

ELEKTROTEHNIČKI FAKULTET UNIVERZITETA U BEOGRADU



DIPLOMSKI RAD

– Implementacija onlajn biblioteke JavaScript funkcija–

Kandidat:

Yazan Shihadeh 2008/0267

Mentor:

doc. dr ZoranČiča

Beograd, Oktobar 2013.

SADRŽAJ

SADRŽAJ	2
1. UVOD	3
2. ARHITEKTURA REALIZOVANE BIBLIOTEKE	5
3. STRANA ZA DODAVANJE FUNKCIJA	8
3.1. DODAJ FUNKCIJU	9
3.1.1. <i>Fajlovi (.js/.html)</i>	12
3.1.2. <i>Otpremanje i premeštanje fajlova</i>	13
3.2. IZMENI FUNKCIJU	13
3.3. BRISANJE FUNKCIJE	15
3.4. DODAVANJE/BRISANJE KORISNIKA	15
4. „PRISTUPNA “ STRANA	17
4.1. STRANA <i>NAZIV</i>	17
4.2. STRANA <i>TIP</i>	19
4.3. STRANA „ <i>PRIMER</i> “	20
5. ZAKLJUČAK	22
6. PRILOZI	23
LITERATURA	29

1. UVOD

JavaScript je skriptni programski jezik koji se prvenstveno koristi za definisanje funkcionalnosti web stranica na klijentskoj strani. Dinamičan, slabo tipiziran jezik, sa skromnom podrškom za objektno orijentisano programiranje, on zapravo predstavlja implementaciju standarda Ecmaskript.

Iako je prvobitna ideja bila da se *JavaScript* liči na *Javu*, on ne sadrži nijedan ključni element *Jave* osim onih koje su oba nasledila iz C-a. Jezik je najpoznatiji po programiranju klijentske funkcionalnosti web stranica, ali se može koristiti i kao skriptni jezik za pristup objektima u drugim aplikacijama.

Jezik se prvo zvao "LiveScript", ali je ime promenjeno u JavaScript, zbog marketinškog dogovora između *Netscape* i *Sun Microsystems-a* oni su ga predstavili decembra 1995., pod verzijom JavaScript 1.0. Ključni element dizajna *JavaScript-a* su nasleđeni iz programskog jezika *Self*.

JavaScript se uključuje u sadržaj HTML dokumenata i omogućava unepredenje HTML strana sa interesantnim efektima. Na primer, korićenjem *JavaScript-aje* moguće odgovoriti na akcije korisnika u samom *www* čitaču. Ono što je bitno zapamtiti jeste da *JavaScript* program može da se izvršava samo u okviru *www* čitača i nigde drugde.

Osnove *JavaScript-a* su:

- Tag
- Komentari
- Prikaz
- Promenljive i njihovi tipovi
- Operatori
- Funkcije
- Objekti

Da bi browser prepoznao JavaScript kod, potrebno je da taj se taj kod postavi unutar para oznaka `<script>` i `</script>`. U zavisnosti od Web čitača oznaka `<script>` ima širok dijapazon atributa koji se mogu podesiti kako bi upravljao skriptom.

Izvršavanje koda:

- U okviru (X)HTML koda: `<script type="text/javascript">...JavaScript kod...;</script>`
- Ili u eksternom fajlu : `<script type="text/javascript" src="jskod.js"></script>`
- Ili nekad bez **script**taga
- Naredba se izvršava tačka zarezom ;

Cilj rada je da se studentima, odnosno korisnicima web strane omogući da na jednom mestu imaju zbirku *JavaScript* funkcija kojima je moguće pristupiti u bilo koje vreme, i za cilj ima da olakša učenje *JavaScript-a*. Svaka funkcija se može pojedinačno pregledati, videti njen opis, kod funkcije i njena praktična primena.

U sledećim poglavljima je se prikazuje način implementacije i funkcionisanja platforme. U poglavlju **Arhitektura realizovane biblioteke** će biti prikazan način realizovanja cele platforme. Poglavlje **Strana za dodavanje funkcije** prikazuje način dodavanja, izmene postojeće funkcije kao i mogućnost brisanja dodate funkcije. U poglavlju „**Pristupna**“ strana će biti opisana strane za pregled, odnosno strane kojoj korisnici pristupaju i na kojoj mogu da pregledaju funkcije. **Prilozi** će sadržati određene kodove kojima je platforma realizovana.

Motivacija za ovaj projekat jeste da se korisnicima olakša proces učenja *JavaScript-a* i da učenje učini što kvalitetnijim. Samo praktično prikazivanja funkcije bi učinilo učenje produktivnijim i skratilo bi vreme potrebno za savladavanje *JavaScript-a*.

2. ARHITEKTURA REALIZOVANE BIBLIOTEKE

Osnovna ideja implementacije onlajn biblioteke *JavaScript* funkcija je da korisnik/posetilac sajta ima na jednom mestu biblioteku funkcija, kojoj može po potrebi pristupiti. Imajući u vidu da je tokom učenja JavaScript-a iz knjiga potrebno duže vreme za razumevanje kako funkcije uopšte funkcionišu, našla se ideja da korisnik na jednom mestu ima pristup raznim funkcijama, da može da pročita opis funkcije, da vidi *kod* i zaista praktičnu primeni tog koda.

Implementacija je izvršena na sledeći način. Prvo, postoji strana koja se može nazvati „Pristupna“. Ona predstavlja onu stranu kojoj korisnik pristupa, na kojoj se mogu videti sve funkcije. One su grupisane po grupama kojima pripadaju ili sortirane po abecednom redu. Naslova strana „Pristupne“ strane se može videti na slici 2.1. Namena ove strane je se da prikažu funkcije koje se nalaze u bazi. Funkcije su sortirane radi lakšeg nalaženja željenog primera, gde je moguće da se svaki primer prikaže pojedinačno na posebnoj strani sa detaljnim opisom njegove namene i realne primene.



Slika 2.1., „Pristupna“ strana

Druga strana predstavlja onu preko koje se dodaju, modifikuju ili brišu funkcije. Svaka dodata funkcija može nakon dodavanja da se izmeni, prepravi ili da joj se menja opis. Stranici za „Dodavanje“ se može pristupiti samo preko korisničkog imena i lozinke. Nakon uspešnog logovanja, prokazuje se glavna strana na kojoj su ponuđene opcije za dodavanje funkcija, izmenu funkcija, brisanje funkcija i opcija za dodavanje i brisanje korisnika koji mogu ovoj strani da pristupe. Izgled strane za „Dodavanje“ je prikazana na slici 2.2. Ukoliko „administrator“ ne želi da funkcija više bude prikazana, ona može biti permanentno obrisana. Ukoliko je potrebno da se nekoj funkciji iz baze izvrši neka promena, na primer ako je došlo do štamparske greške prilikom unosa, ili je potrebno modifikovati sam *kod* u fajlovima *.js* ili *.html* pritiskom na „izmeni funkciju“ i odabirom funkcije kojoj je potrebna izmena otvara se formular sa prethodno upisanim podacima koji se mogu menjati.



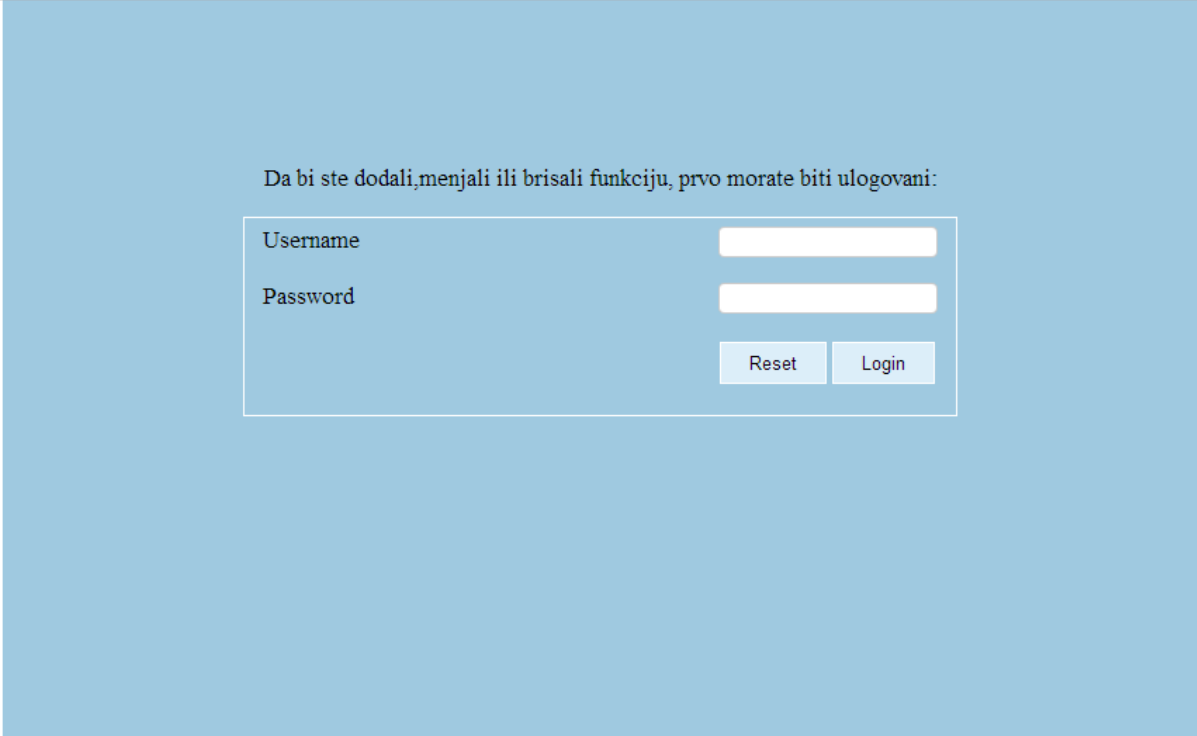
Slika 2.2 Izgled početne strane za dodavanje, brisanje i modifikaciju funkcija

Jedan od glavnih celina za neophodno funcionisanje obe strane je folder „*uploads*“.Ovaj folder sadrži fajlove *.js* formata i fajlove *.html* formata. Iz fajla formata *.js* učitavamo JavaScript funkciju, a iz *.html* fajla učitavamo HTML kod koji je potreban da bi se funkcija izvršila.Prilikom unosa funkcije, otpremaju se fajlovi ekstenzija *.js* i *.html*pod imenima „Kod_js_ID.js“ i „Html_ID.html“ gde ID predstavlja brojnu vrednost rednog broja pod kojim je upisan u bazu.

U daljem tekstu će biti detaljnije opisan način izrade obe strane.

3.STRANA ZA DODAVANJE FUNKCIJA

Prilikom pristupa ovoj strani potrebno je da se korisnik prvo prijavi unosom odgovarajuće lozinke i odgovarajuće šifre (slika 3.1).



Da bi ste dodali,menjali ili brisali funkciju, prvo morate biti ulogovani:

Username

Password

Reset Login

Slika 3.1 Strana za pristup/logovanje

Nakon uspešne prijave, korisnik se preusmerava na glavnu stranu (slika 2.2) sa opcijama:

- Dodaj funkciju
- Izmeni funkciju
- Briši funkciju
- Dodaj/briši korisnika
- Logout

3.1. Dodaj funkciju

Klikom na link „Dodaj funkciju“, otvara se strana sa formularom preko kojeg se vrši dodavanje funkcije (slika 3.1.1). U formular se opisuje naziv funkcije, grupa kojoj pripada, zatim se upisuje opis i dodaju fajlovi ekstenzije (.js) koji sadrži samu *JavaScript* funkciju koja se dodaje i fajl ekstenzije (.html) koji sadrži (X)HTML kod pomoću kojeg se *JavaScript* funkcija poziva. Pritiskom na dugme „Dodaj“ poziva se *php* fajl koji upisuje sadržaj funkcije u bazu (**javascript**) u tabelu (**primeri**). Ilustracija baze je prikazana na slikama 3.1.2 i 3.1.3. Kod *php* fajla(*dodaj_f.php*) je prikazan u priložima pod rednim brojem 1.

HOME

Dodavanje Javascript funkcije:

Naziv funkcije:

Funkcija pripada grupi:

Opis Javascript funkcije koja se dodaje:

Opis funkcije:

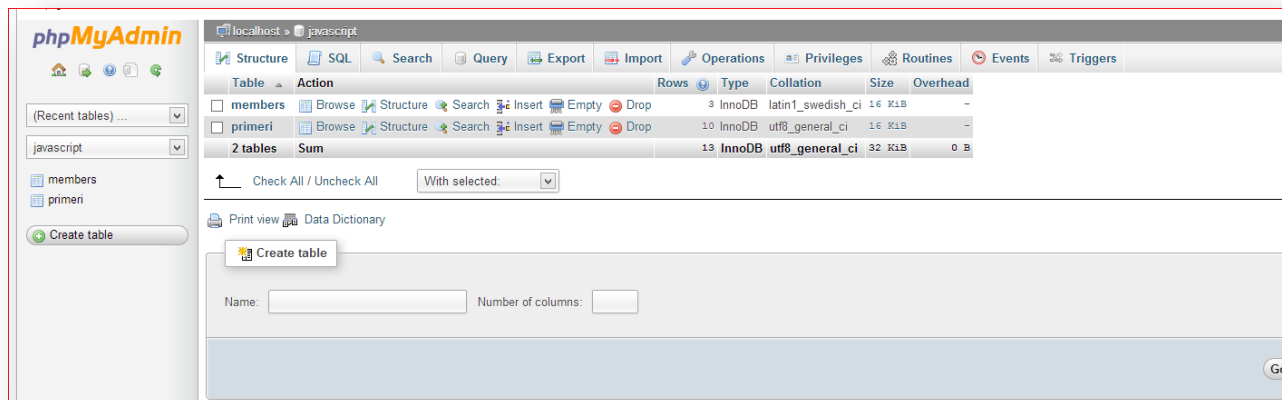
Funkcija se dodaje tako što se upload-uje fajl ekstenzije .js :

Fajl (.js): No file chosen

Dodajte html kod koji je potreban da bi se funkcija izvršila/pozvala:

Fajl (.html): No file chosen

Slika 3.1.1 Dodavanje funkcije



Slika 3.1.2 Baza my_sqli sa tabelama members i primeri

Baza **javascript** sadrži tabele „members“ i „primeri“. Tabela „members“ sadrži kolone *ID*, *username* i *password*. Ova tabela je neophodna za pristup stranici za dodavanje i izmenu funkcija. Kada administrator pokušava da pristupi stranici za dodavanje, prvobitno mora da se prijavi tako što unosi korisničko ime i lozinku. Pomoću odgovarajućeg *php* koda se proveravada li korisničko ime i lozinka postoje u tabeli i ako je odgovor potvrđan startuje se sesija i korisnik se preusmeravana glavnu stranu (slika 3.1). Ukoliko odgovor nije potvrđan, korisnik se ponovo vraća na stranicu za pristup radi novog pokušaja unosa ispravnog korisničkog imena i šifre. Kod za proveru korisničkog imena i šifre prikazan je u priložima pod rednim brojem 2.

Tabela „primeri“ sadrži kolone *ID*, *naziv*, *opis*, *primer*, *html* i *grupa*. Prilikom dodavanja funkcije ime funkcije se upisuje u kolonu *naziv*, opis funkcije u kolonu *opis*. U kolonama *primer* i *html* upisuju se imena *.js* i *.html* fajlova respektivno, dok u koloni *grupa* se upisuje kojoj grupi funkcija koja se dodaje pripada. *ID* kolona predstavlja identifikaciju svakog reda u tabeli i u vidu je *auto_increment* broja.

Showing rows 0 - 9 (~10 total) , Query took 0.0009 sec

```
SELECT *
FROM 'primeri'
LIMIT 0, 30
```

Profiling [Inline] [Edit] [Explain SQL] [Create PHP Code] [Refresh]

Show: Start row: 0 Number of rows: 30 Headers every 100 rows

Sort by key: None

ID	naziv	opis	primer	html	grupa
59	getElementById (backgroundColor)	Pristupa elementu koji se manipuliše preko svoj i...	Kod_js_59.js	Html_59.html	Metode Dom objekata
60	getElementById (fontSize)	Pristupa elementu koji se manipuliše preko svoj i...	Kod_js_60.js	Html_60.html	Metode Dom objekata
61	Event (x i y koordinata)	Objekat Event se generiše automatski kada god se ...	Kod_js_61.js	Html_61.html	Objekat Event
62	Metode objekta string-toLowerCase()	String je objekat koji sadrži niz znakova. Konstr...	Kod_js_62.js	Html_62.html	Osobine i metodi objekata string
63	Metode objekta string-toUpperCase()	String je objekat koji sadrži niz znakova. Konstr...	Kod_js_63.js	Html_63.html	Osobine i metodi objekata string
64	Metode objekta string- search.replace	Metodi search, match, replace i slični vraćaju s...	Kod_js_64.js	Html_64.html	Osobine i metodi objekata string
65	Provera ispravnosti telefonskog broja	Uzorak je niz znakova i specijalnih znakova. Konst...	Kod_js_65.js	Html_65.html	RegExp-Uzorak
66	Provera reči	Uzorak je niz znakova i specijalnih znakova. Konst...	Kod_js_66.js	Html_66.html	RegExp-Uzorak
67	Metoda objekta Date - getTime	Datum je objekat koji nosi informaciju o datumu i ...	Kod_js_67.js	Html_67.html	Datum
68	Datum i vreme	Datum je objekat koji nosi informaciju o datumu i ...	Kod_js_68.js	Html_68.html	Datum

Check All / Uncheck All With selected: Change Delete Export

Show: Start row: 0 Number of rows: 30 Headers every 100 rows

http://localhost/phpmyadmin/sql.php?db=javascript&token=976a142a67a8f9e2e74e43cc338a3&goto=db_structure.php&table=primeri&pos=0 Sun Oct 08 2013 19:32:38 GMT+0200 (Central Europe Daylight Time)

Slika 3.1.3 Tabela “primeri” u bazi javascript

3.1.1. Fajlovi (.js/.html)

Fajl (.js) treba da sadrži samo *JavaScript* funkciju koja predstavlja primer koji želimo da prikažemo na „*Pristupnoj*“ stranici. HTML fajl ekstenzije (.html) treba da sadrži elemente pomoću kojih se poziva funkcija i elemente neophodni za prikazivanje željenog rezultata. Na primer, ukoliko je ideja da se pomoću dugmeta pozove *JavaScript* funkcija, koja će da promeni veličinu slova nekom tekstu koji se nalazi u paragrafu, *html* fajl treba da sadrži `<button>` tag i `<p>` tag koji će sadržati neki tekst kojim se manipuliše.

Primer:

Kod JS funkcije:

Funkcija koja se poziva preko *id*-a postavlja veličinu slova određenom *tag*-u, u ovom slučaju tagu kome je *id*=„element“, postavlja veličinu slova na 25px.

```
function menjajVelicinuSlova(){
document.getElementById('element').style.fontSize="25px";
}
```

HTML kod:

Html fajl sadrži elemente pomoću kojih se poziva javascript funkcija, u ovom slučaju to je tag `<button>`. Tag `<pid="element">` sadrži tekst kojim se manipuliše, a oba tag-a su sadržana u jednom `<div>` tag-u kome po želji određujemo stil da bi se na strani za prikazivanje što lepše prikazalo korisniku.

```
<div style="width:90%;min-height:200px;text-align:center;padding-top:25px;">
<button onclick="menjajVelicinuSlova()">Manjaj Veličinu</button>
<br><br><br>
<pid="element" style="font-size:15px;">Ova slova će postati veća klikom na dugme
"Manjaj Veličinu"</p>
</div>
```

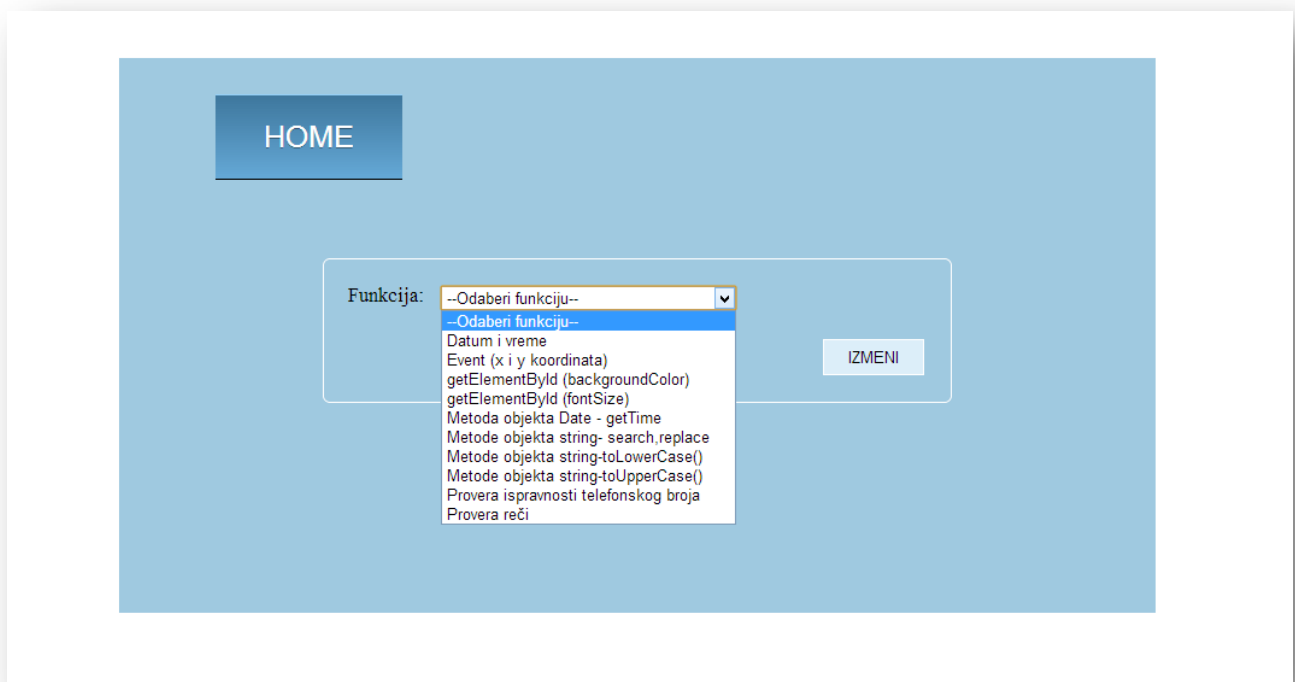
3.1.2. Otpremanje i premeštanje fajlova

Fajlovi koji se unose putem forme se nakon poziva php fajla preimenuju u „Kod_js_ID.js“, gde ID predstavlja *ID* iz baze, i u „Htm_ID.html“, i premeštaju se u prethodno spomenut folder „uploads“, a imena fajlova se smeštaju u kolonama tabele „primeri“, primer i html respektivno. Php kod koji izvršava otpremanje i premeštanje fajlova u željeni direktorijum je sledeći:

```
if($_FILES['fajl']['name'])
{
    //ukoliko nema grešaka...
    if(!$FILES['fajl']['error'])
    {
        //tačna putanja fajla do željene destinacije sa novim imenom
        $destination1=$dir."/uploads/" . $ime_fajla;
        //premeštanje otpremljenog fajla na željenu destinaciju
        move_uploaded_file($_FILES['fajl']['tmp_name'],$destination1);
    }
}
```

3.2. Izmeni funkciju

Klikom na izmeni funkciju otvara se strana preko koje se odabira funkcija koja se izmenjuje (slika 3.2.1). Funkcija se „izvlači“ iz baze pomoću svog ID-a.



Slika 3.2.1 Izbor funkcije za izmenu

Nakon odabira funkcije, otvara se formular sličan formularu za dodavanje funkcije. Ukoliko se menja sadržaj, izvrše se promene i klikom na dugme „Izmeni“ upisuju se izmene koje su načinjene. Slika 3.2.2:

HOME

Editovanje Javascript funkcije:

Naziv funkcije:

Funkcija pripada grupi:

Opis Javascript funkcije koja se dodaje:

Opis funkcije:

Funkcija se dodaje tako što se upload-uje fajl ekstenzije .js :

Kod JavaScript funkcije:

```
function show_coords(event)
{
    var x=event.clientX;
    var y=event.clientY;

    alert("X coords: " + x + ", Y coords: " + y);
}
```

Dodajte html kod koji je potreban da bi se funkcija izvršila pozvala:

Html kod:

```
<div onmousedown="show_coords(event)" style="width:90%;min-height:200px;text-align:center;padding-top:25px;">
<br><br><br>
<p id="element" >Click this area, and an alert box will alert the x and the y coordinates of the mouse pointer.</p>
</div>
```

Izmeni

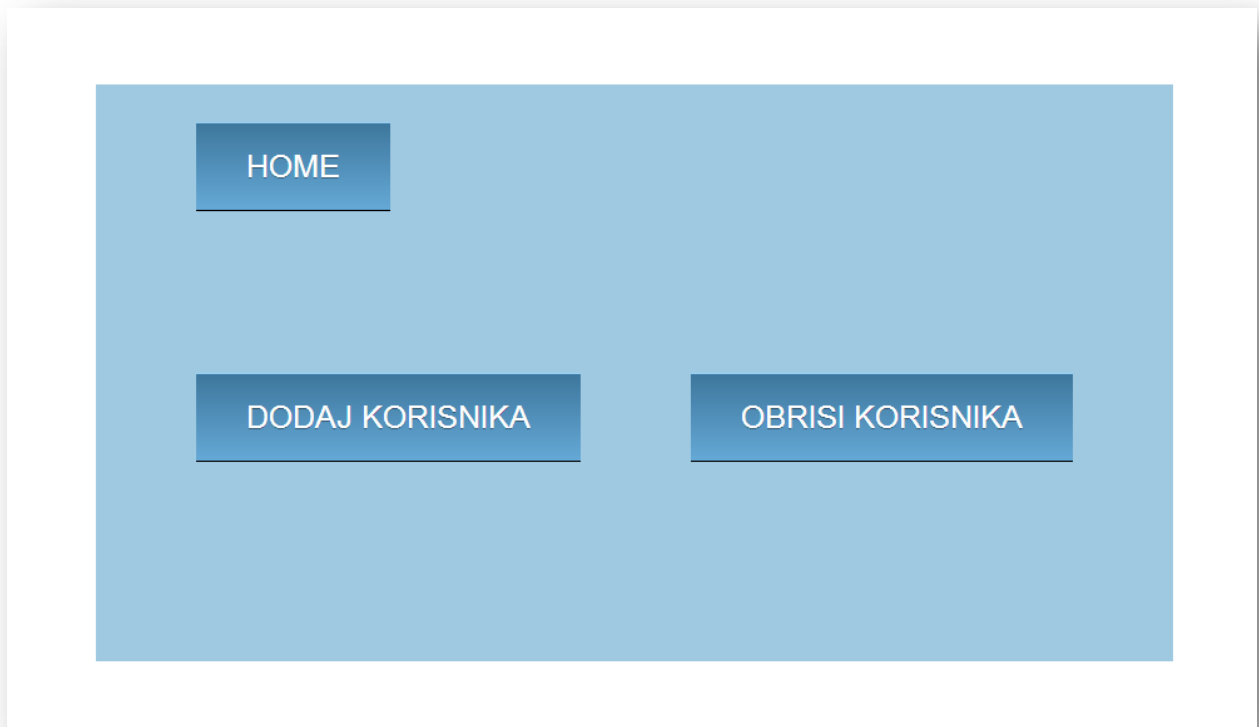
Slika 3.2.2 Izmena funkcije

3.3. Brisanje funkcije

Brisanje funkcije je omogućeno klikom na link sa početne strane „Brisanje funkcije“(slika 2.2). Otvara se strana za odabir funkcije koja se briše. Klikom na dugme briši poziva se *php* fajl „*delete.php*“ koji briše podatke iz baze i briše fajlove iz „*uploads*“ direktorijuma. Kod *php* fajla za brisanje prikazan je u poglavlju „Prilozi“ pod rednim brojem 3.

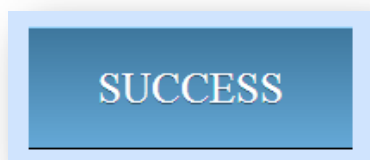
3.4. Dodavanje/brisanje korisnika

Svaki korisnik koji ima pristup ovoj stranici, može dodavati nove korisnike koji mogu da pristupe stranici. Takođe se može i obrisati korisnik, tj. oduzeti mu administratorska prava. Klikom na „Dodaj/briši korisnika“otvara se strana na kojoj se bira akcija koja je željena(slika 3.4.1).



Slika 3.4.1 Strana dodaj/briši korisnika

Posle svake izvršene operacije pojaviće se poruke „uspeha“ odnosno „neuspeha“ u vidu poruke *SUCCEESS* odnosno slike koja ukazuje da operacija nije uspešno izvršena (slike 3.4.2 i 3.4.3 respektivno).



Slika 3.4.2 Success



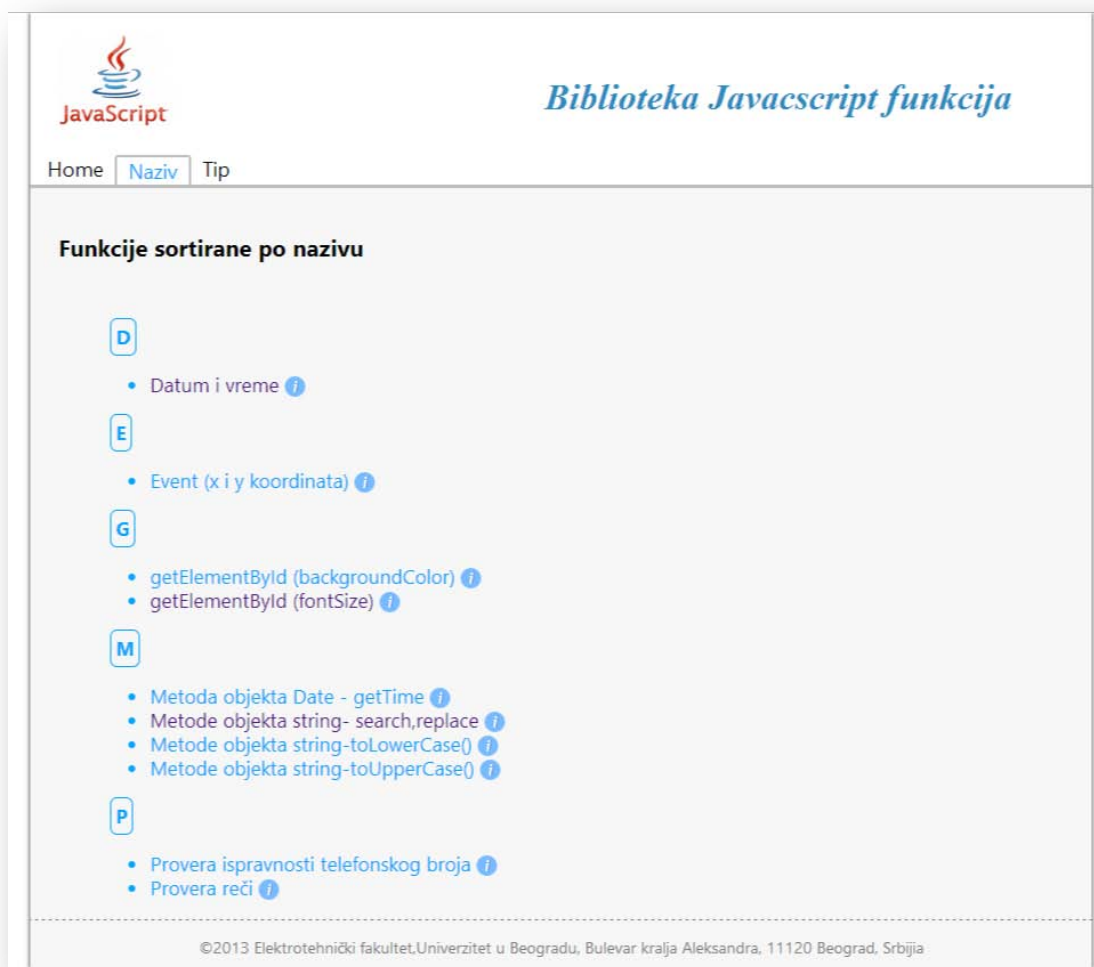
Slika 3.4.3 Neuspeh

4.,,PRISTUPNA “ STRANA

Ova strana je namenjena studentima, odnosno krajnim korisnicama i služi za prikazivanje funkcija. Glavna strana sadrži horizontalni meni sa linkovima ka stranama *Naziv* i *Tip* (slika 2.1). Na početnoj strani, odnosno *home*, stoji poruka za dobrodošlicu kao i mali opis strane i kako ona funkcioniše. U gornjem levom uglu se nalazi slika koja simbolizuje *JavaScript*, a klikom na nju sa bilo koje strane vrši se povratak na početnu stranu.

4.1. Strana *Naziv*

Na ovoj strani izlistane su funkcije i sortirane po abecednom redu (slika 4.1.1).



Slika 4.1.1 Strana *Naziv*

Ispod svakog slova abecede prikazane su funkcije čija imena počinju tim slovom. Pored svakog imena stoji oznaka „i“. Ako korisnik pređe mišem preko nje, pojaviće se u jednom span-u opis date funkcije, te je korisnicima omogućeno lakše i brže pronalaženje željene funkcije, odnosno čitanje njenog opisa (ilustracija na slici 4.1.2).



Slika 4.1.2 Opis

Klikom na bilo koju funkciju otvara se nova strana „primer“. Ta strana sadži opis funkcije, prikazana je grupa kojoj funkcija pripada. Takodje prikazan je njen *kod*, i praktična primena, gde korisnik može da vidi šta zaista funkcija odrađuje. Detaljan opis ove strane biće naveden u daljem tekstu.

4.2. Strana Tip

Ova strana je slična prethodnoj strani, s'tim što su funkcije sortirane, odnosno grupisane, po njihovom tipu, odnosno grupama kojima pripadaju (slika 4.2.1).



The screenshot shows a web page titled "Biblioteka Javascript funkcija". At the top left is the JavaScript logo. Below it are navigation links: "Home", "Naziv", and "Tip" (which is highlighted). The main content area is titled "Funkcije sortirane po grupama kojima pripadaju". It lists several groups of functions, each in a blue-bordered box:

- Datum**
 - Metoda objekta Date - getTime ⓘ
 - Datum i vreme ⓘ
- Metode Dom objekata**
 - getElementById (backgroundColor) ⓘ
 - getElementById (fontSize) ⓘ
- Objekat Event**
 - Event (x i y koordinata) ⓘ
- Osobine i metodi objekata string**
 - Metode objekta string-toLowerCase() ⓘ
 - Metode objekta string-toUpperCase() ⓘ
 - Metode objekta string- search,replace ⓘ
- RegExp-Uzorak**
 - Provera ispravnosti telefonskog broja ⓘ
 - Provera reči ⓘ

At the bottom of the page, there is a footer: "©2013 Elektrotehnički fakultet, Univerzitet u Beogradu, Bulevar kralja Aleksandra, 11120 Beograd, Srbija".

Slika 4.2.1 Strana Tip

4.3. Strana „primer“

Ova strana je možda i od najveće važnosti. Sastoji se iz sledećih delova:

- Opis funkcije
- Funkcija pripada grupi
- JavaScript kod
- Primena JavaScript funkcije

Ovoj strani se može pristupiti klikom na bilo koju funkciju sa strana „Naziv“ i „Tip“. Nakon klika na link, preusmerava se korisnik na stranu „primer“ i uz link šalje se i *ID* funkcije iz baze.

Kada se otvori strana „primer“ pomoću metode `$_GET['id']` dobijamo *ID* funkcije iz linka pomoću koje znamo o kojoj se funkciji radi, i koji će podaci biti ispisani. Tako se u bazi selektuje red čiji je *ID* onajdobijen pomoću `$_GET` metode. Na taj način iščitavamo opis funkcije i grupu kojoj pripada. Takođe pomoću *ID-a* znamo i koje *.js* i *.html* fajlove treba da iščitamo iz foldera „uploads“ jer se na osnovu *ID-a* i formiraju imena ovih fajlova. Izgled stranice primer prikazan je na slici 4.3.1.

Svaki deo o datoj funkciji se predstavlja u zasebnom delu. Svaki deo predstavlja jedan `<div>` koji imaju isti *css property*. Svaki od njih se popunjava pomoću *php* funkcija. Tako se *opis i grupa* iščitavaju iz baze, dok se za ispis JavaScript koda i HTML elemenata za deo u kome se implementiraju primene funkcija iščitavaju iz *.js* i *.html* fajlova. Svaki fajl se pozove i otvori, zatim se iščitava i ispisuje red po red sve dok ne dođe do kraja dokumenta.

Primer ovakvog koda koji je prethodno opisan (za *.js* fajlove):

```
<?php
$f=fopen("../uploads/". $primer. ".","r");//otvara željeni dokument
while(!feof($f)){// sve dok nije kraj fajla
$line_of_text= fgets($f);//čita liniju
print$line_of_text;}//zatim tu liniju ispisuje
fclose($f);

?>
```

Kompletan kod ove strane nalazi se u priložima pod rednim brojem 4.

Primer:

Metode objekta string-toUpperCase()

Opis funkcije:

String je objekat koji sadrži niz znakova. Konstruktor objekta string je STRING, međutim on ne mora da se koristi. U ovom primeru metoda toUpperCase() vraća string u velika slova.

Funkcija pripada grupi:

Osobine i metodi objekata string

Kod:

```
function uper()
{
    var imeStringa=document.getElementById('element').innerHTML;
    var ime=imeStringa.toUpperCase();
    document.getElementById('element').innerHTML=ime;
}
```

Primena javascript funkcije:

Povećaj slova

Klikom na DUGME sva slova ĆE POSTATI velika.

Slika 4.3.1 Stranica „primer“

5. ZAKLJUČAK

Ova platforma predstavlja skelet jednoj većoj platformi, koja će kasnije sadržati veliki broj primera *JavaScript* funkcija i koja bi predstavljala ozbiljnu platformu za učenje i unapređivanje programerskih veština. Sama web strana je jednostavna, lako čitljiva, te je i jako laka za korišćenje.

Prednost za unapređivanje je to što nisu potrebne velike veštine za dodavanje funkcija i nije potrebno puno vremena za povećanje baze funkcija. Funkcije se lako dodaju, lako izmenjuju a u samom procesu unapređivanja može da učestvuje veliki broj korisnika kojima bi bilo omogućeno da pristupe strani za „dodavanje“ funkcija.

6. PRILOZI

Kod 1. *dodaj_f.php*

```
<?php
if(isset($_POST["Ime"])){
    $ime=$_POST["Ime"];}
//Opis
if(isset($_POST["opis"])){
    $op=$_POST["opis"];}

//GRUPA
if(isset($_POST["grupa"])){
    $grupa=$_POST["grupa"];}

if($ime){
    $con=mysqli_connect("localhost","root","","javascript");
    // Check connection
    if(mysqli_connect_errno())
    {
        echo"Failed to connect to MySQL: ". mysqli_connect_error();
    }
    // dadavanje vesti
    $rez=mysqli_query($con,"INSERT INTO primeri(naziv,opis,grupa)
VALUES('$ime','$op','$grupa')");

    $result= mysqli_query($con,"SELECT * FROM primeri ORDER BY ID DESC Limit 1");
    while($row= mysqli_fetch_array($result)){ $side=$row['ID'];
    $ime_fajla="Kod_js_" . $side . ".js";
    $ime_html="Html_" . $side . ".html";

    $dir=dirname(dirname(dirname(__FILE__)));
    //Upload .js fajla u uploads folder
    if($_FILES['fajl']['name'])
    { //if no errors...
    if(!$_FILES['fajl']['error'])
    {
        //now is the time to modify the future file name and validate the file
        // $new_file_name = strtolower($_FILES['fajl']['tmp_name']); //rename file
        $destination1=$dir."/uploads/" . $ime_fajla; //tacna putanja fajla do zeljene
        destinacije]
        //sa novim imenom
        move_uploaded_file($_FILES['fajl']['tmp_name'],$destination1);
    }
    }
    //Upload .html fajla u uploads folder
```



```

mysql_connect("$host", "$username", "$password") or die("cannot connect");
mysql_select_db("$db_name") or die("cannot select DB");

// username and password sent from form
if(isset($_POST["myusername"])){
    $myusername=$_POST["myusername"];
} else{echo "lose";}
if(isset($_POST["mypassword"])){
    $mypassword=$_POST["mypassword"];
}

// To protect MySQL injection (more detail about MySQL injection)
$myusername=stripslashes($myusername);
$mypassword=stripslashes($mypassword);
$myusername=mysql_real_escape_string($myusername);
$mypassword=mysql_real_escape_string($mypassword);
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
password='$mypassword' ";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row
if($count==1){

// Register $myusername, $mypassword and redirect to file "login_success.php"
session_register("myusername");
session_register("mypassword");
header("location:login_success.php");
}
else{
echo "Wrong Username or Password";
header("refresh:2;url=../pages/login.php");
}
?>

```

Kod 3. *delete.php*

```
<?php
if(isset($_POST["function"])){
$id=$_POST["function"];}

$con=mysqli_connect("localhost","root","","javascript");
// Check connection
if(mysqli_connect_errno())
{
echo"Failed to connect to MySQL: ". mysqli_connect_error();
}

$result= mysqli_query($con,"DELETE FROM primeri WHERE ID=".$id.");
if($result){echo"<div style='text-align: center;margin-top:20%;'><span
style='border-top: 1px solid #96d1f8;

background: #65a9d7;
background: -webkit-gradient(linear, left top, left bottom, from(#3e779d),
to(#65a9d7));
background: -webkit-linear-gradient(top, #3e779d, #65a9d7);
background: -moz-linear-gradient(top, #3e779d, #65a9d7);
background: -ms-linear-gradient(top, #3e779d, #65a9d7);
background: -o-linear-gradient(top, #3e779d, #65a9d7);
padding: 20px 40px;

                                -webkit-border-
radius: 0px;
                                -moz-border-radius:
0px;
border-radius: 0px;
                                -webkit-box-shadow:
rgba(0,0,0,1) 0 1px 0;
                                -moz-box-shadow:
rgba(0,0,0,1) 0 1px 0;
box-shadow: rgba(0,0,0,1) 0 1px 0;
text-shadow: rgba(0,0,0,.4) 0 1px 0;
color: white;
font-size: 25px;
font-family: 'Lucida Grande', Helvetica, Arial, Sans-Serif;
text-decoration: none;
vertical-align: middle;'>SUCCESS</span></div>";
unlink('../..../uploads/Kod_js_' . $id . '.js');
unlink('../..../uploads/Html_' . $id . '.html');
header("refresh:3;url=../pages/delete_list.php");
}
else{echo"<div style='text-align: center;margin-top:20%;'><img
src='../style/sad.png' alt='Sad'></div>";
header("refresh:3;url=../pages/delete_list.php");}

mysqli_close($con);
?>
```

Kod 4. *primer.php*

```
<?php$currentPath="../"?>
<?php
$con=mysqli_connect("localhost","root","","javascript");
// Check connection
if(mysqli_connect_errno())
{
echo"Failed to connect to MySQL: ".mysqli_connect_error();
}

$result=mysqli_query($con,"SELECT * FROM primeri
WHERE ID='".$_GET['id']."'");
$id=$_GET['id'];
while($row=mysqli_fetch_array($result)){
$naslov=$row["naziv"];
$opis=$row["opis"];
$primer=$row["primer"];
$html=$row["html"];
$grupa=$row["grupa"];
}

?>
<?phpinclude"../includes/menu.php"?>
<divclass="opis">
<br>
<h3>Primer:</h3>
<h4style='color:#759BB7;'><?phpecho$naslov?></h4>
<pstyle='font-size:16px;'>Opis funkcije:</p>
<divclass='opisf'>

<p><?phpecho$opis;
echo"<br>";
?></p>
</div>
<pstyle='font-size:16px;'>Funkcija pripada grupi:</p>
<divclass='opisf'style="min-height:30px;"><?phpecho$grupa?></div>
<divclass='funk'><pstyle='font-size:16px;'>Kod: </p>
<preclass="prettyprint ">
<?php
$f=fopen("../..../uploads/".$primer."","r");//otvara željeni dokument
while(!feof($f)){// sve dok nije kraj fajla
$line_of_text= fgets($f);//čita liniju
print$line_of_text;}//zatim tu liniju ispisuje
fclose($f);

?></pre>
</div>

<!--Uključivanje funkcije -->
<scripttype="text/javascript"src="<?phpecho"../..../uploads/Kod_js_"$id.".js"?>"
></script>

<!--=====-->

<pstyle='font-size:16px;'>Primena javascript funkcije:</p>

<divclass='opisf'>
```

```
<?php
$f=fopen("../..../uploads/".$html."","r");
while(!feof($f)){
$line_of_text= fgets($f);
echo$line_of_text;}
fclose($f);
?>
</div>

</div>
<?phpinclude"../includes/footer.php";?>
```

LITERATURA

- [1] Frank W. Zammetti, „Practical JavaScript, DOM Scripting, and AJAX Projects“,2007.
- [2] Nicholas C. Zakas, "Professional JavaScript™ for Web Developers",Wiley Publishing Inc., 2005.
- [3] Danny Goodmanwith Michael Morrison, "JavaScript® BibleSixth Edition", Wiley Publishing Inc.,2007.
- [4] <http://www.w3schools.com/js/default.asp>