 1;

$workbook = $excel->Workbooks->Open("http://gadgetshopadvanced.000webhostapp.com/data/export/products.xls") or die("did not open");

$worksheet = $workbook->WorkSheets("products");

$worksheet->activate;

$products = getAllProducts();

$cellA = $worksheet->range("A1");

$cellA->activate;

$cellA->value = "Product ID";

$cellB = $worksheet->range("B1");

$cellB->activate;

$cellB->value = "Product Name";

$cellC = $worksheet->range("C1");

$cellC->activate;

$cellC->value = "Product Description";

$cellD = $worksheet->range("D1");

$cellD->activate;

$cellD->value = "Product Price";

$br = 2;

foreach($products as $product):

$cell = $worksheet->range("A$br");

$cell->activate;

$cell->value = $product->id;

$cell = $worksheet->range("B$br");

$cell->activate;

$cell->value = $product->name;

$cell = $worksheet->range("C$br");

$cell->activate;

$cell->value = $product->description;

$cell = $worksheet->range("D$br");

$cell->activate;

$cell->value = $product->price;

$br++;

endforeach;

$workbook->\_SaveAs("http://gadgetshopadvanced.000webhostapp.com/data/export/products.xls",-4143);

$workbook->Save();

$workbook->Save = true;

$workbook->Close;

$excel->Workbooks->Close();

$excel->Quit();

unset($worksheet);

unset($workbook);

unset($excel);

}

?>

models/products/getAllExcel.php

<?php

include "../../config/connection.php";

include "functionsProducts.php";

exportExcel();

header("location:../../index.php");

?>

models/products/getProducts.php

<?php

include "../../config/connection.php";

include "functionsProducts.php";

header("content-type:application/json");

if(isset($\_POST["limit"])):

$limit = $\_POST["limit"];

$products = getAllProducts($limit);

$pages = getPaginationNum();

echo json\_encode([

"products"=>$products,

"pages"=>$pages

]);

http\_response\_code(200);

else:

$products = getAllProducts();

echo json\_encode($products);

http\_response\_code(200);

endif;

?>

models/products/insertProduct.php

<?php

if(isset($\_POST["send"])):

include "../../config/connection.php";

include "../../models/categories/functionsCategories.php";

include "functionsProducts.php";

$name = $\_POST["productName"];

$description = $\_POST["productDesc"];

$price = $\_POST["productPrice"];

$categoryId = $\_POST["productCat"];

$file = $\_FILES["productPicture"];

$message = [];

$found = false;

$categories = getAllCategories();

foreach($categories as $category):

if($category->id == $categoryId)

$found = true;

endforeach;

if(!$found):

$message[0] = "Category id must be either:";

foreach($categories as $category):

$message[0] .= $category->id.",";

endforeach;

endif;

if($name == "")

$message [] = " name must be filled.";

if($description == "")

$message [] = " description must be filled.";

if($price == "")

$message [] = " price must be filled.";

if($file["name"] == "")

$message [] = " please upload product picture.";

if(count($message) > 0):

$message = implode(" ",$message);

logUserError("CategoryID",$message);

header("location:../../index.php?page=admin&message=$message");

else:

$lastInserted = uploadPictureProduct($file);

if($lastInserted > 0):

try{

$query = "insert into products values(null,?,?,?,?,?)";

$prepared = $conn->prepare($query);

$prepared->execute([$name,$description,$price,$categoryId,$lastInserted]);

$message = "successfully added new product";

header("location:../../index.php?page=admin&message=".$message);

}

catch(Exception $e){

$message = $e->getMessage();

logUserError("databese",$message);

header("location:../../index.php?page=admin&message=".$message);

}

else:

$message = "error in uploading file";

logUserError("file",$message);

header("location:../../index.php?page=admin&message=".$message);

endif;

endif;

else:

http\_response\_code(400);

endif;

?>

models/products/sortPrice.php

<?php

if(isset($\_POST["value"])):

header("content-type:application/json");

include "../../config/connection.php";

include "functionsProducts.php";

$value = $\_POST["value"];

if(isset($\_POST["ids"]))

$ids = $\_POST["ids"];

else

$ids = null;

$products = sortPrice($value,$ids);

echo json\_encode($products);

http\_response\_code(200);

else:

http\_response\_code(400);

endif;

?>

models/products/updateProduct.php

<?php

if(isset($\_POST["sendU"])):

include "functionsProducts.php";

include "../../config/connection.php";

$message = updateProduct($\_POST["name"],$\_POST["description"],$\_POST["price"],$\_POST["category"],$\_FILES["file"],$\_POST["idProduct"]);

$message = implode(" ",$message);

header("location:../../index.php?page=admin&message=".$message);

else:

http\_response\_code(400);

endif;

?>

models/user/deleteUser.php

<?php

if(isset($\_POST["id"])):

include "../../config/connection.php";

header("content-type:application/json");

$id = $\_POST["id"];

try{

$query = "delete from user\_pictures where id\_user = ?";

$prep = $conn->prepare($query);

$prep->execute([$id]);

$query2 = "delete from users where id = ?";

$prep2 = $conn->prepare($query2);

$prep2->execute([$id]);

http\_response\_code(204);

}

catch(PDOException $e){

echo json\_encode(["error"=>"There was a problem with the database:".$e->getMessage()]);

http\_response\_code(500);

}

else:

http\_response\_code(400);

endif;

?>

models/user/functionsUser.php

<?php

function getOne($id){

global $conn;

try{

$query = "SELECT \*,u.id FROM users u INNER JOIN pictures p on u.id\_profile\_pic = p.id WHERE u.id = ?";

$prep = $conn->prepare($query);

$prep->execute([$id]);

$user = $prep->fetch();

return $user;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function getAllUsers(){

return executeQuery("select \*,u.id from users u inner join pictures p on u.id\_profile\_pic = p.id inner join roles r on r.id = u.Role\_id");

}

function getPictures($id){

global $conn;

try{

$query = "SELECT \* FROM pictures p INNER JOIN user\_pictures up on p.id = up.id\_picture WHERE up.id\_user = ?";

$prep = $conn->prepare($query);

$prep->execute([$id]);

$user = $prep->fetchAll();

return $user;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updateProfilePicture($idUser,$idPicture){

global $conn;try{

$query = "update users set id\_profile\_pic = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$idPicture,$idUser]);

return $message = "Profile picture changed successfully";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function uploadPictureUser($file,$userId,$isProfile = false){

$tmpPath = $file["tmp\_name"];

$type = $file["type"];

list($width,$height) = getimagesize($tmpPath);

$newDimensions = 300;

if($width < $newDimensions || $height <$newDimensions)

return $message = "Picture cant be smaller than 300x300";

if($type != "image/jpeg" && $type != "image/png")

return $message = "File must be jpg or png";

if($file["size"] > 10\*1024\*1024)

return $message = "File is too big,Max 10MB";

$fileName = $file["name"];

global $conn;

try{

$insertPic = "insert into pictures values(null,null,null,?)";

$prep = $conn->prepare($insertPic);

$prep->execute([$fileName]);

$lastId = $conn->lastInsertId();

$namePieces = explode(".",$fileName);

$extension = end($namePieces);

$originalName = $lastId . "." . $extension;

$uploadDir = "/assets/images/user\_images/";

switch($type):

case "image/jpeg":

$emptyImage = imagecreatefromjpeg($tmpPath);

break;

case "image/png":

$emptyImage = imagecreatefrompng($tmpPath);

break;

endswitch;

$newImage = imagecreatetruecolor($newDimensions,$newDimensions);

imagecopyresampled($newImage,$emptyImage,0,0,0,0,$newDimensions,$newDimensions,$width,$height);

$uploadProfile = $uploadDir.$lastId."\_profile.".$extension;

switch($type):

case "image/jpeg":

$newImage = imagejpeg($newImage,BASE\_URL.$uploadProfile);

break;

case "image/png":

$newImage = imagepng($newImage,BASE\_URL.$uploadProfile);

break;

endswitch;

$uploadOriginal = $uploadDir.$originalName;

if(move\_uploaded\_file($tmpPath,BASE\_URL.$uploadOriginal)):

$query = "update pictures set original = ?,profile=? where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$uploadOriginal,$uploadProfile,$lastId]);

$updateUserPic = "insert into user\_pictures values(null,?,?)";

$result = $conn->prepare($updateUserPic);

$result->execute([$userId,$lastId]);

if($isProfile):

$updateProfilePic = "update users set id\_profile\_pic = ? where id = ?";

$prepPic = $conn->prepare($updateProfilePic);

$prepPic->execute([$lastId,$userId]);

else:

return "image successfully uploaded";

endif;

else:

return "Image could not be uploaded";

endif;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function adminUpdate($username,$email,$password,$confirmPass,$role,$id,$file){

$message = ["username"=>"","email"=>"","password"=>"","role"=>"","picture"=>""];

if(!empty($username))

$message ["username"] = updateUsername($username,$id);

if(!empty($email))

$message ["email"] = updateEmail($email,$id);

if(!empty($password) && !empty($confirmPass))

$message ["password"] = updatePassword($password,$confirmPass,$id);

if(!empty($role))

$message ["role"] = updateRole($role,$id);

if($file["name"] != "")

$message ["picture"] = uploadPictureUser($file,$id,true);

return $message;

}

function updateRole($role,$id){

global $conn;

try{

$query = "update users set Role\_ID = ? where id = ?";

$prep = $conn->prepare($query);

$prep->execute([$role,$id]);

return $message = "Role successfully updated";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

}

function updateUsername($username,$id){

global $conn;

$regUser = "/^([A-z]\s\*){2,15}/";

$message;

if(preg\_match($regUser,$username)):

try{

$query = "UPDATE users set username = ? where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$username,$id]);

return $message = "Username successfully updated";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

return $message = "Username must start with a letter,Min length 2,Max 15";

endif;

}

function updateEmail($email,$id){

global $conn;

$regEmail = "/^[a-zA-Z0-9.!#$%&'\*+\/=?^\_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)\*$/";

$message;

if(preg\_match($regEmail,$email)):

try{

$query = "UPDATE users set email = ? where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$email,$id]);

return $message = "Email successfully updated";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

return $message = "Email is not in good format";

endif;

}

function updatePassword($noviPass,$noviPass2,$id){

global $conn;

$greskaPass = "";

$regPass = "/^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)(?=.\*[@$!%\*?&])[A-Za-z\d@$!%\*?&]{8,15}$/";

$query = "select password from users where id = ?";

$prepare = $conn->prepare($query);

$prepare->execute([$id]);

$currentPassword = $prepare->fetch();

$newPasswordHash = md5($noviPass);

if(!preg\_match($regPass,$noviPass))

$greskaPass = "Password must have at least 1 uppercase, 1 lovercase letter,1 number and 1 special character,MIN length 8,MAX 15";

if($noviPass != $noviPass2)

$greskaPass = "Passwords do not match";

if($currentPassword == $newPasswordHash)

$greskaPass = "New password cant be the same as old password";

if($greskaPass == ""):

try{

$priprema = $conn->prepare("UPDATE users SET password = ? WHERE id = ?");

$priprema->execute([$newPasswordHash,$id]);

return $greskaPass = "Password successfully updated";

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

return $greskaPass;

endif;

}

function userRegister($username,$email,$password,$confirmPass,$role = 2,$picture = null){

global $conn;

$regUser = "/^([A-z]\s\*){2,15}/";

$regEmail = "/^[a-zA-Z0-9.!#$%&'\*+\/=?^\_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)\*$/";

$regPass = "/^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)(?=.\*[@$!%\*?&])[A-Za-z\d@$!%\*?&]{8,15}$/";

$greske = [];

if(!preg\_match($regUser,$username))

$greske[] = "Username must beggin with a letter,MIN 2,MAX 15 characters.";

if(!preg\_match($regEmail,$email))

$greske[] = "Email is not in good format.";

if(!preg\_match($regPass,$password))

$greske[] = "Password must have at least 1 uppercase, 1 lovercase letter,1 number and 1 special character,MIN length 8,MAX 15 characters.";

if($password != $confirmPass)

$greske[] = "password does not match.";

if($role == null)

$greske [] = "Please select a role.";

if($picture["name"] == "")

$picture = null;

if(count($greske) == 0):

try{

$password = md5($password);

$upitInsert = "INSERT INTO users(username,email,password,Role\_ID,id\_profile\_pic)

VALUES(?,?,?,?,?)";

$priprema = $conn->prepare($upitInsert);

$rezultat = $priprema->execute([$username,$email,$password,$role,26]);

if($picture == null):

return true;

else:

$lastInserted = $conn->lastInsertId();

return uploadPictureUser($picture,$lastInserted,true);

endif;

}

catch(Exception $e){

return "There was a problem with the database:".$e->getMessage();

}

else:

return $greske;

endif;

}

function exportWord(){

$word = new COM("Word.Application");

$word->Visible = 1;

$word->Documents->Add();

$word->Selection->TypeText(`

Full Name:Igor Jovanovic

Index Number:127/18

I was born in Belgrade on the 29th of October 1998. I graduated electro-technical high school "Nikola Tesla" profile: "Electro-Technician of Telecommunications".

I'm currently studding on the ICT College of applied studies, profile: "Web Programming",index number:127/18.

My goal for this website is to show the functions of PHP`);

$word->Documents[1]->SaveAs("http://mygadgetshop.epizy.com/php/index.php/data/export/aboutMe.doc");

$word->Quit;

unset($word);

}

?>

models/user/getMe.php

<?php

include "../../config/connection.php";

include "functionsUser.php";

exportWord();

header("location:../../index.php");

?>

models/user/getUsers.php

<?php

include "../../config/connection.php";

header("content-type:application/json");

$users = executeQuery("select \*,u.id from users u inner join pictures p on u.id\_profile\_pic = p.id inner join roles r on r.id = u.Role\_id");

echo json\_encode($users);

http\_response\_code(200);

?>

models/user/insertUser.php

<?php

if(isset($\_POST["send"])):

include "functionsUser.php";

include "../../config/connection.php";

$username = $\_POST["username"];

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

$password = $\_POST["password"];

$confirmPass = $\_POST["passwordC"];

if(isset($\_POST["role"]))

$roleID = $\_POST["role"];

else

$roleID = null;

$file = $\_FILES["fileUser"];

$register = userRegister($username,$email,$password,$confirmPass,$roleID,$file);

if($register === true):

$message = "successfully added new user";

header("location:../../index.php?page=admin&message=".$message);

endif;

if(is\_array($register)):

$message = implode(" ",$register);

header("location:../../index.php?page=admin&message=".$message);

else:

$message = $register;

header("location:../../index.php?page=admin&message=".$message);

endif;

else:

http\_response\_code(400);

endif;

?>

models/user/updateUser.php

<?php

session\_start();

include "../../config/connection.php";

include "functionsUser.php";

if(isset($\_POST["sendUsername"])):

$message = updateUsername($\_POST["editUsername"],$\_SESSION["account"]->id);

header("location:../../index.php?page=user&message=".$message);

elseif(isset($\_POST["sendEmail"])):

$message = updateEmail($\_POST["editEmail"],$\_SESSION["account"]->id);

header("location:../../index.php?page=user&message=".$message);

elseif(isset($\_POST["sendPassword"])):

$message = updatePassword($\_POST["editPass"],$\_POST["editPass2"],$\_SESSION["account"]->id);

header("location:../../index.php?page=user&message=".$message);

elseif(isset($\_POST["adminUpdate"])):

$message = adminUpdate($\_POST["username"],$\_POST["email"],$\_POST["password"],$\_POST["passwordC"],@ $\_POST["role"],$\_POST["idUser"],$\_FILES["fileUser"]);

$message = implode(" ",$message);

header("location:../../index.php?page=admin&message=".$message);

elseif(isset($\_POST["sendPicture"])):

if(!empty($\_FILES["file"]["name"])):

$message = uploadPictureUser($\_FILES["file"],$\_SESSION["account"]->id);

else:

$message = "Please select an image";

endif;

header("location:../../index.php?page=user&message=".$message);

elseif(isset($\_GET["idUser"]) && isset($\_GET["idPicture"])):

$message = updateProfilePicture($\_GET["idUser"],$\_GET["idPicture"]);

header("location:../../index.php?page=user&message=".$message);

else:

http\_response\_code(400);

endif;

?>

config/,env

SERVER=localhost

DATABASE=id9905572\_gadgetshopadvanced

USERNAME=id9905572\_gadgetshopadvanced

PASSWORD=predragovic98

config/config.php

<?php

DEFINE("BASE\_URL",$\_SERVER["DOCUMENT\_ROOT"]);

DEFINE("ENV", BASE\_URL."/config/.env");

DEFINE("VISITLOG", BASE\_URL."/data/logs/visitLog.txt");

DEFINE("ERRORLOG", BASE\_URL."/data/logs/errorLog.txt");

DEFINE("SERVER",env("SERVER"));

DEFINE("DATABASE",env("DATABASE"));

DEFINE("USERNAME",env("USERNAME"));

DEFINE("PASSWORD",env("PASSWORD"));

function env($name){

$file = file(ENV);

foreach($file as $element){

$row = explode("=",trim($element));

if($row[0] == $name)

return $row[1];

}

}

?>

config/connection.php

<?php

require\_once "config.php";

logSiteVisit();

try{

$conn = new PDO("mysql:host=".SERVER.";dbname=".DATABASE.";characterset=utf-8",USERNAME,PASSWORD);

$conn->setAttribute(PDO::ATTR\_DEFAULT\_FETCH\_MODE,PDO::FETCH\_OBJ);

$conn->setAttribute(PDO::ATTR\_ERRMODE,PDO::ERRMODE\_EXCEPTION);

}

catch(Exception $e){

echo $e->getMessage();

}

function executeQuery($query){

global $conn;

return $conn->query($query)->fetchAll();

}

function executeQueryOneRow($query){

global $conn;

return $conn->query($query)->fetch();

}

function logSiteVisit(){

$open = fopen(VISITLOG, "a");

if($open){

fwrite($open, "{$\_SERVER['REQUEST\_URI']}\t{$\_SERVER['REQUEST\_TIME']}\t{$\_SERVER['REMOTE\_ADDR']}\n");

fclose($open);

}

}

function logUserError($error,$message){

$open = fopen(ERRORLOG, "a");

if($open){

fwrite($open,"$error:\t$message\n");

fclose($open);

}

}

?>

assets/js/index.js

$(document).ready(function(){

//!REGISTER!

$("#forma").submit(function(e){

e.preventDefault();

provera();

})

//!REGISTER!

//--PRODUCTS--

$(document).on("click",".page",function(e){

e.preventDefault();

let filter =$("#filter").prop("value")

let limit = $(this).data("limit");

let value = $("#sortName").val();

if(value == "")

value = null;

console.log(filter)

if(filter > 0)

paginationAJAX(limit,"models/products/filterCategory.php",filter,value)

else

paginationAJAX(limit,"models/products/filterCategory.php",null,value)

})

$(document).on("click",".prev",function(e){

e.preventDefault()

let limit = Number($(".page.active").data("limit")) - 1;

let filter =$("#filter").prop("value")

if(filter > 0)

paginationAJAX(limit,"models/products/filterCategory.php",filter)

else

paginationAJAX(limit,"models/products/getProducts.php")

})

$(document).on("click",".next",function(e){

e.preventDefault()

let filter =$("#filter").prop("value")

let limit = $(".page.active").data("limit") + 1;

if(filter > 0)

paginationAJAX(limit,"models/products/filterCategory.php",filter)

else

paginationAJAX(limit,"models/products/getProducts.php")

})

$("#filter").change(function(){

let id = $(this).prop("value");

let limit = 0

paginationAJAX(limit,"models/products/filterCategory.php",id);

})

$("#sortPrice").change(function(){

let value = $(this).prop("value");

let products = $(".product");

let productIds = []

for(let product of products){

productIds.push(product.querySelector(".id").value)

}

$.ajax({

url:"models/products/sortPrice.php",

method:"post",

type:"json",

data:{value:value,ids:productIds},

success : function(products){

filterProducts(products)

},

error: showAJAXErrors

})

})

$("#sortName").keyup(function(){

let value = $(this).val()

let productIds = $("#filter").prop("value");

paginationAJAX(0,"models/products/filterName.php",productIds,value)

})

//--PRODUCTS--

//!USERUPDATE!

$("#editUsername").click(function(e){

e.preventDefault()

$("#formUsername").toggle();

})

$("#editPass").click(function(e){

e.preventDefault()

$("#formPass").toggle();

})

$("#editEmail").click(function(e){

e.preventDefault()

$("#formEmail").toggle();

})

$("#uploadImage").click(function(e){

e.preventDefault()

$("#formPicture").toggle()

})

//--USEREDIT--

//--ADMIN PANEL--

$(document).on("click",".update",function(e){

e.preventDefault();

$(".updateForma").css("display","inline-block")

id = this.dataset.id;

$("#idProduct").val(id)

$('html, body').animate({

scrollTop: $(".updateForma").offset().top

}, 1000);

})

$(document).on("click",".updateU",function(e){

e.preventDefault();

$(".updateFormaU").css("display","inline-block")

id = this.dataset.id;

$("#idUser").val(id)

$('html, body').animate({

scrollTop: $(".updateFormaU").offset().top

}, 1000);

})

$(document).on("click",".delete",function(e){

e.preventDefault();

let id = $(this).data("id");

$.ajax({

url:"models/products/deleteProduct.php",

method:"post",

type:"json",

data:{id:id},

success:function(){

refreshProductsTable();

},

error:showAJAXErrors

})

})

$(document).on("click",".deleteU",function(e){

e.preventDefault();

let id = $(this).data("id");

$.ajax({

url:"models/user/deleteUser.php",

method:"post",

type:"json",

data:{id:id},

success:function(){

refreshUsersTable();

},

error:showAJAXErrors

})

})

// --ADMIN PANEL--

})

function paginationAJAX(limit,url,id = null,value = null){

$.ajax({

url:url,

method:"post",

type:"json",

data:{id:id,limit:limit,value:value},

success : function(data){

console.log(data);

filterProducts(data.products)

printPagination(data.pages,limit)

},

error: showAJAXErrors

})

}

function filterProducts(products){

let html = ``

for(let product of products){

html += `<li>

<div class = product>

<a href = "index.php?page=products&id=${product.id}"><img src = "${product.original}"/></a>

<p class="name">${product.name}</p><hr>

<p class ="price">$${product.price}</p>

<input type = "hidden" value = "${product.id}" class = "id"/>

<input type = "hidden" value = "${product.Category\_ID}" class = "idC"/>

<input type = "button" value = "Add to cart"/>

</div>

</li>`

}

$("#products").html(html);

}

function printPagination(numberOfProducts,active){

let html = ``

if(active > 0)

html += `<button><a class = "prev" href = "#">PREVIOUS</a></button>`

for(let i = 1;i<numberOfProducts;i++){

if(i == active)

html +=`<a href = "#" class = "page active" data-limit = "${i}">${i+1}</a>`

else

html +=`<a href = "#" class = "page" data-limit = "${i}">${i+1}</a>`

}

if(active != numberOfProducts-1)

html += `<button><a class = "next" href = "#">NEXT</a></button>`

$(".pages").html(html);

}

function refreshProductsTable(){

$.ajax({

url:"models/products/getProducts.php",

method:"get",

type:"json",

success:function(products){

fillProductsTable(products)

},

error:showAJAXErrors

})

}

function refreshUsersTable(){

$.ajax({

url:"models/user/getUsers.php",

method:"get",

type:"json",

success:function(users){

fillUsersTable(users)

},

error:showAJAXErrors

})

}

function fillProductsTable(products){

let html = ``

for(let product of products){

html += `<tr>

<td>${product.id}</td>

<td>${product.name}</td>

<td>${product.description}</td>

<td>${product.price}</td>

<td><img src = "${product.original}"/></td>

<td>${product.Category\_ID}</td>

<td>

<button><a data-id = "${product.id}" class = "delete" name = "deleteU" href = "" >DELETE PRODUCT</a></button>

<button><a data-id = "${product.id}" class = "update" name = "update" href = "#" >UPDATE PRODUCT</a></button>

</td>

</tr>`

}

$("#products").html(html);

}

function fillUsersTable(users){

let html = ``

for(let user of users){

html += `<tr>

<td>${user.id}</td>

<td>${user.username}</td>

<td>${user.email}</td>

<td><img src = "${user.profile}"/></td>

<td>${user.name}</td>

<td>

<button><a data-id = "${user.id}" class = "deleteU" name = "deleteU" href = "" >DELETE USER</a></button>

<button><a data-id = "${user.id}" class = "updateU" name = "update" href = "#" >UPDATE USER</a></button>

</td>

</tr>`

}

$("#users").html(html);

}

function showAJAXErrors(error, status, statusText){

console.error('AJAX ERROR: ');

console.log(status);

console.log(statusText);

if(error.status == 500){

console.log(error.parseJSON);

}

else if(error.status == 400){

alert('Parameters were not sent successfully')

}

}

let postoji = false;

function provera()

{

let username = document.getElementById("usernameReg").value;

let email = document.getElementById("email").value;

let password = document.getElementById("passwordReg").value;

let Cpassword = document.getElementById("Cpassword").value;

let regUser = /^([A-z]\s\*){2,15}/;

let regEmail = /^[a-zA-Z0-9.!#$%&'\*+/=?^\_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)\*$/;

let regPass = /^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)(?=.\*[@$!%\*?&])[A-Za-z\d@$!%\*?&]{8,15}$/;

let greske = [];

let forma = document.getElementById("forma")

if(!regUser.test(username))

greske.push("Username must beggin with a letter,MIN 2,MAX 15 characters")

if(!regPass.test(password))

greske.push("Password must have at least 1 uppercase, 1 lovercase letter,1 number and 1 special character,MIN length 8,MAX 15")

if(password != Cpassword)

greske.push("password does not match")

if(!regEmail.test(email))

greske.push("Email is not in good format")

if(greske.length){

if(!postoji){

$('<span id ="zagreske"></span>').appendTo(forma);

let spanZaGreske = document.getElementById("zagreske")

postoji = true;

for(i in greske){

spanZaGreske.innerHTML += greske[i] + "</br>";

}

}

else{

let spanZaGreske = document.getElementById("zagreske")

spanZaGreske.innerHTML = "";

for(i in greske){

spanZaGreske.innerHTML += greske[i] + "</br>";

}

}

return false;

}

else

{

$('#zagreske').remove();

$.ajax({

url : "models/login/register.php",

method : "post",

type : "json",

data : {

username : username,

email : email,

password : password,

Cpassword : Cpassword

},

success : function(data){

alert("Successfully registered!")

return true;

},

error : showAJAXErrors

})

}

}

assets/css/style.css

@media only screen and (min-width:100px) and (max-width:699px)

{

#header{

width: 100%;

height: 100%;

}

#header div div#figure{

display: none;

}

#header div:first-child div#logo {

width: 50%;

}

#header div{

width: 100%;

}

#body{

width: 100%;

}

#footer{

width: 100%;

}

div#footer div div{

width: 100%;

}

div#footer div div:first-child{

background: none;

width: 100%;

}

div#header div:first-child div#logo a img{

display: none;

}

#ja{

width: 100%;

}

}

@media only screen and (min-width:700px) and (max-width:1200px)

{

#header{

width: 100%;

height: 100%;

}

#header div div#figure{

display: block;

}

#body{

width: 100%;

}

#footer{

width: 100%;

}

}

#body div div div div div#pages{

text-align: center;

font-size: 30px;

}

#body div div div div div.filter-categories{

width: 30%;

float: left;

padding: 0px;

}

#body div div div div div.sort-price{

width: 20%;

float: right;

padding: 0px;

}

#body div div div div div.sort-name{

width: 20%;

float: right;

padding: 0px;

}

.productImg{

width: 50%;

}

.info{

font-size: 20px;

display: block;

margin: 2% 0 2%;

}

.a-info{

color:#00dfdc;

font-size: 20px;

margin: 2% 2% 2%;

float: left;

}

.a-infoR{

color:#00dfdc;

font-size: 20px;

margin: 2% 6% 2%;

float: right

}

.register input.ne{

width: 7%;

}

.registerP{

float: left;

padding: 10px 10px 10px;

width: 45%;

}

.registerU{

float: right;

padding: 10px 10px 10px;

width: 45%;

}

.spanGreske{

font-size: 30px;

width: 20%;

position: relative;

margin-top: 1%;

color:#820048;

}

.error{

color: red;

font-size: 20px;

}

#message{

font-size: 20px;

color:#820048;

}

#profilePic{

padding-bottom: 30px;

}

.profilePicture{

padding-left:5px;

padding-right: 5px;

}

#body div div div div div.product{

border: #4f0428;

border-style: solid;

border-width: 2px;

text-align: center;

font-size: 12px;

padding-bottom: 5px;

}

.pText{

color:#820048;

font-size: 20px;

padding-top: 5px;

padding-bottom: 5px;

}

.spanText{

color:#820048;

}

.shopNow{

position: relative;

margin-left: auto;

margin-right: auto;

display: block;

}

.page{

font-size: 26px;

padding: 3px 16px;

color:#820048;

text-decoration: none;

transition:backgroun-color .1s;

}

.page.active{

background-color: crimson;

color: white;

text-decoration: none;

transition:backgroun-color .1s;

}

.signUp {

color: mediumturquoise;

font-size: 20px;

}

#body div div div div div .updateForma{

display: none;

width: 43%;

padding-top: 2%;

float: right

}

#body div div div div div .updateFormaU{

display: none;

width: 43%;

padding-top: 4%;

float: left;

}

#body div div div div div .userInfo{

font-size: 20px;

}

p .editLink{

float: right;

}

#body div div div div div div#sidebar2 {

background: none;

float: right;

padding: 0;

width: 100%;

margin-top: 27px;

}

.editForme{

display: none;

margin-top: 24px;

}

#body div div div div div p .notice{

margin: 0 0 30px;

}

.slider{

width: 75%;

}

.tabela img {

height: 50px;

width: 50px;

}

.tabela2{

font-size: 13px;

}

.product img

{

width: 100%;

height: 100%;

}

#zagreske{

display: flex;

color: red;

font-size: 15px;

}

.logout{

font-size: 15px;

color: mediumturquoise;

}

.login input {

width: 30%;

padding: 6px;

border-width: 5px;

}

.register input {

width: 30%;

padding: 10px;

border-width: 5px;

}

.register input .ne{

width: 0%;

padding: 10px;

border-width: 5px;

}

.registerU input {

width: 30%;

padding: 10px;

border-width: 5px;

}

.registerU input .ne{

width: 0%;

padding: 10px;

border-width: 5px;

}

.registerP input {

width: 30%;

padding: 10px;

border-width: 5px;

}

.registerP input .ne{

width: 0%;

padding: 10px;

border-width: 5px;

}

#body div div div div div p.name{

color: #0a5382;

white-space: normal;

}

.price{

color: #820048;

}

.anketa{

text-align: center;

}

#kupuj{

margin-left: 48%;

}

body {

background: url(../images/bg-body.jpg) repeat left top;

color: #515151;

font-family: Arial, Helvetica, sans-serif;

font-size: 14px;

}

a {

outline: none;

}

p a {

color: #515151;

}

#body div div div div div div#sidebar ul li a {

color: #515151;

text-decoration: none;

}

#body div div div div div div#sidebar ul li a:hover {

text-decoration: underline;

}

p a:hover {

color: #233323;

}

#header {

background: url(../images/pinklight.png) repeat-x left top;

}

#header div:first-child {

border: none;

height: 100%;

z-index: 2;

}

#header div:first-child div#logo {

background: none;

border: none;

display: block;

position: relative;

padding: 0;

width: 11%;

margin-left: auto;

margin-right: auto;

z-index: 1000;

}

#header div:first-child div#logo a img {

display: inline-block;

height: 75%;

}

#header div:first-child ul {

height: 43px;

list-style: none;

left: 0;

margin: 0;

padding: 0;

text-align: center;

top: 90px;

width: 100%;

margin-top: 2%;

}

#header div:first-child ul li {

display: inline-block;

}

#header div:first-child ul li.home {

left: 56px;

top: 0;

}

#header div:first-child ul li.logoutli {

left: 900px;

top: 0;

}

#header div:first-child ul li.products {

left: 194px;

top: 0;

}

#header div:first-child ul li.about {

left: 630px;

top: 0;

}

#header div:first-child ul li.blog {

left: 740px;

top: 0;

}

#header div:first-child ul li a {

display: block;

color: #FFFFFF;

cursor: pointer;

font-weight: bold;

height: 43px;

letter-spacing: 1px;

line-height: 33px;

padding: 0 0 0 13px;

text-decoration: none;

text-transform: uppercase;

text-shadow: 0px -2px 0px #4f0428;

}

#header div:first-child ul li.current a {

background: url(../images/bg-menu-left.png) no-repeat left top;

color: #FFFFFF;

}

#header div:first-child ul li.current a span {

background: url(../images/bg-menu-right.png) no-repeat right top;

display: block;

height: 43px;

padding: 0 15px 0 5px;

text-align: center;

}

#header div {

height: auto;

width: 100%;

z-index: 1;

}

#header div div#figure {

position: relative;

}

#header div div#figure div {

background: none;

text-align: center;

top: 143px;

width: 100%;

}

#header div div#figure div h1 {

color: #00dfdc;

display: block;

font-size: 55px;

font-weight: normal;

margin: 0;

text-transform: capitalize;

}

#header div div#figure div h2 {

color: #c4c4c4;

display: block;

font-size: 22px;

font-weight: normal;

line-height: 45px;

margin: 0;

text-transform: uppercase;

}

#header div span#background {

background: url(../images/bg-blue.png) no-repeat center top;

display: block;

height: 101px;

}

#body div {

background: url(../images/bg-content2.png) repeat-x left top;

border-bottom: 2px solid #86104a;

padding: 0;

width: 100%;

}

#body div div {

background: url(../images/bg-content.png) no-repeat center top;

border: 0;

padding: 104px 0 0;

width: 100%;

}

#body div div div {

background: url(../images/bg-content3.jpg) repeat left 108px;

border: 0;

margin: 0;

padding: 0;

}

#body div div div div {

padding: 0 0 40px;

width: 100%;

}

#body div div div div h1 {

color: #0a5382;

font-family: "Times New Roman", Times, serif;

font-size: 38px;

font-weight: normal;

letter-spacing: 2px;

margin: 0;

text-align: center;

text-transform: uppercase;

}

#body div div div div h4 {

color: #3f3f3f;

font-weight: normal;

margin: 0 0 50px;

text-align: center;

}

#body div div div div ul {

list-style: none;

margin: 0;

overflow: hidden;

padding: 0;

text-align: center;

}

#body div div div div ul li {

float: left;

padding: 2% 2% 2%;

width: 29%;

}

#body div div div div ul li a img {

display: block;

filter: alpha(opacity=100);

opacity: 1;

}

.select-css {

display: block;

font-size: 16px;

font-family: sans-serif;

font-weight: 700;

color: #444;

line-height: 1.3;

padding: .6em 1.4em .5em .8em;

width: 100%;

box-sizing: border-box;

margin: 0;

border: 1px solid #aaa;

box-shadow: 0 1px 0 1px rgba(0,0,0,.04);

border-radius: .5em;

-moz-appearance: none;

-webkit-appearance: none;

appearance: none;

background-color: #fff;

background-image: url('data:image/svg+xml;charset=US-ASCII,%3Csvg%20xmlns%3D%22http%3A%2F%2Fwww.w3.org%2F2000%2Fsvg%22%20width%3D%22292.4%22%20height%3D%22292.4%22%3E%3Cpath%20fill%3D%22%23007CB2%22%20d%3D%22M287%2069.4a17.6%2017.6%200%200%200-13-5.4H18.4c-5%200-9.3%201.8-12.9%205.4A17.6%2017.6%200%200%200%200%2082.2c0%205%201.8%209.3%205.4%2012.9l128%20127.9c3.6%203.6%207.8%205.4%2012.8%205.4s9.2-1.8%2012.8-5.4L287%2095c3.5-3.5%205.4-7.8%205.4-12.8%200-5-1.9-9.2-5.5-12.8z%22%2F%3E%3C%2Fsvg%3E'),

linear-gradient(to bottom, #ffffff 0%,#e5e5e5 100%);

background-repeat: no-repeat, repeat;

background-position: right .7em top 50%, 0 0;

background-size: .65em auto, 100%;

}

.select-css::-ms-expand {

display: none;

}

.select-css-name {

display: block;

font-size: 16px;

font-family: sans-serif;

font-weight: 700;

color: #444;

line-height: 1.3;

padding: .6em 1.4em .5em .8em;

width: 100%;

box-sizing: border-box;

margin: 0;

border: 1px solid #aaa;

box-shadow: 0 1px 0 1px rgba(0,0,0,.04);

border-radius: .5em;

-moz-appearance: none;

-webkit-appearance: none;

appearance: none;

}

#body div div div div ul li a img:hover {

filter: alpha(opacity=90);

opacity: 0.9;

}

#body div div div div ul li h2 {

font-size: 16px;

font-weight: normal;

margin: 0;

}

#body div div div div ul li h2 a {

background: url(../images/bg-prodname.png) repeat-x;

color: #FFF;

display: block;

height: 57px;

letter-spacing: 0.05em;

line-height: 51px;

text-align: center;

text-decoration: none;

text-transform: uppercase;

}

#body div div div div ul li h2 a:hover {

background: url(../images/bg-prodname-hover.png) repeat-x;

}

#body div div div div div {

background: none;

overflow: hidden;

padding: 0 0 56px;

width: 100%;

text-align: center;

}

#body div div div div div h3 {

color: #056f76;

font-size: 16px;

line-height: 30px;

margin: 0;

text-transform: uppercase;

}

#body div div div div div p {

line-height: 30px;

margin: 0 25% 0;

width: 50%;

}

#body div div div div div div#article {

background: none;

float: left;

padding: 0;

width: 100%;

}

#body div div div div div div#sidebar {

overflow-x:auto;

background: none;

float: left;

padding: 0;

width: 100%;

margin: 0 auto;

}

#body div div div div div div#sidebar h4 {

color: #056f76;

font-weight: bold;

line-height: 30px;

margin: 0;

text-align: left;

text-transform: uppercase;

}

#body div div div div div div#sidebar ul {

list-style: disc;

margin: 0 0 30px;

padding: 0 0 0 15px;

text-align: left;

}

#body div div div div div div#sidebar ul li {

float: none;

line-height: 30px;

margin: 0;

padding: 0;

width: auto;

}

#footer {

min-height: 413px;

}

#footer div {

margin: 0 auto;

overflow: hidden;

padding: 28px 0 0;

width: 100%;

}

#footer div div:first-child {

background: url(../images/footer-separator.png) no-repeat right top;

float: left;

width: 80%;

}

#footer div div:first-child p {

color: #f0b7d2;

line-height: 30px;

margin: 2% 0 2%;

width: 90%;

text-align: justify;

}

#footer div div p a {

color: #f0b7d2;

}

#footer div div h3 {

color: #FFFFFF;

font-size: 20px;

font-weight: normal;

margin: 0 0 30px;

padding: 50px 0 0;

text-shadow: 0 1px #820048;

text-transform: uppercase;

}

#footer div div {

float: right;

min-height: 299px;

padding: 0;

width: 16%;

}

#footer div div ul {

list-style: none;

margin: 0;

padding: 0;

}

#footer div div ul li {

display: block;

margin: 20px 0 0;

}

#footer div div ul li a {

color: #f0b7d2;

display: block;

height: 22px;

line-height: 22px;

text-decoration: none;

text-indent: 40px;

text-transform: lowercase;

}

#footer div div ul li a:hover, #footer div div p a:hover {

color: #FFFFFF;

}

#footer div div ul li:first-child {

margin: 0;

}

#footer div div ul li a#googleplus {

background: url(../images/icons.png) no-repeat 3px -104px;

}

#footer div div ul li a#googleplus:hover {

background: url(../images/icons.png) no-repeat 3px -126px;

}

#footer div div ul li a#twitter {

background: url(../images/icons.png) no-repeat 0 -54px;

height: 20px;

line-height: 20px;

}

#footer div div ul li a#twitter:hover {

background: url(../images/icons.png) no-repeat 0 -74px;

}

#footer div div ul li a#facebook {

background: url(../images/icons.png) no-repeat 8px 0;

}

#footer div div ul li a#facebook:hover {

background: url(../images/icons.png) no-repeat 8px -22px;

}

#footer p.footnote {

background: #991456;

border-radius: 5px;

color: #c14381;

height: 30px;

font-size: 12px;

line-height: 30px;

margin: 30px auto 0;

text-align: center;

width: 260px;

}