**Univerzitet u Sarajevu**

**Elektrotehnički faklutet**

**Dokumentacija**

**WT projekat**

**Student: Kenan Mahmutović**

**Index: 16009**

**Sarajevo, Septembar 2015.**

**1.SPIRALA**

* Datoteka index.html





Index.html datoteka sadrži osnovnu strukturu početne stranice. Važno je dobro rasporediti elemente da bi svaka promjena nad stranicom bila jednostavna i preglednija. U <head> tagu se nalazi style.css fajl u kojem se čuvaju podešavanja o vizuelnom izgledu stranice. Meni se sastoji iz neuređene liste, pri čemu svaki član te liste predstavlja jedan link koji vodi do određene stranice. Login forma prestavlja formu preko koje će se korisnik moći ulogovati na stranicu. Kod *<a>* tagova koristimo *href* atribut da postavimo link na neki objekat.

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

<title>Auto-Skola Iris</title>

<link href="style.css" media="screen" rel="stylesheet" />

</head>

U head dijelu nalazi se ime web stranice koje se prikauzje na tabu u pretrazivacu, i tu se obicno referencira .css dokument koji cuva pravila o vizulenom izgledu stranice.

<div id="main">

Div main predstavlja kontejner u koji ce se u daljem razvoju stranice smjestati sve stvari koje nisu staticne.

* Datoteka aboutUs.html

<div id="content">

<!-- your text goes here -->

<div id="main">

<div>

<h2 class="header">O nama</h2>

<div class="O nama">

<p>Dobrodošli u Auto školu "Iris", jednu od vodećih firmi u regionu koje se bave obukom kandidata za upravljanje motornim vozilima. Od osnivanja firme, pa do danas, nastojali smo da stalno poboljšavamo kvalitet obuke za sve koji odluče da nam ukažu povjerenje, i sa ponosom možemo da kažemo da uspješno poslujemo već 20 godina, te da smo samo stalnim radom i trudom uspjeli da stvorimo ugled firme.

</p>

<p>Auto škola "Iris" svojim kandidatima nudi vrhunsku obuku uz prijatnu atmosferu i moderno okruženje. Kandidati se osposobljavaju za sljedeće kategorije: AM,A1,A2,A,B1,B,BE,C1,C1E,C,CE</p>

</div>

</div>

</div>

<div style="clear:both;"></div>

</div>

Ova datoteka je identicna kao i index.html, jedino se razlikuje po sadrzaju *main* div-a, te nije potrebno dalje objasnjavanje.

* Datoteka kalendar.html

<div id="content">

<!-- your text goes here -->

<div id="main\_kalendar">

<div>

<h2>Tabela događaja</h2>

</div>

<div class="tabela">

<table id="tabela" style="width:100%">

<tr>

<th>Događaj</th>

<th>Datum</th>

<th>Vrijeme</th>

</tr>

<tr>

<td>Predavanje</td>

<td>21.03.2015.</td>

<td>09:00h</td>

</tr>

<tr>

<td>Predavanje</td>

<td>26.03.2015.</td>

<td>09:00h</td>

</tr>

<tr>

<td>Teorijski ispit</td>

<td>27.03.2015.</td>

<td>09:00h</td>

</tr>

<tr>

<td>Praktični ispit</td>

<td>03.04.2015.</td>

<td>10:00h</td>

</tr>

</table>

</div>

</div>

</div>

Div *content* sada sadrži div *main\_kalendar* koji sadrži tabelu koju sam napravio pomoću taga *<table>*. Redove tabele sam prikazao pomoću taga *<tr>*, a sadržaj svakog reda pomoću taga *<th>*. Sve ostalo je isto na i na index.html stranici.

* Datoteka kontakt.html

<div id="content">

<!-- your text goes here -->

<div id="main">

<div class="informacije">

<h1> Informacije</h1>

<p>Adresa: Ul. Rifata Budzevica, Sarajevo 71000</p>

<p>Tel/fax: +387 33 288 277</p>

<p>Mob: +387 66 899 254</p>

<p>E-mail: </p>

</div>

<div class="kontaktforma">

<p>Ime:</p>

<input type="text" name="ime" class="input1" />

<p>Prezime:</p>

<input type="text" name="prezime" class="input1" />

<p>E-mail:</p>

<input type="text" name="email" class="input1" />

<p>Poruka:</p>

<textarea rows="4" cols="50" class="input1"></textarea>

<br />

<input type="submit" value="Posalji" />

<input type="reset" value="Poništi" />

</div>

</div>

</div>

Div *content* sada sadrži dva diva *informacije* i *kontaktforma*. Div *informacije* sadrži osnovne podatke o firmi, dok *kontaktforma* sadrži polja koja služe za unos podataka od strane korisnika. Ta polja sam ubacio koristeći tag *<input>*. Atribut *type* nam služi da postavimo koji tip vijednosti će to polje da prima, npr. broj, tekst itd.

Ostalo je identično.

* Datoteka novosti.html

<div id="content">

<!-- your text goes here -->

<div id="main">

<div class="novosti">

<div>

<h2 class="header">Novosti</h2>

<div class="datum\_objave">

<p class="header">Datum objave: 17.03.2015.</p>

</div>

Div *content* sada sadrži novi div koji ima atribut *class* sa vrijednošću “novosti”, atribut *class* nam služi da u .css fajlu odredimo pravila za svaki element koji ima istu vrijednost tog atributa.

<div id="novosti">

<h3>Pomjeranje termina</h3>

<p>

Termin za polaganje teorijskog dijela vozačkog ispita zakazan za 20.03.2015. godine se pomjera za sedam dana, odnosno za 27.03.2015. godine. Polaganje počinje u 09:00h, mole se svi kandidati da budu ispred sale za polaganje barem 15 minuta prije početka ispita. <span class="vazno">Ne zaboravite da ponesete ličnu kartu sa sobom. Kandidati koji ne budu imali ličnu kartu sa sobom neće moći pristupiti polaganju teorijskog ispita.</span>

</p>

</div>

</div>

</div>

<div style="clear:both;"></div>

</div>

Div sa id novosti sadrži jednu novost, odnosno tekst unesen u element *<p>*. Ostalo je isto.

* Datotek style.css

Ova datoteka kao što je već spomenuto sadrži pravila za vizuelni prikaz html elemenata. Pravilo za klasu pravimo tako što ispred imena klase stavimo “.”, dok pravilo za id pravimo tako što ispred pravila stavimo znak “#”.

Ispod će biti prikazano nekoliko pravila, obzirom da ova datoteka ima blizu 300 redova pa nije praktično da se kopira cijela.

body {

background-color: #fff;

padding: 0px;

margin: 0px;

font-family: Trebuchet MS, Helvetica, sans-serif;

font-size: 14px;

line-height: 24px;

color: #494949;

background-image: url(*images/body-bg.jpg*);

background-repeat: repeat-x;

background-color: cadetblue;

}

Ovo pravilo odnosi se na <body> deo html stranice. Idući od vrha prema dole, imamo atribute za pozadinu, padding, margine, font, veličinu fonta, boju, pozadinsku sliku.

a:hover {

text-decoration: none;

}

Ovo pravilo se odnosi na tag *<a>*, a izvrši se kada se pređe mišem preko tog taga.

.partneri h3 {

color: cadetblue;

font-size: 25px;

}

Ovo pravilo se odnosi na tag *<h3>* koji se nalazi unutar elementa sa klasom “partneri”. Ovo pravilo postavlja boju i veličinu slova.

#menu ul li {

list-style: none;

float: left;

clear: none;

margin-left: 1px;

margin-top: 5px;

}

Ovo pravilo se odnosi na tag *<li>* koji se nalazi unutar taga *<ul>* a koji se nalazi unutar elementa sa id=“menu“.

Sva ostala pravila su u biti slična i mijenjanju samo izgled određenih dijelova stranice.

2.SPIRALA

* Datoteka index.html

<li onmouseover="showMenu();" onmouseout="hideMenu();"><a>Uputstva</a>

<div id="drop\_down">

<a>Teorijski ispit</a>

<a>Prakticni ispit</a>

<a>Prva pomoc</a>

</div>

</li>

Ovaj dio koda se odnosi na novi tab u meniju „Uputstva“ koji sadrži u sebi meni, koji se prikaže kada se pređe mišem preko ovog linka.

Takvo ponašanje sam ostvario pomoću funckije *showMenu()* koja se aktivira kada se pređe mišem preko linka, odnosno kada okine event *onmouseover*.

Meni se sakrije okidanjem eventa *onomuseover* kada miš napusti link. Meni sam napravio tako što sam u jedan div stavio tri taga *<a>* i poredao ih vertikalno. Div je po defaultu sakriven, ali pozivom funkcije *onmouseover* mijenja se njegov css atribut *visibility* i postavlja se na *visible*, što kao posljedicu ima prikazivanje ovog diva. Za sakrivanje diva iskoristena je ista metoda ali je atribut *visibility* postavljen na *hidden*.

Sadržaj datoteke showMenu.js:

var item = document.getElementById('drop\_down');

var showMenu = function ()

{

item.style.visibility = 'visible';

}

var hideMenu = function ()

{

item.style.visibility = 'hidden';

}

Document.getElementById nam daje referencu na neki element, koju sam smjestio u varijablu item. showMenu i hideMenu su funkcije koja koriste tu referencu i postavljaju joj atribut na vrijednosti „visible“ i „hidden“ respektivno.

Datoteku showMenu.js referenciramo tako što dodamo u footer sljedeći kod: <script src="showMenu.js"></script>.

Zadatak vezan za validaciju forme sam uradio tako što sam napravio funkciju validateForm čija se implementacija nalazi ispod.

function validateForm() {

var ime = document.kontaktforma.ime;

var prezime = document.kontaktforma.prezime;

var email = document.kontaktforma.email;

var poruka = document.kontaktforma.poruka;

var pattern = /[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+(?:\.[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+)\*@(?:[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?/i;

if (ime.value == null || ime.value == "") {

document.getElementById('imeError').style.visibility = 'visible';

return false;

} else {

document.getElementById('imeError').style.visibility = 'hidden';

}

if (prezime.value == null || prezime.value == "") {

document.getElementById('prezimeError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('prezimeError').style.visibility = 'hidden';

}

if (!pattern.test(email.value) || email.value == "") {

document.getElementById('emailError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('emailError').style.visibility = 'hidden';

}

if (poruka.value == "") {

document.getElementById('porukaError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('porukaError').style.visibility = 'hidden';

}

}

Ova funkcija preuzima parametre iz forme sa stranice, i smješa ih u varijable ime , prezime, email, poruka.

Svaka varijabla se provjerava da li je prazna i u slučaju da jeste, skripta prekida izvršavanje tako što vrati false.

Pattern predstavlja regex pattern koji sam koristio da provjerim da li je mejl u validnom formatu.

U slučaju da neka varijabla nije prošla validaciju, to se odrazilo na glavnom dokumentu tako što sam prikazao sličicu sa greškom pored polja koje nije prošlo validaciju. Sliku sam prikazao tako što sam promijenio njen atribut visibility. To se vidi u ovoj liniji

document.getElementById('imeError').style.visibility = 'visible';

Sve ostale stranice su identične u odnosu na prvu spirali, osim što je na svakoj stranici bilo potrebno ažurirati linkove na stranice zbog nove funkcije koja prikazje meni. Također je bilo potrebno dodati na svakoj stranici referencu na nove Javascript fajlove koje sam dodao. Dodano je par redova u .css da ispravno prikažu meni.

**3.SPIRALA**

Zadatak validacije forme putem AJAX-a. Dodao sam novu datoteku pod nazivom registracija.html

<div class="registracijaForma">

<form name="registracija" onsubmit="return validateRegistration()">

<p><span class="required">\* </span>Ime:</p>

<input type="text" id="ime" name="ime" class="input1" />

<img src="images/error.jpg" id="imeError" />

<p><span class="required">\* </span>Prezime:</p>

<input type="text" id="prezime" class="input1" />

<img src="images/error.jpg" id="prezimeError" />

<p><span class="required">\* </span>E-mail:</p>

<input type="text" id="email" class="input1" />

<img src="images/error.jpg" id="emailError" />

<p>Broj telefona:</p>

<input type="text" id="broj" class="input1" />

<img src="images/error.jpg" id="brojError" />

<p><span class="required">\* </span>Adresa:</p>

<input type="text" id="adresa" class="input1">

<img src="images/error.jpg" id="adresaError" />

<p><span class="required">\* </span>Opcina:</p>

<input type="text" id="opcina" class="input1" oninput="mjestoEnable()">

<img src="images/error.jpg" id="opcinaError">

<p><span class="required">\* </span>Mjesto</p>

<input type="text" id="mjesto" class="input1" disabled>

<img src="images/error.jpg" id="mjestoError">

<br /> <br />

<input type="submit" value="Posalji" />

<input type="reset" value="Poništi" />

</form>

</div>

Div sa klasom registracija forme predstavlja formu koju korisim za unos nekih podatka, od kojih se dva polja validiraju putem AJAX-a. Za servis sam odabrao validaciju općine i mjesta. Kada se forma submit-a, pokreće se funkcija validateRegistraion() koja vrši validaciju unesenih polja.

Sadržaj funkcije je ispod.

function validateRegistration() {

var ime = document.registracija.ime;

var prezime = document.registracija.prezime;

var email = document.registracija.email;

var broj = document.registracija.broj;

var adresa = document.registracija.adresa;

var opcina = document.registracija.opcina;

var mjesto = document.registracija.mjesto;

var pattern = /[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+(?:\.[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+)\*@(?:[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?/i;

if (ime.value == null || ime.value == "") {

document.getElementById('imeError').style.visibility = 'visible';

return false;

} else {

document.getElementById('imeError').style.visibility = 'hidden';

}

if (prezime.value == null || prezime.value == "") {

document.getElementById('prezimeError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('prezimeError').style.visibility = 'hidden';

}

if (!pattern.test(email.value) || email.value == "") {

document.getElementById('emailError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('emailError').style.visibility = 'hidden';

}

if(broj.value != "")

{

if(!isNumeric(broj.value))

{

document.getElementById('brojError').style.visibility = 'visible';

return false;

}

else

{

document.getElementById('brojError').style.visibility = 'hidden';

}

}

else

{

document.getElementById('brojError').style.visibility = 'hidden';

}

if (adresa.value == "") {

document.getElementById('adresaError').style.visibility = 'visible';

return false;

}

else {

document.getElementById('adresaError').style.visibility = 'hidden';

}

if (opcina.value == "")

{

document.getElementById('opcinaError').style.visibility = 'visible';

return false;

}

else

{

document.getElementById('opcinaError').style.visibility = 'hidden';

}

if (mjesto.value == "")

{

document.getElementById('mjestoError').style.visibility = 'visible';

return false;

}

else

{

document.getElementById('mjestoError').style.visibility = 'hidden';

}

if(ajaxValidacija(opcina.value,mjesto.value) == true)

{

alert("Uspješno ste se registrovali!");

}

}

function mjestoEnable()

{

if(document.getElementById('opcina').value=="")

{

document.getElementById('mjesto').disabled=true;

return;

}

document.getElementById('mjesto').disabled=false;

}

function ajaxValidacija(opcina,mjesto)

{

var ajax = new XMLHttpRequest();

var \_opcina = encodeURIComponent(opcina);

var \_mjesto = encodeURIComponent(mjesto);

ajax.onreadystatechange = function()

{

if(ajax.readyState == 4 && ajax.status == 200)

{

alert(ajax.responseText);

return true;

}

}

ajax.open("GET","http://zamger.etf.unsa.ba/wt/mjesto\_opcina.php?opcina="+ \_opcina + "&mjesto="+\_mjesto,true);

ajax.send();

}

Dakle svaka varijabla se najprije provjerava da li je prazna, i u slučaju da jeste na formi se ispise slika za grešku na isti način kao i kod kontakt forme ranije. Varijable općina i mjesto se provjeravaju tako što se proslijede u funkciju ajaxValidacija(opcina,mjesto) koja prima dva parametra. Sadržaj funkcije je dat ispod:

function ajaxValidacija(opcina,mjesto)

{

var ajax = new XMLHttpRequest();

var \_opcina = encodeURIComponent(opcina);

var \_mjesto = encodeURIComponent(mjesto);

ajax.onreadystatechange = function()

{

if(ajax.readyState == 4 && ajax.status == 200)

{

alert(ajax.responseText);

return true;

}

}

ajax.open("GET","http://zamger.etf.unsa.ba/wt/mjesto\_opcina.php?opcina="+ \_opcina + "&mjesto="+\_mjesto,true);

ajax.send();

}

Dakle funkcija prima dva parametra koja se prvo kodiraju da bi mogla da se proslijede servisu, a zatim ajax poziva servis koji u slučaju da je sve ok vraća poruku da je su varijable ispravne, odnosno da postoji mjesto u datoj općini.

Učitavanje podstranica se ostvaruje pomoću AJAX-a na način da AJAX učita metodom „GET“ neku stranicu, i vrati nam njen sadržaj kroz atribut responseText. Sadržaj funkcije koja vrši ovo učitavanje je ispod:

function fetchPage(page)

{

var ajax = new XMLHttpRequest();

ajax.onreadystatechange = function ()

{

if (ajax.readyState == 4 && ajax.status == 200)

{

document.getElementById("wrapper").innerHTML = ajax.responseText;

}

}

ajax.open("GET", page, true);

ajax.send();

}

Naredba document.getElementById("wrapper").innerHTML = ajax.responseText;

Pristupa elementu sa zadanim id-em i mijenja njegov sadržaj, odnosno ubacuje u njegovo tijelo sadržaj stranice koji je dobavljen pomoću AJAX-a. Na taj način smo ostvarili učitavanje podstranica bez reloada stranice. Na svim stranicama smo morali ažurirati linkove do stranica, tako što je na svakoj stranici meni zamijenjen dijelom koda koji se nalazi ispod.

<div id="menu">

<ul>

<li>

<a onclick="fetchPage('index.html')">Početna</a></li>

<li><a onclick="fetchPage('aboutUs.html')">O nama</a>

</li> <li><a onclick="fetchPage('kalendar.html')">Kalendar</a>

</li> <li><a onclick="fetchPage('novosti.html')">Novosti</a>

</li> <li><a onclick="fetchPage('kontakt.html')">Kontakt</a>

</li><li onmouseover="showMenu();" onmouseout="hideMenu();"><a>Uputstva</a>

<div id="drop\_down">

<a>Teorijski ispit</a>

<a>Prakticni ispit</a>

<a>Prva pomoc</a>

</div>

</li>

</ul>

</div>

4.SPIRALA

Najprije zadatak validacije sa PHP-om.

Dodao sam fajl kontakt.php koji je identičan datoteci kontakt.html iz prošle spirale osim što sada sadrži malo php koda u sebi koji obavlja validaciju forme na serveru.

$ime = $prezime = $email = $broj = $poruka = $unos = "";

$imeErr = $prezimeErr = $emailErr = $brojErr = $porukaErr = $unosErr = "";

if($\_SERVER["REQUEST\_METHOD"] == "POST")

{

$valid = 1;

if(empty($\_POST["ime"])){

$imeErr = '<style type="text/css">

#imeError {

visibility: visible;

}

</style>';

$valid = 0;

}

else{

$ime = test\_input($\_POST["ime"]);

if(validateName($ime)){}

else

{

$imeErr = '<style type="text/css">

#imeError {

visibility: visible;

}

</style>';

$valid = 0;

}

}

Na početku inicijaliziramo sve varijable i postavljamo da su prazne. Varijabla valid ima vrijenost 1 jer je pretpostavka da je forma ispravna. Dalje provjeravamo koja je metoda u pitanju, ako se sa forme submitaju podaci metodom POST, onda se podaci procesiraju u kodu koji slijedi. Za svaku varijablu provjeravam da li je prazna, ukoliko jeste postavljam vrijednost varijable valid na 0. Za svaku varijablu je isti postupak. Ukoliko varijabla nije prazna, provjerava se da li je unos ispravan, npr

if(empty($\_POST["email"]))

{

$emailErr = '<style type="text/css">

#emailError {

visibility: visible;

}

</style>';

$valid = 0;

}

else{

$email = test\_input($\_POST["email"]);

if(validateEmail($email)){

}

else {

$emailErr = '<style type="text/css">

#emailError {

visibility: visible;

}

</style>';

$valid = 0;

}

}

U ovom dijelu ispitujem da li je varijabla koja čuva mejl prazna, ukoliko nije dalje ispitujem da li je format mejla ispravna, na taj način što vrijednost varijable $email proslijedim funkciji validateEmail koja vraća true ili false u zavisnosti od ispravnosti unosa. Tijelo funkcije validateEmail je ispod.

function validateEmail($email){

$regexp = "^([\_a-z0-9-]+)(\.[\_a-z0-9-]+)\*@([a-z0-9-]+)(\.[a-z0-9-]+)\*(\.[a-z]{2,6})$";

if(eregi($regexp,$email)) return 1;

else return 0;

}

Dakle, primljenu varijablu poredi sa regexom i vraća rezultat. Ukoliko ne prođe validaciju, varijabla emailError dobija vrijednost

'<style type="text/css">

#emailError {

visibility: visible;

}

</style>';

koja će kada se ispiše kreirati CSS pravilo koje će prikazati sliku o grešci na formi.

function validateNumber($number){

if(!is\_nan($number) && is\_finite($number)) return 1;

else return 0;

}

function validateName($name){

$regexp1 = "^[A-Z]'?[-a-zA-Z]([a-zA-Z])\*$";

if(eregi($regexp1,$name)) return 1;

else return 0;

}

Funkcije validateNumber i validateName jednostavno provjeravaju da li varijabla sadrži samo brojeve odnosno samo slova respektivno.

Da bismo ispunili zahtjev da vrijednost koje su submitane sa forme ostanu upisane u poljima, u atribut svakog input polja na formi postavljamo ovaj mali dio php koda

<?php

if(isset($\_REQUEST['prezime']))

print $\_REQUEST['prezime'];

?>"

koji ispisuje vrijednost odgovarajuće varijable. Ispod je kod forme.

<div class="kontaktforma">

<form name="kontaktforma" id="kontaktForma" method="POST" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

<p><span class="required">\* </span>Ime:</p>

<input type="text" id="ime" name="ime" class="input1" value="<?php

if(isset($\_REQUEST['ime']))

print $\_REQUEST['ime'];

?>" /> <span><?php echo $imeErr ?></span>

<img src="images/error.jpg" id="imeError" />

<p><span class="required">\* </span>Prezime:</p>

<input type="text" id="prezime" name="prezime" class="input1" value="<?php

if(isset($\_REQUEST['prezime']))

print $\_REQUEST['prezime'];

?>" />

<span><?php echo $prezimeErr ?></span>

<img src="images/error.jpg" id="prezimeError" />

<p><span class="required">\* </span>E-mail:</p>

<input type="text" id="email" name="email" class="input1" value="<?php

if(isset($\_REQUEST['email']))

print $\_REQUEST['email'];

?>" />

<span><?php echo $emailErr ?></span>

<img src="images/error.jpg" id="emailError" />

<p>Broj telefona:</p>

<input type="text" id="broj" name="broj" class="input1" value="<?php

if(isset($\_REQUEST['broj']))

print $\_REQUEST['broj'];

?>" />

<span><?php echo $brojErr ?></span>

<img src="images/error.jpg" id="brojError" />

<p><span class="required">\* </span>Poruka:</p>

<textarea rows="4" id="poruka" cols="50" name="poruka" class="input1"><?php

if(isset($\_REQUEST['poruka']))

print $\_REQUEST['poruka'];

?></textarea>

<span><?php echo $porukaErr ?></span>

<img src="images/error.jpg" id="porukaError" />

<br />

<img src="images/captcha.jpg" id="captcha"><br />

<p><span class="required">\* </span>Unesite tekst sa slike:</p>

<input type="text" id="unos" name="unos" class="input1" value="<?php

if(isset($\_REQUEST['unos']))

print $\_REQUEST['unos'];

?>">

<span><?php echo $unosErr ?></span>

<img src="images/error.jpg" id="unosError" /><br>

<input type="submit" value="Posalji" onclick="validateForm();" />

<input type="reset" value="Poništi" />

</form>

</div>

Ovaj dio koda

<form name="kontaktforma" id="kontaktForma" method="POST" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

je bitan jer u njemu postavljamo atribute method i action, koji znače da je metoda POST, odnosno da je akcija isti php fajl kome je forma i poslana.

if($valid){

echo "

<script type=\"text/javascript\">

var e = document.getElementById('kontaktForma'); e.action='potvrda.php'; e.submit();

</script>

";

}

?>

Ovaj dio koda se izvrši ako je validacija uspješna, a on overrida atribute forme na način da je preusmjeri u drugi php fajl, u ovom sluvaju potvrda.php.

Datoteka potvrda.php

<?php

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

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

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

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

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

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<title>Potvrda kontakt forme</title>

<link href="style.css" media="screen" rel="stylesheet" />

</head>

<body>

<div id="wrapper">

<!-- header image -->

<div><img alt="" src="images/head.jpg" class="glavnaSlika" />

</div>

<!-- menu -->

<div id="menu">

<ul>

<li>

<a href="index.php">Početna</a>

</li>

<li>

<a onclick="fetchPage('aboutUs.html')">O nama</a>

</li>

<li>

<a onclick="fetchPage('kalendar.html')">Kalendar</a>

</li>

<li>

<a onclick="fetchPage('novosti.php');">Novosti</a>

</li>

<li>

<a href="kontakt.php">Kontakt</a>

</li>

<li onmouseover="showMenu();" onmouseout="hideMenu();"><a>Uputstva</a>

<div id="drop\_down">

<a>Teorijski ispit</a>

<a>Prakticni ispit</a>

<a>Prva pomoc</a>

</div>

</li>

</ul>

</div>

<div id="pregledPodataka">

<form id="emailForma" method="POST" action="slanjeMaila.php">

<h3>Provjerite da li ste ispravno popunili kontakt formu</h3>

<p>Ime: <?php echo $ime?></p><br>

<p>Prezime: <?php echo $prezime?></p><br>

<p>Email: <?php echo $email?></p><br>

<p>Broj: <?php echo $broj?></p><br>

<p>Poruka: <?php echo $poruka?></p><br>

<h3>Da li ste sigurni da želite poslati ove podatke?</h3>

<input type="button" value="Siguran sam" onclick="var e = document.getElementById('editForma'); e.action='slanjeMaila.php'; e.submit();">

</form>

</div>

<div class="kontaktforma">

<h3>Ako ste pogrešno popunili formu, možete ispod prepraviti unesene podatke</h3>

<form name="kontaktforma" id="editForma" method="POST" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

<p><span class="required">\* </span>Ime:</p>

<input type="text" id="ime" name="ime" class="input1" value="<?php

if(isset($\_REQUEST['ime']))

print $\_REQUEST['ime']; ?>">

<p><span class="required">\* </span>Prezime:</p>

<input type="text" id="prezime" name="prezime" class="input1" value="<?php

if(isset($\_REQUEST['prezime']))

print $\_REQUEST['prezime'];

?>" />

<p><span class="required">\* </span>E-mail:</p>

<input type="text" id="email" name="email" class="input1" value="<?php

if(isset($\_REQUEST['email']))

print $\_REQUEST['email'];

?>" />

<p>Broj telefona:</p>

<input type="text" id="broj" name="broj" class="input1" value="<?php

if(isset($\_REQUEST['broj']))

print $\_REQUEST['broj'];

?>" />

<p><span class="required">\* </span>Poruka:</p>

<textarea rows="4" id="poruka" cols="50" name="poruka" class="input1"><?php

if(isset($\_REQUEST['poruka']))

print $\_REQUEST['poruka'];

?></textarea>

<br />

<input type="submit" value="Posalji" />

<input type="reset" value="Poništi" />

</form>

</div>

<!-- footer -->

<div id="footer">

<div id="copyright">Copyright &copy; 2015 <a href="#">Auto škola "Iris"</a>. All Rights Reserved.

<br />

<p>

<a href="http://validator.w3.org/check?uri=referer">

<img src="http://www.w3.org/Icons/valid-xhtml10" alt="Valid XHTML 1.0 Transitional" height="31" width="88" />

</a>

</p>

</div>

</div>

</div>

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

</body>

</html>

Sastoji se od dvije forme, jedne koja je u donjem dijelu i čije su vrijendost automatski postavljene kada se pozove ovaj fajl. Korisnik može da mijenja vrijednosti u poljima donje forme i klikom na dugme Pošalji ti podaci se prebacuju na gornji dio stranice gdje su prikazani sa mogućnošću da se dalje mijenjaju ili da se kao takvi pošalju mejlom.

<input type="button" value="Siguran sam" onclick="var e = document.getElementById('editForma'); e.action='slanjeMaila.php'; e.submit();">

Ovaj dio koda u gornjoj formi služi da kada se klikne na njega, pošalje vrijednosti iz donje forme u php fajl koji obavlja slanje mejla.

**PHPMailer**

<?php

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

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

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

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

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

require 'PHPMailer/PHPMailerAutoload.php';

$mail = new PHPMailer;

$mail->isSMTP();

$mail->Host = 'smtp.live.com';

$mail->SMTPAuth = true;

$mail->Username = 'auto\_skola\_iris@hotmail.com';

$mail->Password = 'AutoSkolaIris';

$mail->SMTPSecure = 'tls';

$mail->Port = 587;

$mail->addReplyTo($email, $ime);

$mail->From = 'auto\_skola\_iris@hotmail.com';

$mail->FromName = 'Auto Skola IRIS';

$mail->addAddress('auto\_skola\_iris@hotmail.com', 'Auto Skola IRIS');

$mail->WordWrap = 50;

$mail->isHTML(false);

$mail->Subject = 'Poruka';

$mail->Body = "Ime: ".$ime."\n"."Prezime: ".$prezime."\n"."Email: ".$email."\n"."Broj: ".$broj."\n"."Poruka: ".$poruka;

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

echo 'Nije moguće poslati poruku.';

echo 'Detaljnije o grešci:' . $mail->ErrorInfo;

echo ' <br>';

echo '<a href="index.php">Povratak na početnu stranicu.</a>';

exit;

}

echo 'Zahvaljujemo se što ste nas kontaktirali';

echo '<a href="index.php">Povratak na početnu stranicu.</a>';

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<title></title>

</head>

<body>

</body>

</html>

Za slanje mejla iskoristio sam biblioteku PHPMailer, u sljedećim spiralama koristio sam SendGrid.

**NOVOSTI**

<?php

$sveNovosti = glob("novosti/\*.txt");

$novosti = array();

foreach($sveNovosti as $novost){

array\_push($novosti,file($novost));

}

foreach($novosti as $temp){

$sadrzaj = $temp;

$dateTime = trim($sadrzaj[0]);

$imeAutora = trim($sadrzaj[1]);

$naslovNovosti = trim(ucfirst(strtolower($sadrzaj[2])));

$slika = $sadrzaj[3];

$opisNovosti = "";

$detaljniOpisNovosti = "";

$detaljnije = "";

$imaDetaljno = 0;

for($i = 4; $i< count($sadrzaj); $i++){

if($sadrzaj[$i] === "--\n"){

$imaDetaljno = 1;

continue;

}

else if(!$imaDetaljno) {

$opisNovosti .= trim($sadrzaj[$i]);

}

if($imaDetaljno){

$detaljniOpisNovosti .= trim($sadrzaj[$i]);

}

}

if($imaDetaljno){

$datum = encodeURIComponent($dateTime);

$autor = encodeURIComponent($imeAutora);

$naslov = encodeURIComponent($naslovNovosti);

$slicica = encodeURIComponent($slika);

$opis = encodeURIComponent($opisNovosti);

$detaljno = encodeURIComponent($detaljniOpisNovosti);

$detaljnije = "<input type='button' value='detaljnije' onclick=\"fetchPage('detaljnije.php?dateTime=$datum&autor=$autor&naslov=$naslov&opis=$opis&detaljno=$detaljno&slika=$slicica');\">";

}

echo ' <div class="novost">

<div class="naslov\_vijesti">

<h3>'.$naslovNovosti.'</h3>

</div>

<div class="datum\_objave">

<p>Autor:'.$imeAutora.'</p>

<p>Datum objave:'.$dateTime.'</p>

</div>

<div class="tekst\_novosti">

<p><img alt="" src='.$slika.' class="slika" />'.$opisNovosti.'

</p>

</div>

<div class="detaljnije">'.$detaljnije.'</div>

</div> ';

}

function encodeURIComponent($str) {

$revert = array('%21'=>'!', '%2A'=>'\*', '%27'=>"'", '%28'=>'(', '%29'=>')');

return strtr(rawurlencode($str), $revert);

}

?>

Ovaj dio php koda radi tako što povuče u jednu varijablu sve fajlove sa extenzoijom txt a zatim učitava svaki fajl kroz petlju u jednu varijablu. Taj fajl se dalje razbija na redove, pri čemu svaki red ide u drugu varijablu.

for($i = 4; $i< count($sadrzaj); $i++){

if($sadrzaj[$i] === "--\n"){

$imaDetaljno = 1;

continue;

}

else if(!$imaDetaljno) {

$opisNovosti .= trim($sadrzaj[$i]);

}

if($imaDetaljno){

$detaljniOpisNovosti .= trim($sadrzaj[$i]);

}

}

Ova funkcija služi da odvoji osnovni opis novosti od detaljnog.

$detaljnije = "<input type='button' value='detaljnije' onclick=\"fetchPage('detaljnije.php?dateTime=$datum&autor=$autor&naslov=$naslov&opis=$opis&detaljno=$detaljno&slika=$slicica');\">";

Ovaj dio na stranici prikazuje dugme detaljnije koje preusmjerava na drugi php fajl koji sadrži prikaz novosti sa detaljnijim opisom.Ukoliko ne postoji detaljniji opis ova varijabla je prazna, te neće prikazati ništa.

<?php

$dateTime = $\_GET['dateTime'];

$imeAutora = $\_GET['autor'];

$naslovNovosti = $\_GET['naslov'];

$slika = $\_GET['slika'];

$opisNovosti = $\_GET['opis'];

$detaljniOpisNovosti = $\_GET['detaljno'];

$opis = $opisNovosti.$detaljniOpisNovosti;

echo '

<head>

<meta charset="utf-8" />

<title>Auto-Skola Iris</title>

<link href="style.css" media="screen" rel="stylesheet" />

</head>

<div id="main">

<div class="novosti">

<div class="novost">

<div class="naslov\_vijesti">

<h3>'.$naslovNovosti.'</h3>

</div>

<div class="datum\_objave">

<p>Autor:'.$imeAutora.'</p>

<p>Datum objave:'.$dateTime.'</p>

</div>

<div class="tekst\_novosti">

<p><img alt="" src='.$slika.' class="slika" />'.$opis.'

</p>

</div>

</div>

</div>

</div> ';

?>

Ovaj dio koda se nalazi u fajlu detaljnije.php i njegova svrha je da prikaže vijest sa detaljnijim opisom. Nema ništa specifično.

**5.SPIRALA**

NOVOSTI U BAZI

<?php

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$baza->exec("set names utf8");

$rezultat = $baza->query("select id,naslov,tekst,autor,slika,UNIX\_TIMESTAMP(vrijeme) vrijeme1 from novosti order by vrijeme1");

if(!$rezultat){

$greska = $veza->errorInfo();

print "SQL greska:".$greska[2];

exit();

}

foreach($rezultat as $temp){

$imaDetaljno = 0;

$sadrzaj = $temp;

$dateTime = date("d.m.Y. H:i:s", $sadrzaj['vrijeme1']);

$id = $sadrzaj['id'];

$imeAutora = trim($sadrzaj['autor']);

$naslovNovosti = trim(ucfirst(strtolower($sadrzaj['naslov'])));

$opisNovosti = trim($sadrzaj['tekst']);

$slika = $sadrzaj['slika'];

$detaljniOpisNovosti = "";

$detaljnije = "";

$indeks = strpos($opisNovosti, '--');

if($indeks !== FALSE){

$imaDetaljno = 1;

$pomocna = $opisNovosti;

$opisNovosti = substr($opisNovosti, 0 , $indeks);

$detaljniOpisNovosti = substr($pomocna, $indeks);

}

$detaljnije = "<input type='button' value='Idi na vijest' onclick=\"fetchPage('detaljnije.php?id=$id');\">";

//dobijanje broja komentara

$komentari = $baza->query("select id, vijest,tekst, autor, UNIX\_TIMESTAMP(vrijeme) vrijeme2, email from komentari where vijest = '$id'");

$brojKomentara = $komentari->rowCount();

echo ' <div class="novost">

<div class="naslov\_vijesti">

<h3>'.$naslovNovosti.'</h3>

</div>

<div class="datum\_objave">

<p>Autor: '.$imeAutora.'</p>

<p>Datum objave: '.$dateTime.'</p>

</div>

<div class="tekst\_novosti">

<p><img alt="" src='.$slika.' class="slika" />'.$opisNovosti.'

</p>

</div>

<p>Broj komentara:'.$brojKomentara.'</p>

<div class="detaljnije">'.$detaljnije.'</div>

</div> ';

}

Ovaj dio koda se nalazi u fajlu novosti.php i služi da pristupi bazi i pokupi sve novosti.

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$baza->exec("set names utf8");

$rezultat = $baza->query("select id,naslov,tekst,autor,slika,UNIX\_TIMESTAMP(vrijeme) vrijeme1 from novosti order by vrijeme1");

Ovaj dio služi za pristup bazi. Query naredba vrši odmah upit nad bazom, dok možemo upit da pripremimo koristeći prepare naredbu a zatim da je izvršimo kasnije.

foreach($rezultat as $temp){

Ovaj dio prolazi kroz svaku novost koju smo pokupili iz baze, smješta vrijednosti u odgovarajuće varijable i ispisiuje novost na stranici.

Komentari na novosti se ostavljaju kada se klikne na dugme „Idi na vijest“.

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$naredba = $baza->prepare('select id,naslov,tekst,autor,slika,UNIX\_TIMESTAMP(vrijeme) vrijeme1 from novosti where id=:id');

$naredba->execute(array('id' => $id));

$rezultat = $naredba->fetch();

//preuzimanje komentara

$komentari = $baza->query("select id, vijest,tekst, autor, UNIX\_TIMESTAMP(vrijeme) vrijeme2, email from komentari where vijest = '$id'");

$brojKomentara = $komentari->rowCount();

Ovaj dio koda se nalazi u fajlu detaljnije.php. Njegova svrha je da pristupi bazi, preuzme novost prema parametru ID koji primi kroz metodu GET. Zatim, na osnovu tog ID parametra, pristupi bazi i preuzme sve komentare koji su vezani za datu vijest.

</div>

<div id = "komentari">';

foreach($komentari as $komentar){

$autor = $komentar['autor'];

$vrijeme = date("d.m.Y. H:i:s", $komentar['vrijeme2']);

$tekst = $komentar['tekst'];

$email = $komentar['email'];

echo '

<div class="komentar">

<p class="imeKomentar">'.$autor.'</p>

<p class="imeKomentar"><span class="ravnanjeDesno">'.$vrijeme.'</span></p>

<p>'.$tekst.'</p>

<a href="mailto:'.$email.'">'.$email.'</a>

</div>

';

}

echo '

</div>

Ovaj dio koda služi da ispiše sve komentare na stranici. Prolazeći kroz foreach petlju php generiše html kod koji prikazuje komentar. Ovaj cijeli div je sakriven.

<a id="komentarNaredba" onclick="prikaziKomentare();">Prikazi sve komentare</a>

Klik na ovaj link prikazuje/sakriva komentare.

Ispod je sadržaj funkcije koja se bavi ovime.

prikaziKomentare = function ()

{

if (document.getElementById('komentarNaredba').innerHTML === "Prikazi sve komentare")

{

document.getElementById('komentari').style.visibility = 'visible';

document.getElementById('komentari').style.height = '400px';

document.getElementById('komentarNaredba').innerHTML = "Sakrij sve komentare";

}

else

{

document.getElementById('komentari').style.visibility = 'hidden';

document.getElementById('komentari').style.height = '5px';

document.getElementById('komentarNaredba').innerHTML = "Prikazi sve komentare"

}

}

noviKomentar = function ()

{

document.getElementById('dodajKomentar').style.visibility = 'visible';

}

Funkcija referencija div komentari i mijenja mu atribut visible.

<input type="button" onclick="noviKomentar();" value="Komentarisi">

</div>

<div id="dodajKomentar">

<form id="FdodajKomentar" name = "noviKomentarForma" method="POST" action="dodajKomentar.php">

<p>Ime:</p>

<input type="text" name="ime">

<p>Komentar:</p>

<input type="text" name="tekst">

<p>Email:</p>

<input type="text" name="email">

<input type="hidden" value='.$id.' name="id">

<input type ="submit" value="Dodaj komentar">

</form>

</div>

</div> ';

Ovo je forma za dodavanje novog komentara. Nakon submita parametri se šalju u fajl dodajKomentar.php.

<?php

if($\_SERVER["REQUEST\_METHOD"] == "POST"){

$location = $\_SERVER['HTTP\_REFERER'];

$ime = htmlspecialchars($\_POST['ime']);

$noviKomentar = htmlspecialchars($\_POST['tekst']);

$emailTemp = htmlspecialchars($\_POST['email']);

$vijest = htmlspecialchars($\_POST['id']);

try{

echo $location.$ime.$noviKomentar.$emailTemp.$vijest;

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$ubaci = $baza->prepare("insert into komentari (vijest,tekst,autor,email) values(:id,:noviKomentar,:ime,:email)");

if( $ubaci->execute(array('id'=>$vijest,'noviKomentar'=>$noviKomentar, 'ime'=>$ime, 'email'=>$emailTemp))){

echo "uspjesno!";

}

else echo 'Neeee';

header("Location: novosti.php");

}

catch(PDOException $e){

echo 'ERROR'.$e->getMessage();

}

}

?>

Ovaj dio koda služi za dodavanje komentara u bazu. Za to koristimo ovu naredbu $ubaci = $baza->prepare("insert into komentari (vijest,tekst,autor,email) values(:id,:noviKomentar,:ime,:email)");

Nakon ubacivanja preusmjeravamo se na novosti.php.

ADMIN

<?php

session\_start();

if(!isset($\_SESSION['username'])){

if(isset($\_POST['username'])){

$username = htmlspecialchars($\_POST['username']);

$password = htmlspecialchars($\_POST['password']);

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$upit = $baza->prepare("select \* from korisnici where username =:username");

$upit->execute(array('username'=>$username));

$rezultat = $upit->fetch();

if($rezultat['password'] !== $password){

echo 'p>Pogresno korisnicko ime ili lozinka. </p>';

echo'<a href="">Zaboravio si sifru?</a>';

}

else{

echo 'Uspjesno ste prijavljeni';

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

header("Location: adminPanel.php");

}

}

else {

include 'login.html';

}

}

else{

header("Location: adminPanel.php");

}

?>

Ovaj dio koda se nalazi u admin.php i služi da provjeri da li postoji prijavljen korisnik, ako postoji usmjerava ga na fajl adminPanel.php, ako ne postoji učitava podstranicu login.html i tjera korisnika da se prijavi, odnosno onemogućuje pristup admin panelu ukoliko korisnik nije prijavljen.

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

Ova naredba nam postavlja globalnu varijablu username koja je dostupna širom stranice, ali da bi joj pristupili moramo najprije otvoriti sesiju na stranici sa koje želimo da joj pristupimo.

<?php

if($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_REQUEST['dodajNovost'])){

$autor = htmlspecialchars($\_POST['autor']);

$naslov = htmlspecialchars($\_POST['naslov']);

$tekst = htmlspecialchars($\_POST['tekst']);

if(!empty($autor) && !empty($naslov) && !empty($tekst)){

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$upit = $baza->prepare("insert into novosti (naslov,tekst,autor) values (:naslov,:tekst,:autor)");

if($upit->execute(array('naslov'=>$naslov, 'autor'=>$autor, 'tekst'=>$tekst))){

echo "<script type='text/javascript'>alert('Uspjesno ste dodali novost!');</script>";

}

}

}

if($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_REQUEST['dodajKorisnika'])){

$username = htmlspecialchars($\_POST['username']);

$password = htmlspecialchars($\_POST['password']);

$email = htmlspecialchars($\_POST['email']);

if(!empty($username) && !empty($password) && !empty($email)){

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$upit = $baza->prepare("insert into korisnici values (:username,:password,:email)");

if($upit->execute(array('username'=>$username, 'password'=>$password, 'email'=>$email))){

echo "<script type='text/javascript'>alert('Uspjesno ste dodali korisnika!');</script>";

}

}

}

?>

Ovaj dio koda se bavi dodavanjem novosti i korisnika. Ukoliko je metoda kojom mu se šalju parametri POST i ukoliko je vrijednost dugmeta sa forme sa koje se šalju parametri „dodajNovost“ tada se izvršava dio koda koji dodaje novost u bazu. Identično se izvršava i dio koda koji dodaje novog korisnika.

MIJENJANJE NOVOSTI

<?php

session\_start();

if(!isset($\_SESSION['username'])){

header("Location: admin.php");

}

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

if($\_SERVER["REQUEST\_METHOD"] == "POST"){

$id = htmlspecialchars($\_POST['id']);

$autor = htmlspecialchars($\_POST['autor']);

$naslov = htmlspecialchars($\_POST['naslov']);

$tekst = htmlspecialchars($\_POST['tekst']);

$izmjena = $baza->prepare("update novosti set naslov = :naslov, tekst = :tekst, autor=:autor where id =:id");

if($izmjena->execute(array('naslov'=>$naslov,'tekst'=>$tekst, 'autor'=>$autor, 'id'=>$id))){

echo "<script>alert('Uspjesno ste izmijenili novost!');

window.location.href='adminPanel.php';</script>";

}

}

$id = htmlspecialchars($\_REQUEST['id']);

$upit = $baza->prepare("select id,naslov,tekst,autor,slika,UNIX\_TIMESTAMP(vrijeme) vrijeme1 from novosti where id=:id");

$upit->execute(array('id'=>$id));

$rezultat= $upit->fetch();

$autor = $rezultat['autor'];

$naslov = $rezultat['naslov'];

$tekst = $rezultat['tekst'];

?>

<div id="main">

<form id="izmjenaNovosti" method="post" action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]);?>">

<h5>Autor:</h5> <input type="text" name="autor" value="<?php echo $autor?>"><br>

<h5>Naslov:</h5> <input type="text" name="naslov" value="<?php echo $naslov?>"><br>

<h5>Naslov:</h5> <textarea rows="5" cols="40" name="tekst" ><?php echo $tekst?></textarea>

<input type="hidden" value="<?php echo $id;?>" name="id">

<input type="submit" value="Izmijeni" name="Izmijeni">

</form>

</div>

Ovaj dio koda u sebi sadrži formu sa vrijednosti novosti čiji su parametri poslani iz fajla promijeniNovost.php. Svrha ovog dijela koda je da izvrši update novosti u bazi i da preusmjeri nazad na admin panel.

PROMIJENINOVOSTI.PHP

<?php

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$upit = $baza->query("select id,naslov,tekst,autor,slika,UNIX\_TIMESTAMP(vrijeme) vrijeme1 from novosti");

foreach($upit as $novost){

$id = $novost['id'];

$imeAutora = trim($novost['autor']);

$naslovNovosti = trim(ucfirst(strtolower($novost['naslov'])));

$dateTime = date("d.m.Y. H:i:s", $novost['vrijeme1']);

$opisNovosti = trim($novost['tekst']);

$slika = $novost['slika'];

$obrisi = "<input type='button' value='Obrisi' onclick=\"fetchPage('obrisiNovost.php?id=$id');\">";

$izmijeni = "<input type='button' value='Izmijeni' onclick=\"fetchPage('izmijeniNovost.php?id=$id');\">";

$urediKomentare = "<input type='button' value='urediKomentare' onclick=\"fetchPage('urediKomentare.php?id=$id');\">";

echo'

<div id="main">

<div class="novostPanel">

<h5>'.$imeAutora.'</h5><h5>'.$dateTime.'</h5>

<h5>'.$naslovNovosti.'</h5>

<h5>'.$opisNovosti.'</h5>

<h5>'.$slika.'</h5>

'.$izmijeni.'

'.$obrisi.'

'.$urediKomentare.'

</div>

</div>

';

}

?>

Ovaj fajl služi da se izlistaju sve novosti i da korisnik klikom na određenu novost pristupi formi preko koje može da je promijeni. Klikom na dugme obrisi korisnik briše odabranu novost. Identičo se izvršava i sa korisnikom, odnosno izmjenom i brisanjem korisnika te nema potrebe navoditi ovdje to.

<?php

session\_start();

if(!isset($\_SESSION['username'])){

header("Location: admin.php");

}

else{

$id = htmlspecialchars($\_REQUEST['id']);

$baza = new PDO("mysql:dbname=wtbaza;host=localhost;charset=utf8","wtuser","wtuser");

$upit = $baza->prepare("delete from komentari where id=:id");

if($upit->execute(array('id'=>$id)))

{

echo 'Uspjesno ste obrisali komentar!';

}

}

?>

Ovaj dio koda briše komentar iz baze.

**6.SPIRALA**

**FUNCKIONALNOSTI PREBACITI NA SERVISE**

var valid = 1;

ajax.onreadystatechange = function ()

{

if (ajax.readyState == 4 && ajax.status == 200)

{

var json = ajax.responseText;

var obj = JSON.parse(json);

if (obj.ime)

{

document.getElementById('imeError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('imeError').style.visibility = 'hidden';

if (obj.prezime)

{

document.getElementById('prezimeError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('prezimeError').style.visibility = 'hidden';

if (obj.email)

{

document.getElementById('emailError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('emailError').style.visibility = 'hidden';

if (obj.broj)

{

document.getElementById('brojError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('brojError').style.visibility = 'hidden';

if (obj.poruka)

{

document.getElementById('porukaError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('porukaError').style.visibility = 'hidden';

if (obj.unos)

{

document.getElementById('unosError').style.visibility = 'visible';

valid = 0;

} else document.getElementById('unosError').style.visibility = 'hidden';

if (!valid) return false;

else

{

//var d = new Date();

//d.setTime(d.getTime() + (24 \* 60 \* 60 \* 1000));

//var expires = "expires=" + d.toUTCString();

//document.cookie = "ime=" + ime + ";" + expires;

//document.cookie = "prezime=" + prezime + ";" + expires;

//document.cookie = "email=" + email + ";" + expires;

//document.cookie = "broj=" + broj + ";" + expires;

//document.cookie = "poruka" + poruka + ";" + expires;

//if (fetchPage('potvrda.html'))

//{

// potvrda(ime, prezime, email, broj, poruka);

//}

//updateFormu();

var ajax1 = new XMLHttpRequest();

ajax1.onreadystatechange = function ()

{

if (ajax1.readyState == 4 && ajax1.status == 200)

{

document.getElementById('content').innerHTML = ajax1.responseText;

potvrda(ime, prezime, email, broj, poruka);

updateFormu();

}

}

ajax1.open("GET", 'potvrda.html', true);

ajax1.send();

}

}

}

ajax.open("GET", "kontaktValidacija.php?ime=" + ime + "&prezime=" + prezime + "&email=" + email + "&broj=" + broj + "&poruka=" + poruka + "&unos=" + unos, true);

ajax.send();

Ovaj dio koda u fajlu kontaktValidacija.js vrši AJAX poziv servisa za validaciju forme.

Prvi AJAX poziv vrši validaciju a drugi preusmjerava na fajl potvrda.php.

$rezultat = array('ime'=> $imeErr, 'prezime'=>$prezimeErr, 'email'=>$emailErr, 'broj'=>$brojErr, 'poruka'=>$porukaErr, 'unos'=>$unosErr);

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

echo json\_encode($rezultat);

Nakon klasične validacije varijabli, sve se smješta u niz i parsira u JOSN format koji se vraća iz servisa.

if (obj.ime)

{

document.getElementById('imeError').style.visibility = 'visible';

valid = 0;

Ovaj dio koda prikazuje sliku sa greškom na formi ukoliko je servis vratio true za neku varijablu što znači da nije prošla.

NOVOSTI SERVIS

$list = array();

foreach($rezultat as $temp){

$imaDetaljno = 0;

$sadrzaj = $temp;

$dateTime = date("d.m.Y. H:i:s", $sadrzaj['vrijeme1']);

$id = $sadrzaj['id'];

$imeAutora = trim($sadrzaj['autor']);

$naslovNovosti = trim(ucfirst(strtolower($sadrzaj['naslov'])));

$opisNovosti = trim($sadrzaj['tekst']);

$slika = $sadrzaj['slika'];

$detaljniOpisNovosti = "";

$detaljnije = "";

$indeks = strpos($opisNovosti, '--');

if($indeks !== FALSE){

$imaDetaljno = 1;

$pomocna = $opisNovosti;

$opisNovosti = substr($opisNovosti, 0 , $indeks);

$detaljniOpisNovosti = substr($pomocna, $indeks);

}

//dobijanje broja komentara

$komentari = $baza->query("select id, vijest,tekst, autor, UNIX\_TIMESTAMP(vrijeme) vrijeme2, email from komentari where vijest = '$id'");

$brojKomentara = $komentari->rowCount();

$list[] = array('id' => $id, 'imeAutora' => $imeAutora, 'dateTime' => $dateTime, 'naslov' => $naslovNovosti, 'opis' => $opisNovosti,

'detaljniOpis' => $detaljniOpisNovosti, 'slika' => $slika, 'brojKomentara' => $brojKomentara);

}

echo json\_encode($list);

Ovaj dio koda preuzima sve novosti iz baze i kodira u JSON format te ih kao takve vraća nazad na AJAX pozivu.

function dobaviNovosti()

{

var ajax = new XMLHttpRequest();

ajax.onreadystatechange = function ()

{

if (ajax.readyState == 4 && ajax.status == 200)

{

var json = ajax.responseText;

var obj = JSON.parse(json);

ispisiNovosti(obj);

}

}

ajax.open("GET", "novostiServis.php", true);

ajax.send();

}

Ova funkcija vrši AJAX poziv servisa i ukoliko dobije iz servisa podatkae poziva funkciju ispisiNovosti koja na formi ispisuje novosti.

function ispisiNovosti(obj)

{

for (var i = 0; i < obj.length; i++)

{

var div = document.createElement("div");

div.setAttribute('class', 'novost');

var div1 = document.createElement("div");

div1.setAttribute('class', 'naslov\_vijesti');

var h3 = document.createElement("h3");

var datum = document.createTextNode(obj[i].naslov);

h3.appendChild(datum);

var div2 = document.createElement("div");

div2.setAttribute('class', 'datum\_objave');

var p = document.createElement("p");

var autor = document.createTextNode(obj[i].imeAutora);

p.appendChild(autor);

var p1 = document.createElement("p");

var autor = document.createTextNode(obj[i].dateTime);

p1.appendChild(autor);

var div3 = document.createElement("div");

div3.setAttribute('class', 'tekst\_novosti');

var p2 = document.createElement("p");

var opis = document.createTextNode(obj[i].opis);

p2.appendChild(opis);

var p3 = document.createElement("p");

var brKomentara = document.createTextNode("Broj komentara:"+obj[i].brojKomentara);

var a = document.createElement("a");

a.setAttribute("onclick", "dobaviKomentare("+obj[i].id+");");

p3.appendChild(a);

a.appendChild(brKomentara);

var dugme = document.createElement("input");

dugme.setAttribute("type", "button");

dugme.setAttribute("value", "detaljnije");

dugme.setAttribute("onclick", "fetchPage('detaljnije.html'); dobaviKomentare("+obj[i].id+");");

document.getElementById('target').appendChild(div);

div.appendChild(div1);

div1.appendChild(h3);

div.appendChild(div2);

div2.appendChild(p);

div2.appendChild(p1);

div.appendChild(div3);

div3.appendChild(p2);

div.appendChild(p3);

div.appendChild(dugme);

}

}

Ova funkcija kroz JavaScript generiše html kod koji prikazuje novosti na stranici.

function dobaviKomentare(id)

{

var ajax = new XMLHttpRequest();

ajax.onreadystatechange = function ()

{

if (ajax.readyState == 4 && ajax.status == 200)

{

var obj = ajax.responseText;

var vijest = JSON.parse(obj);

ispisiVijest(vijest,id);

}

}

ajax.open("GET", "komentariServis.php?id=" + id, true);

ajax.send();

}

Ova funkcija vrši AJAX poziv servisa komentariServis koji vraća niz komentara za odgovarajuću novost i poziva funkciju ispiši vijest koja ispisuje vijest zajedno sa komentarima.

Ispisi vijest se od fukcije ispisiNovosti razlikuje po dijelu koda ispod:

for(var i = 0; i< obj.\_komentari.length; i++)

{

var div\_kom1 = document.createElement("div");

div\_kom1.setAttribute("class","komentar");

var p\_ime = document.createElement("p");

p\_ime.setAttribute("class","imeKomentar");

p\_ime.appendChild(document.createTextNode(obj.\_komentari[i].autor));

var p\_vr = document.createElement("p");

p\_vr.setAttribute("class","imeKomentar");

p\_vr.appendChild(document.createTextNode(obj.\_komentari[i].vrijeme));

var p\_email = document.createElement("p");

p\_email.appendChild(document.createTextNode(obj.\_komentari[i].email));

var p\_tekst = document.createElement("p");

p\_tekst.appendChild(document.createTextNode(obj.\_komentari[i].tekst));

div\_kom.appendChild(div\_kom1);

div\_kom1.appendChild(p\_ime);

div\_kom1.appendChild(p\_vr);

div\_kom1.appendChild(p\_email);

div\_kom1.appendChild(p\_tekst);

}

Koji na formi ispisuje komentare.

var input = document.createElement("input");

input.setAttribute("type", "button");

input.setAttribute("onclick", "noviKomentar();");

input.setAttribute("value", "Komentarisi");

div.appendChild(input);

var div\_dodajKom = document.createElement("div");

div\_dodajKom.setAttribute("id", "dodajKomentar");

var forma = document.createElement("form");

forma.setAttribute("id", "FdodajKomentar");

forma.setAttribute("name", "noviKomentarForma");

//forma.setAttribute("method", "POST");

//forma.setAttribute("action", "dodajKomentar.php");

var u\_ime = document.createElement("p");

u\_ime.appendChild(document.createTextNode("Ime: "));

var inp\_ime = document.createElement("input");

inp\_ime.setAttribute("type", "text");

inp\_ime.setAttribute("name", "ime");

var u\_kom = document.createElement("p");

u\_kom.appendChild(document.createTextNode("Komentar: "));

var inp\_komentar = document.createElement("input");

inp\_komentar.setAttribute("type", "text");

inp\_komentar.setAttribute("name", "tekst");

var u\_email = document.createElement("p");

u\_email.appendChild(document.createTextNode("Email: "));

var inp\_email = document.createElement("input");

inp\_email.setAttribute("type", "text");

inp\_email.setAttribute("name", "email");

var inp\_hidden = document.createElement("input");

inp\_hidden.setAttribute("type", "hidden");

inp\_hidden.setAttribute("value", id);

inp\_hidden.setAttribute("name", "id");

var br = document.createElement("br");

var inp\_submit = document.createElement("input");

inp\_submit.setAttribute("type", "button");

inp\_submit.setAttribute("onclick","dodajKomentar();")

inp\_submit.setAttribute("value", "Dodaj Komentar");

document.getElementById('target').appendChild(div\_dodajKom);

div\_dodajKom.appendChild(forma);

forma.appendChild(u\_ime);

forma.appendChild(inp\_ime);

forma.appendChild(u\_kom);

forma.appendChild(inp\_komentar);

forma.appendChild(u\_email);

forma.appendChild(inp\_email);

forma.appendChild(inp\_hidden);

forma.appendChild(br);

forma.appendChild(inp\_submit);

Ovaj dio koda ispisuje formu za dodavanje komentara i submitanje komentara dalje.