

**VELEUČILIŠTE U RIJECI**

Deni Pešić

**IZGRADNJA SUSTAVA INTERNETSKOG FORUMA**

(završni rad)

Rijeka, 2014.



# **VELEUČILIŠTE U RIJECI**

Poslovni odjel  
Stručni studij Informatika

## **IZGRADNJA SUSTAVA INTERNETSKOG FORUMA**

(završni rad)

MENTOR

Vlatka Davidović, predavač

STUDENT

Deni Pešić

MBS: 2422036146/11

Rijeka, rujan 2014.

# **I Z J A V A**

Izjavljujem da sam završni rad pod naslovom IZGRADNJA SUSTAVA  
INTERNETSKOG FORUMA izradio samostalno pod nadzorom i uz stručnu  
pomoć mentora Vlatke Davidović, predavač .

Deni Pešić

---

(potpis studenta)

# SADRŽAJ

1.	UVOD .....	1
2.	INTERNETSKI FORUM.....	2
3.	OPIS SUSTAVA FORUMA .....	3
3.1.	GRUPE.....	3
3.2.	KATEGORIJE I POTKATEGORIJE .....	3
3.3.	TEME .....	4
3.4.	ODGOVORI.....	4
4.	TEHNOLOGIJA KORIŠTENA U RAZVOJU SUSTAVA FORUM .....	5
4.1.	HTML .....	5
4.2.	PHP .....	5
4.3.	MYSQL .....	7
4.4.	JAVASCRIPT .....	8
4.5.	JQUERY.....	9
4.6.	AJAX .....	10
5.	MODELIRANJE SUSTAVA FORUM.....	11
6.	IZGRADNJA SUSTAVA FORUM.....	13
6.1.	KREIRANJE BAZE PODATAKA.....	13
6.2.	STRUKTURA WEB SJEDIŠTA .....	14
6.3.	KATEGORIJE KORISNIKA .....	16
6.4.	KATEGORIJE TEMA .....	17
6.5.	DIZAJN STRANICA.....	18
6.6.	DINAMIČKI DIO SUSTAVA .....	23
6.6.1.	<i>Spajanje na bazu podataka pomoću PHP-a .....</i>	<i>24</i>
6.6.2.	<i>Registracija korisnika.....</i>	<i>25</i>
6.6.3.	<i>Prijava korisnika .....</i>	<i>31</i>
6.6.4.	<i>Profil .....</i>	<i>34</i>
6.6.4.1.	<i>Izmjena e-pošte i potpisa .....</i>	<i>37</i>
6.6.4.2.	<i>Izmjena profilne slike.....</i>	<i>37</i>
6.6.5.	<i>Unos kategorija .....</i>	<i>39</i>
6.6.6.	<i>Unos potkategorija .....</i>	<i>42</i>
6.6.7.	<i>Početna stranica .....</i>	<i>45</i>
6.6.8.	<i>Potkategorije.....</i>	<i>50</i>
6.6.9.	<i>Nova tema .....</i>	<i>58</i>
6.6.10.	<i>Tema .....</i>	<i>61</i>
6.6.11.	<i>Odgovor.....</i>	<i>71</i>
6.6.12.	<i>Moderiranje foruma .....</i>	<i>78</i>
6.6.12.1.	<i>Izmjena naslova .....</i>	<i>78</i>
6.6.12.2.	<i>Izmjena i brisanje.....</i>	<i>80</i>
6.6.12.3.	<i>Micanje teme.....</i>	<i>80</i>
6.6.12.4.	<i>Spajanje tema .....</i>	<i>82</i>
6.6.12.5.	<i>Pribadanje, sakrivanje i zaključavanje .....</i>	<i>84</i>
6.6.13.	<i>Administracija sustava .....</i>	<i>85</i>
6.6.13.1.	<i>Pregled i ažuriranje korisnika.....</i>	<i>86</i>
6.6.13.2.	<i>Pregled i ažuriranje kategorija i potkategorija.....</i>	<i>88</i>
6.6.13.3.	<i>Pregled i ažuriranje grupa.....</i>	<i>92</i>

<b>7. ZAKLJUČAK .....</b>	<b>95</b>
<b>LITERATURA .....</b>	<b>96</b>
<b>POPIS SLIKA.....</b>	<b>97</b>
<b>POPIS TABLICA .....</b>	<b>98</b>
<b>POPIS ISJEČAKA .....</b>	<b>98</b>
<b>POPIS KRATICA .....</b>	<b>100</b>

## **Sažetak**

Izgradnja sustava kvalitetnog, dinamičkog internetskog foruma nije jednostavan zadatak. Prilikom izrade potrebno je paziti na bilo kakve propuste koje bi kasnije korisnici foruma mogli iskoristiti za nepoželjnu izmjenu i prikaz podataka iz baze podataka, također je potrebno omogućiti jednostavnost i interaktivnost korištenja funkcija foruma. Cilj ovog rada je prikaz izrade interaktivnog i dinamičkog internetskog foruma koristeći HTML, JavaScript, jQuery JavaScript knjižnice, MySQL, Ajax i PHP tehnologije. Ove tehnologije omogućavaju rješavanje nekih od problema kao što su direktni prikaz utjecaja izmijenjenih podataka, validacija polja te neodobrena izmjena i prikaz podataka.

Ključne riječi: varijabla, skripta, forum, upit, baza.

## **1. Uvod**

Internetski forum, koji se još naziva i oglasna ploča, je mjesto na internetu gdje osobe mogu razgovarati u obliku objavljenih poruka uporabom web preglednika. Oni se razlikuju od chat soba u kojima članovi obično svi razgovaraju istovremeno. Kod foruma, članovi objavljuju poruke koje drugi članovi mogu čitati kad god pristupe forumu.

Sustav internetskog foruma može biti napisani pomoću različitih programskih jezika poput ASP, Java, Perl te PHP. Primjer sustava internetskog foruma u ovom radu izrađen je pomoću PHP programskog jezika koji je orijentiran po Perl i C sintaksi te prvenstveno namijenjen programiranju dinamičnih web stranica.

U ovom je radu obrađen internetski forum kao web aplikacija, objašnjena je sama struktura foruma te je opisan sustav foruma. Definirane su i objašnjene tehnologije korištene za izradu sustava Forum te su izrađeni modeli na kojima se temelji baza podataka sustava Forum i funkcije tog sustava. Objašnjen je cjelokupni postupak izrade sustava Forum, od kreiranja baze podataka, strukture web sjedišta, dizajna stranica te sama izrada dinamičkog dijela sustava Forum.



## **2. Internetski forum**

Prvi internetski forumi datiraju još iz 1990-ih godina. Internetski forum je zapravo web stranica koja nudi online izmjenu informacija između ljudi o određenoj temi na način da pruža prostor za pitanja i dogovore. Osim samog teksta, na forumu sadržaj može biti u obliku slike, videa, datoteka i linkova.

Internetski forumi mogu biti potpuno anonimni ili je potrebna registracija s korisničkim imenom i lozinkom. Kod foruma koji imaju stalne korisnike, često se stvara osjećaj virtualne zajednice. Video igrice, tehnologija, politika, sport, religija i muzika popularne su teme foruma, ali postoje forumi i za razne druge teme.

### **3. Opis sustava foruma**

Forum je hijerarhijske strukture. Može sadržavati više kategorija te svaka od tih kategorija može sadržavati više potkategorija. Unutar svake potkategorije nalaze se diskusije koje se nazivaju teme. Na svaku temu može odgovoriti beskrajni broj korisnika koji spadaju u grupu članova foruma. Temama upravlja grupa moderatora foruma koji paze da je sadržaj primijeren te da se sudionici diskusije drže teme. Također i postoji administrator foruma koji upravlja cijelim sustavom foruma na način da kreira kategorije, potkategorije te određuje moderatore, pravila, izgled i način funkcioniranja foruma.

#### **3.1. Grupe**

Posjetitelji foruma svrstavaju se u grupe. Te grupe mogu imati različita prava i privilegije. Uobičajeno je da neregistrirani članovi mogu samo pregledavati forume, dok registrirani članovi mogu još i stvarati nove teme i odgovarati na postojeće. Administratori, koji imaju potpunu kontrolu foruma, mogu promovirati registrirane korisnike u moderatore te im time omogućiti moderaciju foruma.

#### **3.2. Kategorije i potkategorije**

Kako bi internetski forum bio uredan i organiziran, mora imati definirane kategorije koje će članovima pomoći pronaći informacije koje traže te minimizirati zbunjenost. Posjetitelja ne bi trebalo zasuti brojnim kategorijama već te kategorije dalje filtrirati pomoću potkategorija. Unutar tih potkategorija će posjetitelji pronaći teme koje su relevantne za njih te će lakše sudjelovati u tim temama.

### **3.3.Teme**

Teme su diskusije koje korisnici foruma kreiraju te se nalaze unutar odabranih potkategorija. Svaka tema sadrži naslov, sadržaj, datum kreiranja te informacije o korisniku koji je započeo temu. Korisnici mogu ostavljati svoje odgovore na temu. Odgovori su uobičajeno poredani od najstarijeg do najnovijeg. Broj odgovora nema granica te svaki korisnik može ostaviti koliko god želi odgovora pa čak i jedan iza drugog.

### **3.4.Odgovori**

Odgovor je sadržaj koji je stvorio neki korisnik unutar neke teme. Sadrži datum kreiranja, podatke o korisniku te sam sadržaj odgovora. Korisnik koji je kreirao odgovor, uobičajeno ima mogućnost izmjene ili brisanja sadržaja. Ukoliko korisnik ima tzv. potpis, prikazuje se na dnu odgovora te sadrži kratki tekst, sliku ili hipervezu.

## **4. Tehnologija korištena u razvoju sustava Forum**

Sustav Forum izgrađen je kao web aplikacija te je baziran na slijedećim tehnologijama: HTML, PHP, MySQL, JavaScript, JQuery i Ajax.

### **4.1. HTML**

HTML je skraćenica za HyperText Markup Language te služi za označavanje hipertekstualnih dokumenata. Jednostavnije, to je jezik koji opisuje web stranice koristeći običan tekst. Jezik nije kompleksan.

Svaka se web stranica sastoji od HTML datoteka, a svaka je HTML datoteka obična tekstualna datoteka sa .html ekstenzijom te se sastoji od mnogo HTML oznaka (eng. tag) kao i sadržaj same web stranice. Navigacija između datoteka se vrši preko hiperveza.

HTML oznake su skrivene ključne riječi unutar web stranice koje definiraju kako će web preglednik formatirati i prikazati sadržaj. Oznake imaju unaprijed odabrana imena.

Većina oznaka sastoji se od dva dijela, dio koji otvara i dio koji zatvara oznaku. Jedan primjer je `< i >` koji otvara i `</i>` koji zatvara. U ovom slučaju, tekst između početnog i završnog dijela biti će nakošen. Neke su oznake iznimke, poput `<img>` oznake koja prikazuje sliku te ju nije potrebno zatvoriti jer se sadržaj navodi kao atribut, a ne kao tekst između dva dijela oznake kao u slučaju `<i>` oznake.

Atributi nam dozvoljavaju prilagođavanje oznaka te ih se definira u prvom dijelu oznake. Atributi se obično definiraju na način da se napiše naziv atributa poput 'width' te mu se dodijeli vrijednost koristeći znak za jednako i navodnici, npr. `'width="100px"'`.

### **4.2. PHP**

"Razvoj PHP-a započeo je 1994. godine kada je programer Rasmus Lerdorf napisao seriju CGI Perl skripti pomoću kojih je održavao svoju osobnu web stranicu. Za pisanje skripti koristio

je C programski jezik pomoću kojeg ih je i proširio kako bi imale mogućnost rada sa web obrascima i mogućnost komuniciranja s bazama podataka. Ovu je implementaciju nazvao "Personal Home Page/Forms Interpreter" ili PHP/FI.

PHP/FI mogao se koristiti za izgradnju jednostavnih dinamičkih web aplikacija. Lerdorf je 1995. godine objavio prvu verziju svog koda kako bi ubrzao pronalazak bugova. Eventualno se formirao tim programera te je započeo razvoj PHP/FI 2 koji je službeno izašao krajem 1997. godine.

Programeri Zeer Suraski i Andi Gutmans preradili su PHP/FI parser te time i promijenili ime jezika u PHP: Hypertext Preprocessor kakav poznajemo danas. PHP 3 je službeno izašao 1998. godine. Također su i preradili samu jezgru PHP-a te time stvorili Zend Engine koji služi za interpretiranje PHP koda.

PHP 4 izlazi 2000. godine te koristi Zend Engine 1.0. Sredinom 2004. godine izašao je PHP 5. Ova verzija sadrži podršku za objektno-orijentirano programiranje te PHP Data Objects ekstenziju. PHP 5 je jedina stabilna verzija koja se još uvijek razvija“ (The PHP Group, 2008.)[3]

PHP je server-side skriptni jezik za dinamično generiranje HTML koda. Sličan je drugim server-side skriptnim jezicima koji omogućuju dinamično generiranje sadržaja na web serveru, poput ASP.NET i JavaServer Pages.

Pomoću PHP-a, HTML stranica se prvo generira na web serveru, popunjava se dinamičkim sadržajem te se šalje klijentu. Klijent ne može vidjeti PHP kod već mu se samo prikazuje HTML kod. PHP je po svojoj sintaksi sličan jezicima kao što su Perl i C te je danas jedan od najkorištenijih i najnaprednijih server-side skriptnih tehnologija.

PHP se može primijeniti na većini web poslužitelja, na mnogim operativnim sustavima i platformama. Također se može i koristiti sa mnogim relacijskim sustavima za upravljanje bazama podataka.

Prednost PHP jezika nad ostalim jezicima je ta što podržava veliki broj sustava za upravljanje bazama podataka, što je od velikog značaja jer sustav za upravljanje bazama podataka osigurava bogat broj funkcija za rad s njima.

Podržane baze podataka su:

*Tablica 1. PHP podržane baze podataka*

Adabas D	Ingres	Oracle (OCI 7 i OCI 8)
dBase	InterBase	Ovrimos
Empress	mSQL	Solid
PostgreSQL	FilePro(read-only)	Sybase
FrontBase	Hyperwave	Direct MS-SQL
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

Izvor: [http://leiron.be/infodocs/supported\\_database.html](http://leiron.be/infodocs/supported_database.html)[4]

Obzirom na podržane sustave za upravljanje bazama podataka, za sustav Forum, korišten je MySQL.

### 4.3. MySQL

Kako bi organizacija, prikupljanje i dohvat podataka bili jednostavniji, potreban je sustav za upravljanje podataka. „MySQL je jedan od najpopularnijih open-source sustava za upravljanje bazama podataka.“ (Solid IT, 2014.)[1] PHP i MySQL često se koriste zajedno, zbog toga je popularnost MySQL-a vezana za PHP. Tehnički gledano, baze podataka su dobro strukturirane zbirke podataka koji variraju od osnovnih do najsloženijih podataka, od običnih imena pa do ogromnih količina podataka za veće korporacije.

Karakteristika MySQL je da je relacijska baza podataka. Relacijske su baze podataka one baze podataka koje vrše pohranu podataka u odvojenim tablicama umjesto da ih spremaju sve na jedno mjesto. To znači da je pohrana i dohvat podataka učinkovita jer su podaci organizirani u specifičnim lokacijama.

Kako bi pohrana podataka u bazu bila moguća, prvo se mora kreirati baza podataka. U kreiranu bazu moguće je pohraniti velik broj tablica. Svaka tablica mora imati definirane stupce.

Svaki stupac služi za spremanje specifičnih podataka. Moguće je limitirati podatke koji se mogu unijeti u pojedini stupac, npr. samo numerički podaci ili samo određeni broj znakova.

Nakon što su podaci uneseni u bazu podataka, bivaju pohranjeni u odgovarajuću tablicu koja se onda sastoji od redaka i stupaca.

MySQL može upravljati s više baza podataka koje mogu sadržavati više tablica. Razine pristupa mogu se dodijeliti različitim korisnicima kako bi im se pružile ili ukinule određene privilegije.

MySQL koristi naredbe/funkcije bazirane na SQL (Structured Query Language). To nam omogućuje korištenje ključnih riječi kako bi naveli koje podatke želimo vratiti. MySQL implementira neke dodatne mogućnosti za poboljšanje funkcionalnosti. Ključne je riječi općenito lako za razumjeti a često se odnose na engleske riječi ili fraze (npr. SELECT, UPDATE, WHERE, INSERT).

#### **4.4. JavaScript**

JavaScript je objektno orijentiran skriptni jezik, tj. lagan programski jezik kojeg tumači web preglednik kada se učitava web stranica.

S obzirom da je web preglednik taj koji tumači JavaScript jezik, može doći do nekih nedosljednosti u načinu na koji će biti prikazana web stranica u različitim web preglednicima.

JavaScript radi u web pregledniku, to znači da ukoliko imamo stariji web preglednik, koji nema JavaScript podršku ili ako je JavaScript onemogućen, JavaScript kod neće raditi.

Za razliku od onoga što se događa s jezicima poput PHP-a koji se pokreću na serveru, s JavaScriptom nikada ne možemo biti sigurni kako će posjetiteljev web preglednik utjecati na naš kod i da li je JavaScript uopće omogućen na web pregledniku.

„JavaScript koristi takozvani „Same Origin Policy“. To znači da JavaScript kod sa neke druge domene ne može utjecati na sadržaj na našoj domeni.“ (W3C, 2010.)[5]

Kako je JavaScript jezik koji se izvodi na klijentskoj strani, ograničen je na ono što može učiniti i sam klijent, poput direktnog pristupa serveru i njegovim resursima kao što su baze podataka.

Jedna od najvećih prednosti JavaScripta je mogućnost da dohvati informacije o web pregledniku i HTML stranici. Ovakav pristup informacijama omogućava JavaScriptu veliku moć modificiranja iskustva prilikom pregledavanja web stranica.

JavaScript knjižnice poput jQuery-a postoje kako bi nam još više olakšale stvari koje možemo učiniti sa JavaScript-om. Osim jQuery-a postoje i druge JavaScript knjižnice, Prototype, Ext, Dojo Toolkit, MooTools, Script.aculo.us i drugi.

## 4.5. jQuery

jQuery nije poseban jezik, već samo dobro napisan JavaScript kod. Točnije, „jQuery je brza i sažeta JavaScript knjižnica koja pojednostavljuje rukovanje događajima, animiranje i Ajax interakcije za brzu izradu web stranica i aplikacija.“(The jQuery Foundation, 2010.)[2]

Kako bi mogli raditi sa jQuery-em, potrebno je znati barem osnove JavaScripta, HTML-a i CSS-a (Cascading Style Sheets).

jQuery povećava produktivnost programera jer mu omogućava da postigne kritičnu funkcionalnost korisničkog sučelja pisanjem vrlo male količine koda. Također pomaže u poboljšanju performansi web aplikacije, brz je, može se proširiti pomoću dodataka te mu se može prilagoditi ponašanje.

jQuery sintaksa dizajnirana je kako bi olakšala navigaciju u dokumentu, odabir DOM (Document Object Model) elemenata, kreiranje animacija, rukovanje događajima i razvoj Ajax aplikacija.



## 4.6. Ajax

AJAX je akronim za **A**synchronous **J**JavaScript and **X**ML (Extensible Markup Language). Ajax nije programski jezik ili alat. Sam Ajax je generički naziv za razne JavaScript tehnike korištene za dinamično spajanje na web server i bazu podataka te ažuriranje dijelova web stranice bez potrebe za osvježanjem web stranice. Glavna svrha Ajaxa je poboljšati brzinu, performanse i upotrebljivost web aplikacija.

Pomoću Ajaxa izvode se uzvratni pozivi koji velikom brzinom šalju, dohvaćaju i spremaju podatke na i sa servera. Zbog toga, iskorištenost mreže svedeno je na minimum te će brže doći do izvršenja operacije. Na mjestima s ograničenim bandwidthom to može uvelike poboljšati performansu mreže. Slanjem samo potrebnih podataka, ograničena je količina podataka koje treba biti obrađena na serveru. Nema potrebe obrade svih elemenata obrazca ili slanja cijele stranice nazad na klijenta.

Riječ „asinkroni“ u nazivu Ajax-a označava asinkroni način obavljanja poziva na web server. Više se događaja izvršava neovisno jedan o drugome. Nakon što klijent pokrene Ajax povratni poziv na server, neće morati čekati na odgovor te može nastaviti koristiti web aplikaciju dok se zahtjev obrađuje. Nakon završetka obrade, server šalje odgovor klijentu kojeg zatim klijent obrađuje po potrebi.

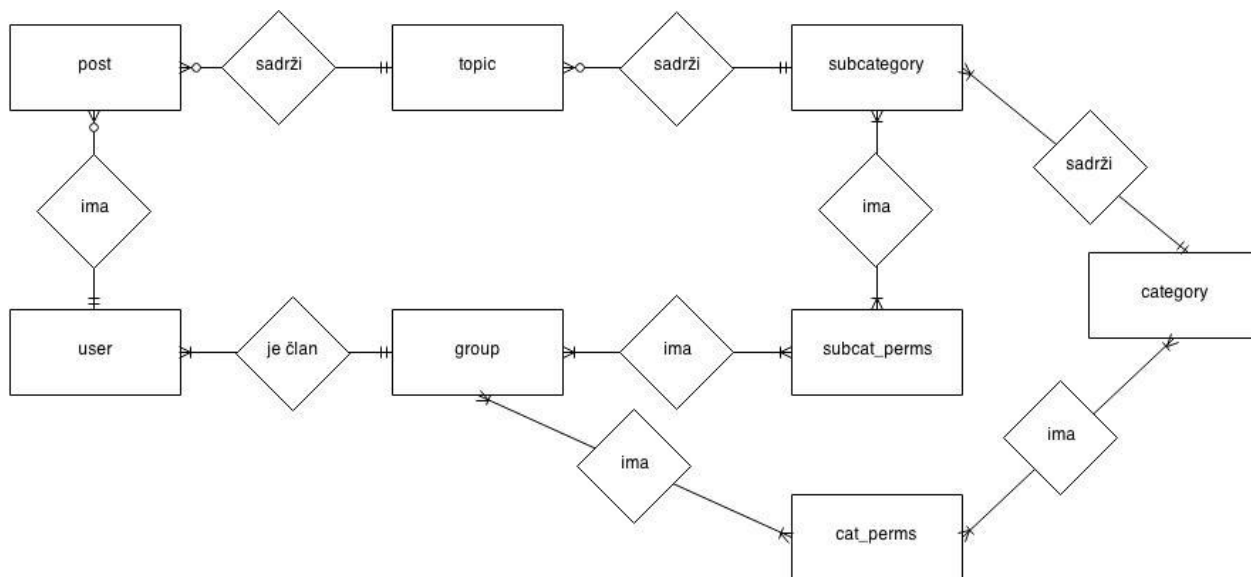
Programski jezik na strani klijenta je JavaScript, a XML je markup jezik za definiranje podataka. JSON (JavaScript Object Notation) je još jedan markup jezik koji se može koristiti za definiranje podataka. Puno je lakši za korištenje sa JavaScript-om nego XML.

Veliki napredak za JavaScript i Ajax je jQuery. Pomoću jQuery i JSON web service, Ajax uzvratni pozivi postali su standardne programske prakse za projektiranje i razvoj web aplikacija.

## 5. Modeliranje sustava Forum

Izradom modela sustava Forum dobiva se predodžba o samoj strukturi baze podataka te strukturi web sjedišta. ER modelom na apstraktan način utvrđuje se opis podataka i odnos između podataka te kasnije služi kao pomoć pri izradi relacijske baze podataka.

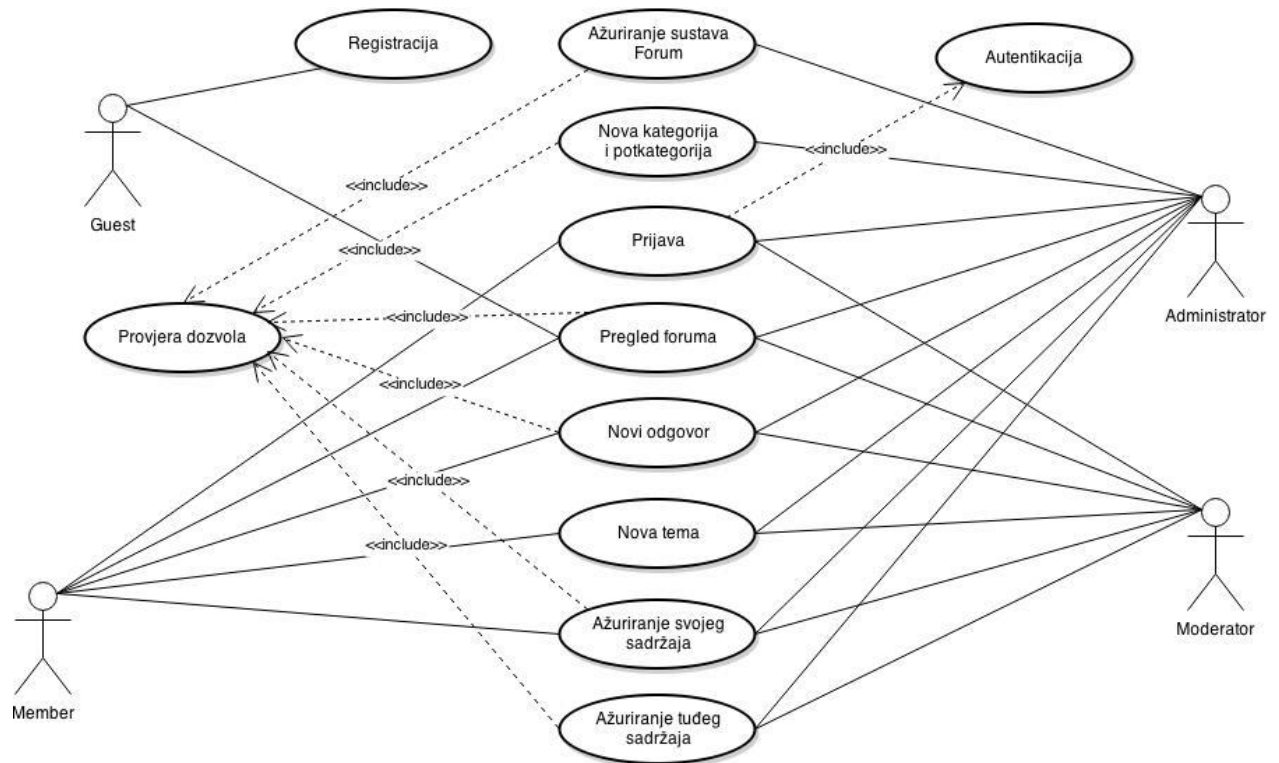
Slika 1. ER model



Izvor: Autor

Dijagramom slučaja korištenja prikazuju se različiti korisnici sustava te njihove interakcije sa sustavom.

Slika 2. Dijagram slučaja korištenja



Izvor: Autor

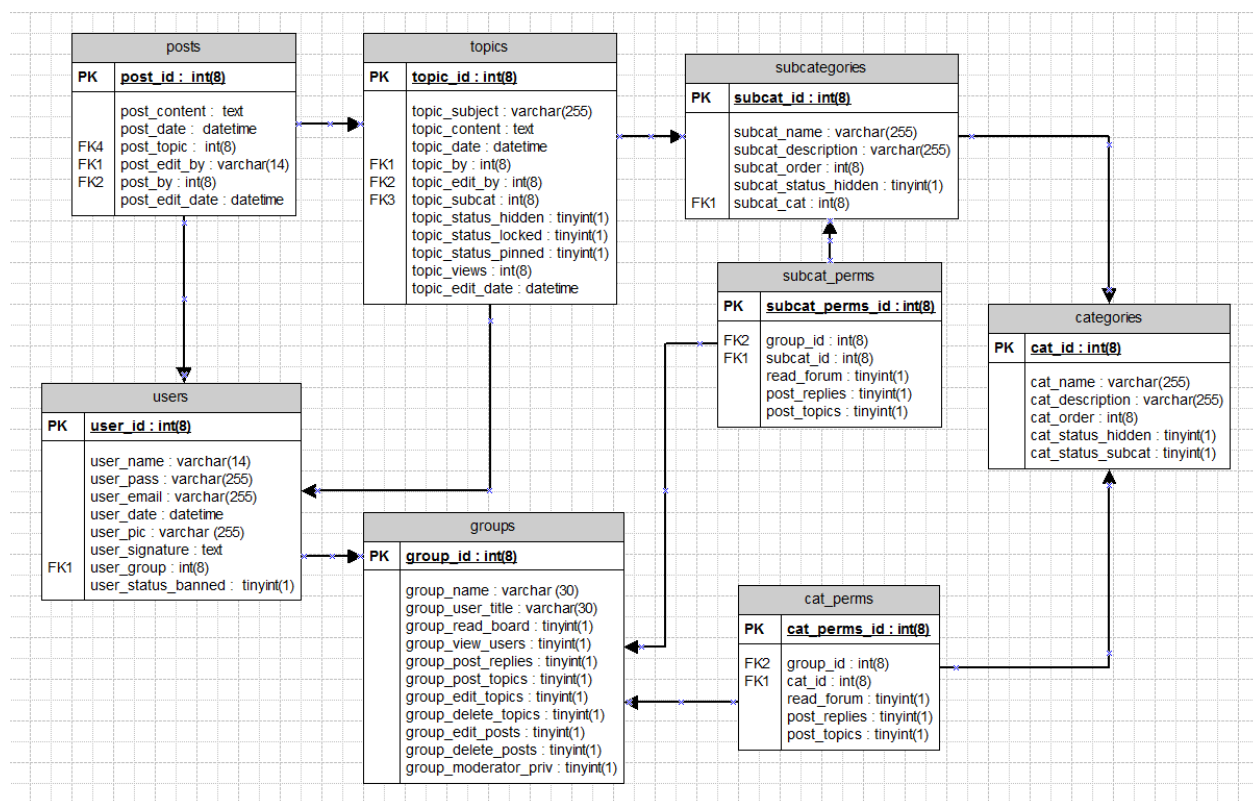
## 6. Izgradnja sustava Forum

Na temelju izrađenog modela sustava Forum, kreira se baza podataka te se određuje struktura web sjedišta. Također se definira dizajn web sjedišta i izrađuje se dinamički dio web aplikacije.

### 6.1. Kreiranje baze podataka

Kako bi izrađeni forum imao pristup podacima, potrebna je baza podataka gdje će svi podaci biti spremljeni. Na temelju ER modela sustava foruma izrađen je relacijski model baze podataka te je prema tom modelu kreirana sama baza podataka.

Slika 3: Relacijski model baze podataka

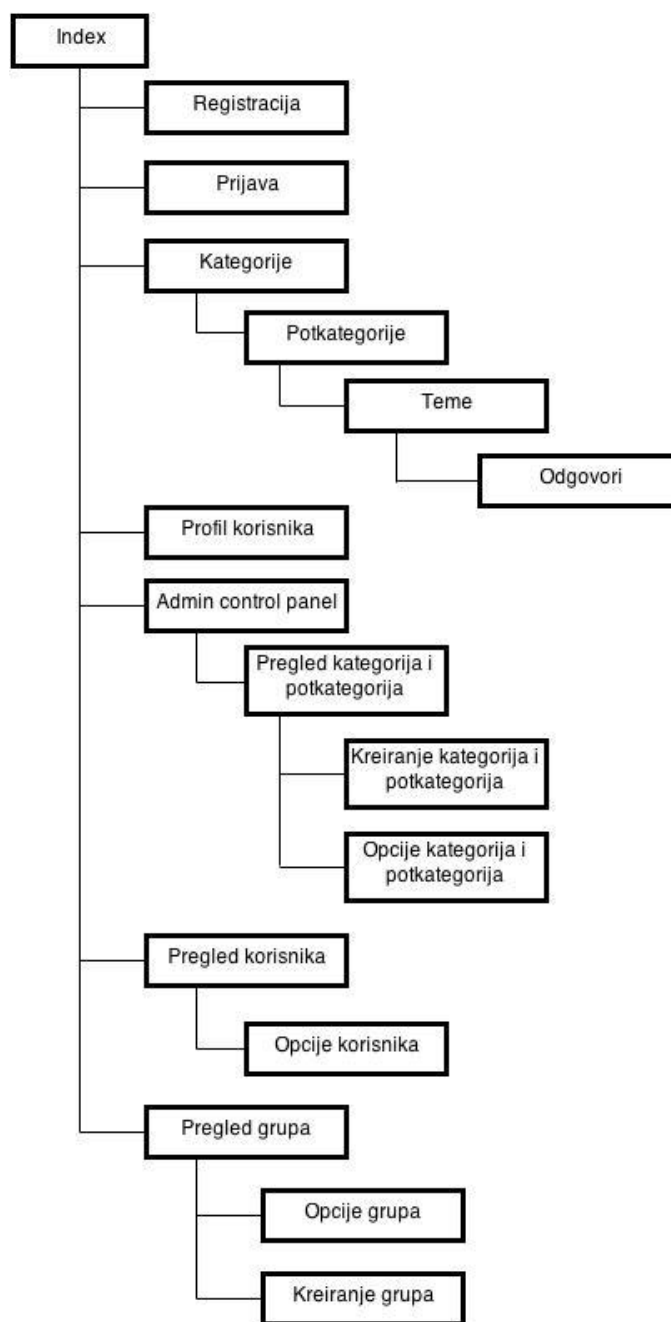


Izvor: Autor

## **6.2. Struktura web sjedišta**

Kako bi sustav Forum mogao pravilno funkcionirati, posjetitelji foruma moraju biti organizirani i smješteni u grupe sa različitim dozvolama i privilegijama. Stranice sa funkcijama web sjedišta moraju biti povezane na logičan način te diskusije unutar foruma moraju biti organizirane unutar kategorija i potkategorija.

Slika 4: Struktura web sjedišta

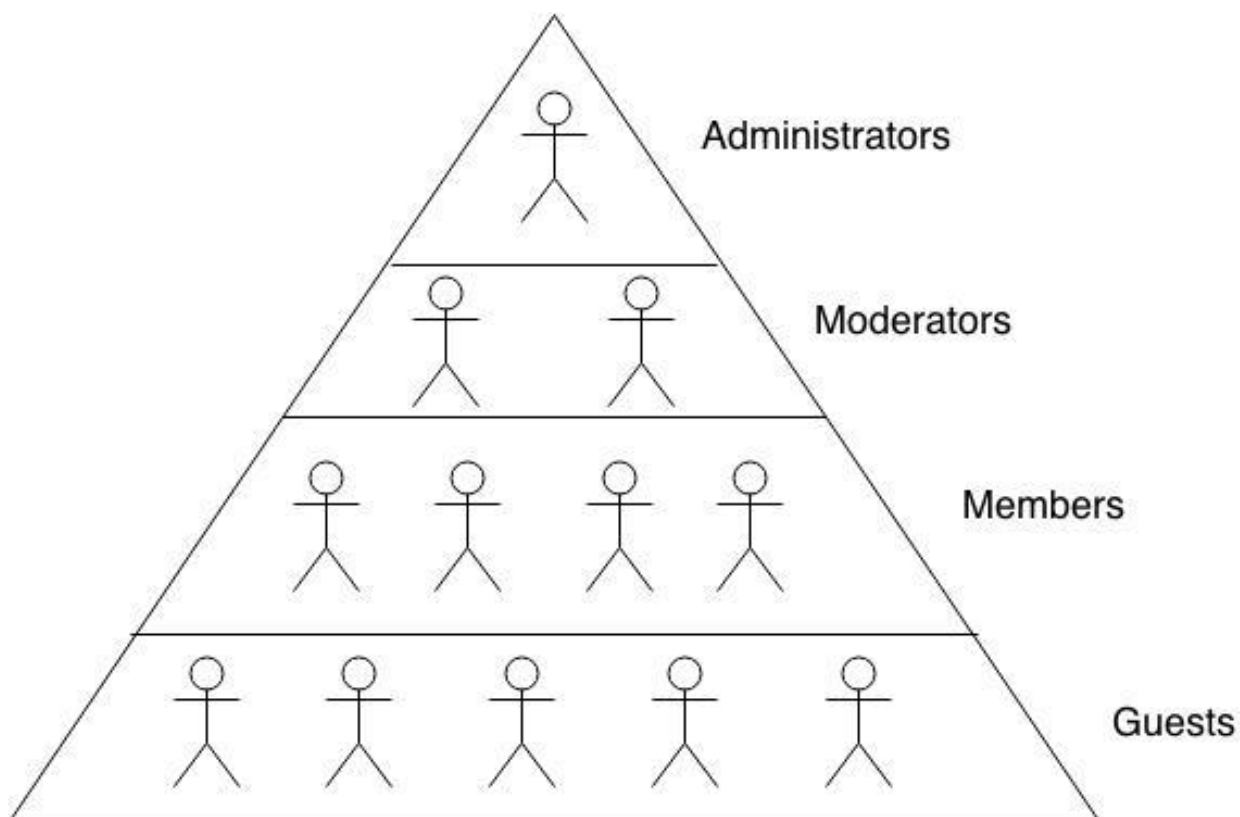


Izvor: Autor

### **6.3. Kategorije korisnika**

Samim dolaskom na web sjedište sustava Forum, posjetitelj je smješten u jednu od grupa sustava Forum, točnije u grupu „Guests“. Naime, grupe su hijerarhijske strukture piramide. Na samom vrhu piramide je „Administrators“ grupa. Obično je član te grupe samo jedan korisnik, administrator, koji ima potpunu kontrolu nad sustavom foruma, uključujući i kontrolu nad drugim grupama. Ispod administratora nalazi se „Moderators“ grupa čiji članovi imaju pravo ažuriranja sadržaja i članova korisnika grupe „Members“, koji su obični članovi foruma s limitiranim dozvolama, i nalaze se ispod moderatora. Članovi „Members“ grupe, ovisno o postavkama administratora, imaju samo pravo kontrole samo nad svojim sadržajem, stvari poput stvaranje novih tema, odgovora i ažuriranje istih. Kako bi posjetitelj postao član ove grupe, mora se prvo registrirati u sustav foruma. Na samom dnu piramide su ranije spomenuti članovi grupe „Guests“ koji samo imaju pravo pregleda foruma. Ukoliko posjetitelj nije registriran i prijavljen u sustav, smatra se članom ove grupe.

Slika 5: Primjer hijerarhijske struktura kategorija korisnika



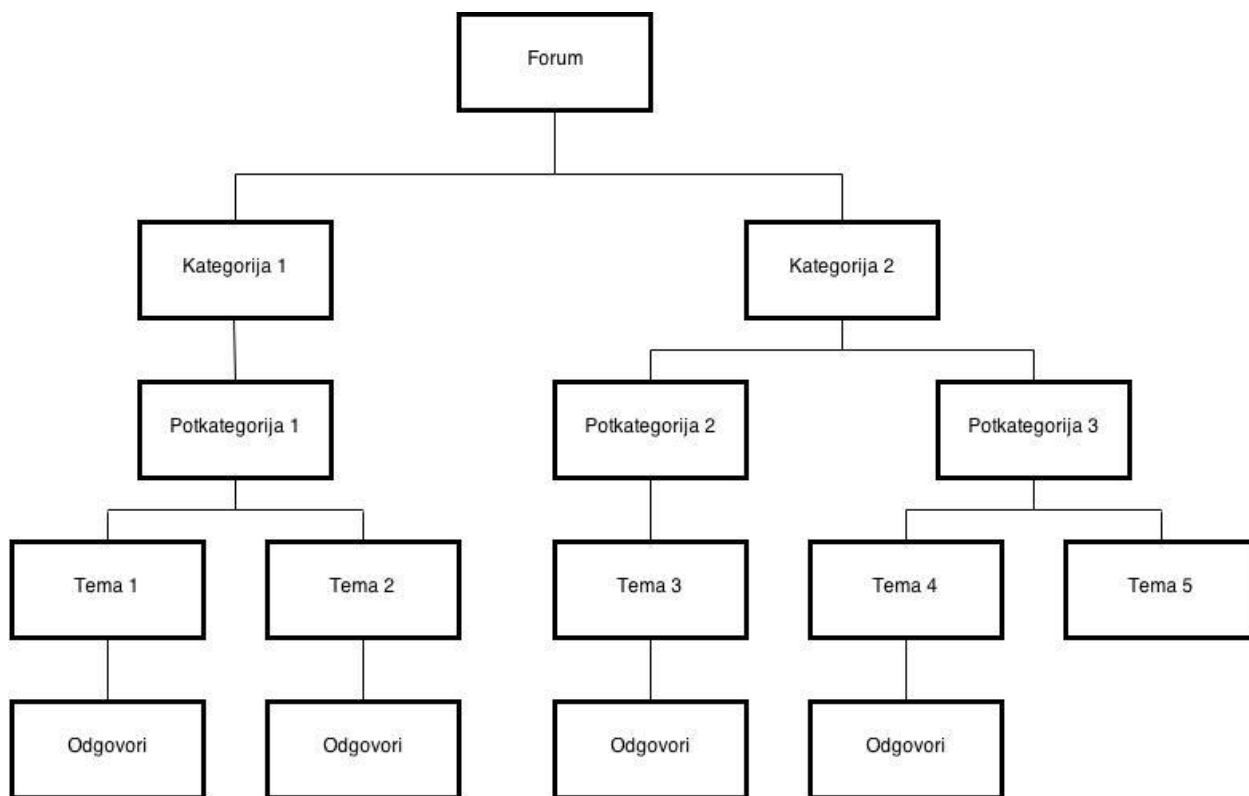
Izvor: Autor

## 6.4. Kategorije tema

Diskusije unutar foruma vode se u obliku tema i odgovora na te teme. Svaka se tema nalazi u svojoj potkategoriji. Potkategorije mogu imati beskrajan broj tema te se nalaze unutar pojedine kategorije. Svaka kategorija obično sadrži mali broj potkategorija kako bi diskusije bile organiziranije i preglednije te je zbog toga i sam broj kategorija malen.



Slika 6: Primjer hijerarhijske strukture kategorija tema



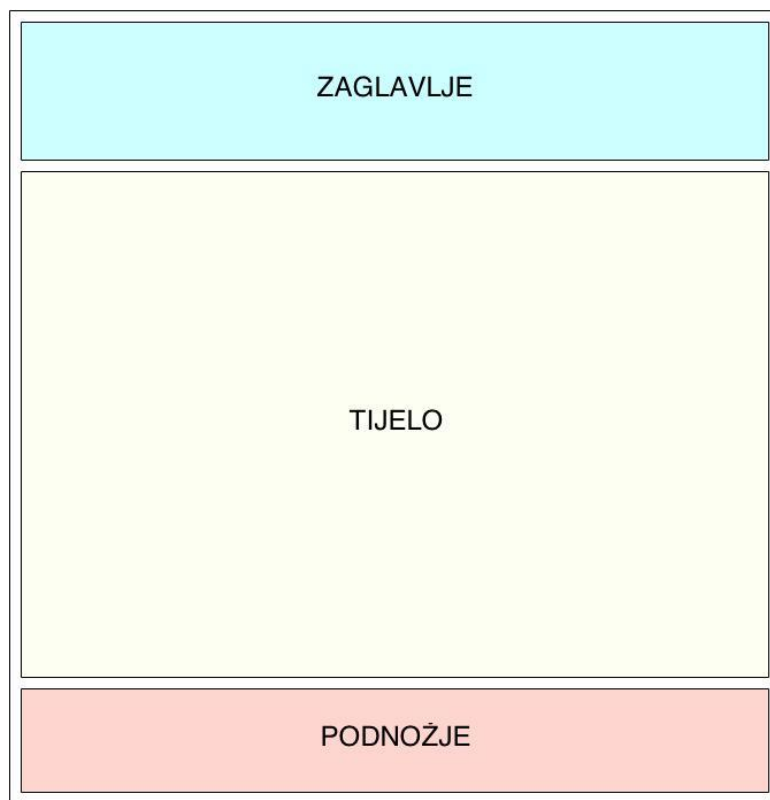
Izvor: Autor

## 6.5. Dizajn stranica

Prilikom izrade web stranica, uobičajeno je da stranice budu podijeljene u više glavnih dijelova. Uobičajeni dijelovi stranice su „header“, „body“, „footer“ te lijevi ili desni „sidebar“. Header uvijek sadrži HTML !DOCTYPE deklaraciju kako bi web preglednik znao koja se verzija HTML-a koristi, početak `<html>` oznake, te `<head>` oznaku unutar koje se stranica povezuje sa potrebnim CSS, JavaScript, jQuery datotekama, zatim meta oznake i naslov stranice. JavaScript i jQuery skripte uobičajeno se također umeću unutar `<head>` oznake, ali se mogu i umetnuti unutar `<body>` oznake. Mnogo programera preporuča implementiranje skripti na samom kraju dokumenta, prije nego li se zatvori `<body>` oznaka, obično u footer dijelu u kojem se zatvaraju `<html>` i `<body>` oznake te omotači.

Kako bi prikaz web sjedišta bio jednoličan na svim stranicama našeg foruma, stranice web sjedišta sustava Forum podijeljene su na zaglavlje, tijelo i podnožje.

Slika 7: Dizajn stranica



Izvor: Autor

Zaglavlje je spremljeno u datoteku naziva `header.php` koji se sastoji od sljedećeg koda:

Isječak 1: `header.php`

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="nl" lang="nl">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <meta name="description" content="A short description." />
  <meta name="keywords" content="put, keywords, here" />
```

```

<title>PHP-MySQL forum</title>
<link rel="stylesheet" href="style.css" type="text/css">
<script type="text/javascript"
src="//ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
<script src="ckeditor/ckeditor.js"></script>
<script src="skripte.js"></script>

</head>
<?php
include 'session.php';
echo '<body>
    <div id="wrapper">
    <div class="alert"></div>
    <div class="menu">
        <a class="logo" href="index.php">FORUM</a>
        <a class="item" href="index.php">HOME</a>';
        if($_SESSION['user_group'] == 1 || ($_SESSION['user_group'] == 2){
            echo '<a class="item" href="acp.php">Control Panel</a>';
        }
    echo '<div class="userbar">';
        if($_SESSION['signed_in'])
        {
echo '<div class="user_name">'. htmlentities($_SESSION['user_name'])
.'</div>
<div class="tip">';

if(!$_SESSION['user_pic']){
    echo '</img>';
        }
else{
    echo '<div class="userbar_image" style="background-
image:url('.$_SESSION['user_pic'].')"></div>';
        }
echo '<div class="tooltip">
    <div class="tooltip_signout"><a id="sign_out" href="signout.php">Sign
out</a></div>

```

```

<div class="my_profile"><a href
="user.php?id='.$_SESSION['user_id'].'">Profile</a></div>

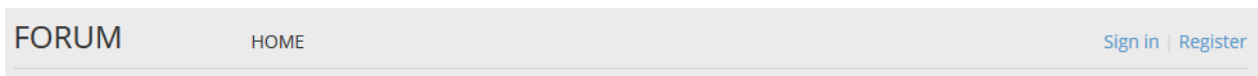
</div>
</div>';
}
else
{
echo '<a href="signin.php">Sign in</a> | <a href="signup.php">Register</a>';
}
?>

    </div>
</div>

<div id="content">
<div id="pad">

```

Slika 8: Header stranice



Izvor: Autor

Osim uobičajenog sadržaja zaglavlja, `header.php` datoteka sustava Forum sadrži i logotip foruma te dinamični sadržaj koji ovisi o tome da li je korisnik prijavljen ili nije te kojoj grupi pripada. Također sadrži i početnu `<body>` oznaku nakon koje slijede ostale oznake koje omotavaju sadržaj foruma.

Isječak 2: `include session.php`

```
include 'session.php';
```

Ovaj dio koda koji se nalazi unutar `header.php` datoteke, poziva sadržaj `session.php` datoteke. Implementiran se kod ponaša kao da je dio same `header.php` datoteke, a prednost

ovakvog načina integriranja koda je što se na jednak način sadržaj datoteke može implementirati unutar drugih datoteka bez potrebe stalnog ponavljanja više redova koda.

`session.php` datoteka sadrži sljedeći kod:

Isječak 3: `session.php`

```
<?php
session_start();
if (!isset($_SESSION['signed_in']) || !isset($_SESSION['user_name']))
{
    $_SESSION['signed_in'] = "";
    $_SESSION['user_name'] = "";
}
if (!isset($_SESSION['user_group']))
{
    $_SESSION['user_group'] = "3";
    $usergroup = $_SESSION['user_group'];
}
else{
    $usergroup = $_SESSION['user_group'];
}??>
```

Kako bi korisnik foruma nakon prijave u sustav mogao ostati prijavljen te koristiti neke od funkcija foruma, neki od korisničkih podataka spremaju se u session varijable. Te varijable imaju rok trajanja te se obično brišu nakon što korisnik napusti web stranicu ili obriše privremene „kolačiće“. Informacije unutar tih varijabli mogu se iskoristiti na drugim stranicama sustava Forum.

Baš kao što se i `session.php` sadržaj implementira unutar `header.php` datoteke, `header.php` datoteka biti će na isti način implementirana unutar drugih stranica web sjedišta sustava Forum, koji se smatraju tijelom foruma.

Na isti način, na samom kraju svake stranice, nalaziti će se i `footer.php` datoteka koja sadrži programski kod podnožja:

Isječak 4: `footer.php`

```
</div><!-- pad -->
</div><!-- content -->
</div><!-- wrapper -->
</body>
</html>
```

Time se zatvaraju omotači sadržaja, te `<body>` i `<html>` oznake. Ostale će stranice sadržavati kod sličan slijedećem primjeru:

Isječak 5: primjer tijela stranice

```
<?php
    <!-- početak header-a -->
        include 'header.php';
    <!-- kraj header-a -->

    <!-- početak body-a -->
        PHP i HTML kod te sadržaj foruma
    <!-- kraj body-a -->

    <!-- početak footer-a -->
        include 'footer.php';
    <!-- kraj footer-a -->
?>
```

## 6.6. Dinamički dio sustava

Cijeli sustav Forum izgrađen je s ciljem da pruži dinamično i interaktivno iskustvo korisniku. Za svaku akciju koju korisnik izvrši, on biva obavješten o rezultatu te akcije gotovo trenutačno, bez osvježivanja stranice, zahvaljujući jQuery-u i Ajaxu. Način na koji je izvedena dinamičnost i interaktivnost sustava Forum biti će prikazana kroz postupke izrade pojedinih

dijelova web aplikacije i to kroz izradu slijedećih funkcija sustava Forum: prikaz početne stranice, registracija i prijava korisnika, uređenje korisničkog profila, kreiranje kategorija i potkategorija te prikaz istih, kreiranje tema i odgovora te prikaz i ažuriranje istih, moderiranje foruma i administracija sustava.

### 6.6.1. Spajanje na bazu podataka pomoću PHP-a

Kako bi sustav Forum imao mogućnost upisivanja i dohvaćanja podataka iz baze podataka, potrebno je uspostaviti konekciju sa bazom. Podaci za uspostavu konekcije dobavljaju se prilikom izrade baze podataka

Najprije je kreirana nova PHP datoteka i imenovana je `connect.php`. Unutar datoteke upisan je slijedeći kod:

Isječak 6: `connect.php`

```
<?php
//connect.php
$server = 'localhost';
$username = 'deni';
$password = '1234';
$database = 'forum';

$conn = mysqli_connect($server, $username, $password, $database) or
die('There was an error');

if (mysqli_connect_errno())
{
    trigger_error('Database connection failed: ' . mysqli_connect_error(),
E_USER_ERROR);
}
?>
```

Informacije potrebne za spajanje na bazu podataka spremljene su u PHP varijable koje su nazvane `$server`, `$username`, `$password` i `$database`. Te varijable upisuju se kao parametri `mysqli_connect()` funkcije koja služi za uspostavu konekcije sa bazom podataka. Takva se funkcija spremakao vrijednost `$conn` varijable. Ta će se varijabla koristiti na svim stranicama foruma kao parametar ostalih `mysqli` funkcija kako bi upisivanje i dohvaćanje podataka u/na bazu bilo moguće. Ukoliko dođe do greške prilikom spajanja na bazu, ispisuje se prikladna poruka.

## 6.6.2. Registracija korisnika

Ukoliko posjetitelj želi postati član foruma, mora se prvo registrirati. Zbog toga se izgrađuje obrazac putem kojega će to i moći učiniti.

Obrazac se izgrađuje u PHP datoteci s nazivom `signup.php`. Na samom početku `signup.php` dokumenta poziva se `header.php` datoteka, a na kraju `footer.php` datoteka. Između njih nalazi se HTML kod za prikaz obrasca.

Isječak 7: `signup.php`

```
<?php
//signup.php
include 'header.php';
?>
<div class="sign_in_body">
  <div class="sign_in_container">
    <div class="sign_in_title">Register</div>
    <form method="post" class="sign_up_form" action="">
      <input id="username" placeholder="username" type="text"
        name="user_name" value="" required maxlength="14" pattern="[A-Za-z-
0-9]+">
      <span id="name-format" class="help">
        Alphanumeric
      </span>
      <input id="password" placeholder="password" type="password"
        name="user_pass" required
      <input id="password_repeat" placeholder="repeat password"
```



```

        type="password" name="user_pass_check" required>
        <span id="pass_match" class="pass_match">
            Incorrect
        </span>
        <input id="email" placeholder="email" type="email"
        name="user_email" required>
        <input class="submit" type="submit" value="REGISTER" />
    </form>
</div>
</div>

<?php
include 'footer.php';
?>

```

Prije nego što korisnik može poslati ispunjen obrazac, mora biti obavješten da nema grešaka. Zbog toga je unutar `<input>` oznaka unešen HTML5 `required` atribut kojim se upozorava korisnika da polja ne mogu ostati prazna, korištenjem HTML5 `pattern` atributa određuje se koje znakove korisnik može upotrijebiti u korisničkom imenu, te `maxlength` atributom određuje se maksimalna dužina korisničkog imena. Validacija emaila vrši se jednostavnim dodavanjem `type="email"` atributa polju za unos emaila.

Od korisnika koji ispunjava obrazac, traži se da upiše lozinku dva puta te se provjerava da li su unesene lozinke identične. Takva provjera nije moguća samo sa HTML-om, potreban je i jQuery. Validaciju lozinke postiže se pomoću jQuery-a.

U HTML kodu prvom polju za unos lozinke dodijeljen je identifikator `password`, a drugom `password_repeat`. Kada god korisnik upisuje podatke u `password_repeat` polje, te stisne i pusti tipku na tipkovnici, jQuery uspoređuje unesenu vrijednost s vrijednošću koju pronalazi u `password` polju. Ukoliko su vrijednosti različite, onemogućuje tipku za slanje obrasca te prikazuje element sa porukom „Incorrect“ koji je sakriven pomoću CSS-a.

#### Isječak 8: jQuery password validacija

```
$(document).ready(function() {
    $("#password_repeat").keyup(function() {
        if($(this).val() != $("#password").val()) {
            $('.pass_match').show();
            $('input[type="submit"]').attr('disabled', 'disabled');
        }
        else {
            $('.pass_match').hide();
            $('input[type="submit"]').removeAttr('disabled');
        }
    });
});
```

Osim validacije obrasca korištenjem informacije dostupne u samom obrascu, potrebna je i validacija pomoću podataka iz baze podataka. Kako ne mogu postajati dva računa s istim imenom i emailom, ako korisnik odabere već postojeće korisničke podatke, potrebno ga je obavijestiti. To se postiže korištenjem jQuery i Ajax.

Nakon pritiska na gumb za slanje obrasca, pokreće se jQuery kod koji sprječava HTML zadanu akciju gumba. Slijedi slanje asinkronog Ajax zahtjeva koji POST metodom šalje odabrane podatke `register.php` datoteci. Podaci koji se šalju su vrijednosti unesene unutar polja obrasca naziva `username`, `password` i `email`.

#### Prvi dio jQuery koda:

#### Isječak 9: jQuery 1. dio slanje obrasca registracije

```
$(document).on('submit', '.sign_up_form', function(e) {
    e.preventDefault();
    $.ajax({
        type: "POST",
        url: "register.php",
```

```

    data:
    'username='+$('#username').val()+'&password='+$('#password').val()+'&email='
    +$('#email').val(),
    success: function(html){
    ...

```

Unutar `register.php` datoteke, u varijable `$username`, `$email` i `$password` spremaju se podaci koji su poslani POST metodom. Kako bi se izbjegla SQL injekcija koristi se `mysqli_real_escape_string()` funkcija te `sha1()` funkcija kako bi se raspršila lozinku prije nego što se upiše u bazu podataka. Upit u bazu podataka, kojim se dohvaća `user_name` i `user_email` iz `users` tablice gdje su `user_name` ili `user_password` jednaki podacima iz POST metode, sprema se u `$sql` varijablu. Zatim se u `$result` varijablu sprema rezultat upita koji se pokreće `mysqli_query()` funkcijom.

Da bi jQuery skripta mogla prikazati korisniku povratnu informaciju o tome da li su korisničko ime i email slobodni ili zauzeti, potrebno je pomoću `if...else` izjave doći do željenih podataka koje je potrebno vratiti jQuery skripti.

Programski kod `register.php` datoteke:

Isječak 10: `register.php`

```

<?php
include 'connect.php';

$username = mysqli_real_escape_string($conn, $_POST['username']);
$email = mysqli_real_escape_string($conn, $_POST['email']);
$password = sha1($_POST['password']);

$sql = "SELECT user_name, user_email
        FROM users
        WHERE user_name = '$username' OR user_email = '$email'";

```

```

$result = mysqli_query($conn, $sql);

if(!$result)
{
    echo 'Error';
}
else if(mysqli_num_rows($result) == 0){
    mysqli_query($conn, "INSERT INTO users(user_name, user_pass, user_email
, user_date, user_group)
                                VALUES('$username', '$password', '$email',
NOW(), 4)") or die(mysqli_error($conn));
    if(mysqli_errno($conn)){
        echo 'Error';
    }
    else{
        echo 'Success';
    }
}
else{
    while($row = mysqli_fetch_assoc($result)){
        if($row['user_name'] == $username){
            echo 'User exists';
        }
        else if($row['user_email'] == $email){
            echo 'Email exists';
        }
    }
}
?>

```

Ovisno o tome da li su uvjeti `if..else` izjave istiniti ili lažni, `$register` datoteka ispisuje određenu poruku koju jQuery skripta očita te s obzirom na poruku obavještava korisnika. Ukoliko je došlo do greške te je `$result` varijabla prazna, jQuery obavještava korisnika da je došlo do greške. Ukoliko upit uspješno dohvati podatke iz baze podataka, ali ne pronade zatraženo ime i email, pokreće se novi upit koji upisuje podatke iz obrasca, jQuery obavještava korisnika o

uspješno obavljenom upisu podataka ili eventualnoj greški prilikom upisa. Ako su ime ili email pronađeni u bazi podataka, korisnik biva obavješten te mora izabrati nove jedinstvene podatke.

Drugi dio jQuery koda koji obavještava korisnika:

Isječak 11: jQuery 2. dio obavijest registracije

```
...
if(jQuery.trim(html) == 'Success'){
    $('.alert').text('Successfully registered,
redirecting...').fadeIn().delay(3000).fadeOut(function(){
        window.location.replace('signin.php');
    });
}
else if(jQuery.trim(html) == 'User exists'){
    $('.incorrect').text('That name is taken.').show();
    $('#username').css('border-color', 'red');
}
else if(jQuery.trim(html) == 'Email exists'){
    $('.incorrect').text('An account already exists with that
email.').show();
    $('#email').css('border-color', 'red')
}
else{
    $('.incorrect').text('There was a problem, try again.').show();
} } }));});
```

Slika 9: Obrazac za registraciju

Register

Deni

alphanumeric

••••

•••••

incorrect

korisnik@email.com

That name is taken.

REGISTER

Izvor: Autor

### 6.6.3. Prijava korisnika

Nakon što se korisnik registrirao na forum, mora imati i mogućnost prijave sa svojim novim korisničkim računom. Zbog toga je potrebno izgraditi obrazac pomoću kojega to može i učiniti. Ovaj će obrazac također sadržavati validaciju pomoću jQuery-a i Ajax-a.

Datoteku koja sadrži obrazac za prijavu nazvana je `signin.php`. Unutar datoteke provjerava se da li je korisnik već prijavljen u sustav. Ukoliko je, nudimo mu se opcija da se odjavi iz sustava, ako nije, prikazuje mu se obrazac za prijavu.

## Isječak 12: signin.php

```
<?php
//signin.php

include 'connect.php';
include 'header.php';

if(isset($_SESSION['signed_in']) && $_SESSION['signed_in'] == true)
{
    echo 'You are already signed in, you can <a href="signout.php">sign
out</a> if you want.';
}
else
{
    echo '<div class="sign_in_body">
    <div class="sign_in_container">
        <div class="sign_in_title">Sign in</div>
        <form class="signin_form" method="post">
            <input id="username" placeholder="username" type="text"
name="user_name" required>
            <input id="password" placeholder="password" type="password"
name="user_pass" required>
            <div class="incorrect"></div>
            <input id="sign_in" class="submit" type="submit" value="SIGN IN" />
        </form>
    </div>
    </div>';
}
include 'footer.php';?>
```

jQuery kod za slanje i validaciju obrasca za prijavu korisnika, vrlo je sličan kodu za registraciju. Za razliku od koda za registraciju, ovaj kod poziva kod `login.php` datoteke.

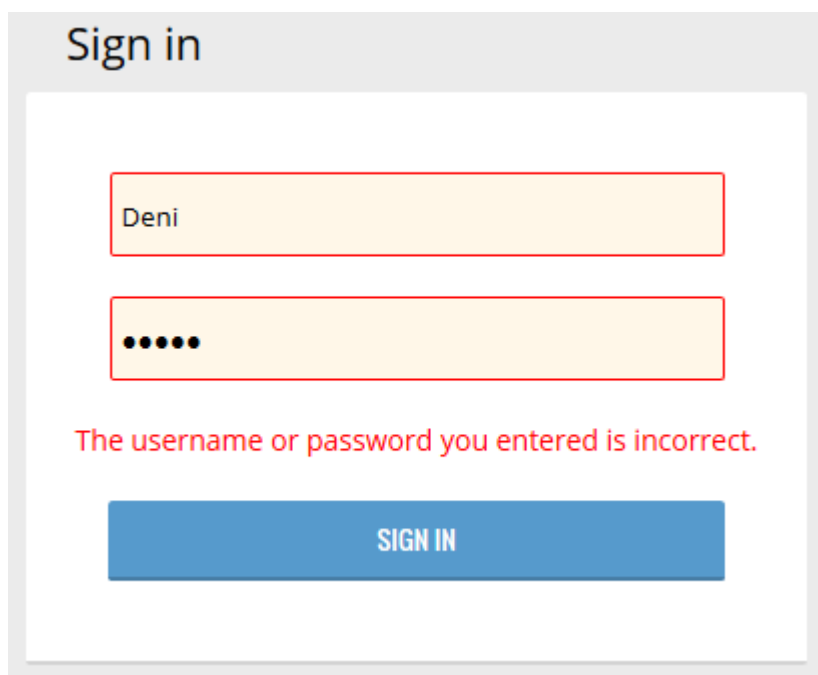
### Isječak 13: jQuery slanje i validacija signin obrasca

```
$(document).on('submit', '.signin_form', function(e){
    e.preventDefault();
    $.ajax({
        type: "POST",
        url: "login.php",
        data:
'username='+$('#username').val()+'&password='+$('#password').val(),
        success: function(html){
            if(jQuery.trim(html) == 'Success'){
                history.go(-1);
            }
            else if(jQuery.trim(html) == 'Fail'){
                $('#incorrect').text('The email or password you entered is
incorrect.').show();
                $('#username, #password').css('border-color', 'red')
            }
            else{
                $('#incorrect').text('There was a problem, try again.').show();
            }
        }
    });
});
```

Nakon što se podaci iz obrasca za prijavu pošalju `login.php` datoteci, ti se podaci spremaju u PHP varijable `$username` i `$password`. Varijable se koriste u upitu u bazu podataka kojim se provjerava da li se kombinacija podataka nalazi u `username` tablici. Ukoliko su oba ili samo jedan podatak netočni, korisniku se prikazuje poruka da su podaci netočni. Ako su u podaci točni, informacije o korisniku spremaju se u `SESSION` varijable koje omogućuju korisniku da ostane prijavljen u sustav dok koristi stranice foruma. Korisnika se vraća na stranicu s koje je pristupio obrascu za prijavu.



Slika 10: Obrazac za prijavu

A sign-in form titled "Sign in" in a grey header. It contains two yellow input fields with red borders. The first field contains the text "Deni". The second field contains five black dots, representing a password. Below the fields is a red error message: "The username or password you entered is incorrect." At the bottom is a blue button with the text "SIGN IN" in white capital letters.

Izvor: Autor

#### 6.6.4. Profil

Nakon što se korisnik prijavio u sustav, u header dijelu stranice pojavljuje se njegova privremena profilna slika. Ukoliko korisnik klikne na tu sliku, pojavljuje se izbornik sa „Profile“ i „Sign out“ opcijama. Klik na „Profile“ opciju vodi korisnika na stranicu sa njegovim profilom, točnije `user.php` datoteku.

Unutar datoteke koristi se korisnikov identifikacijski broj dobavljen iz URL-a kako bi se ispisali podaci o korisniku. Ispisuje se profilna slika, korisničko ime, naziv i naslov grupe, broj tema i odgovora te datum kada je korisnik kreirao račun. Također se i ispisuju obrasci za izmjenu emaila, profilne slike te potpisa.

Isječak 14: `user.php`

```
$sql = "SELECT * FROM users LEFT JOIN groups ON users.user_group =  
groups.group_id WHERE user_id = '$id'";  
$result = mysqli_query($conn, $sql);
```

```

$count_topic = "SELECT topic_id FROM topics WHERE topic_by = '$id'";
$ctr = mysqli_query($conn, $count_topic);
$topic_row_cnt = mysqli_num_rows($ctr);
$count_post = "SELECT post_id FROM posts WHERE post_by = '$id'";
$cpr = mysqli_query($conn, $count_post);
$post_row_cnt = mysqli_num_rows($cpr);
while($row = mysqli_fetch_assoc($result)){
echo '<div class="breadcrumbs"><a href="index.php">Home</a> &#8594; ';
echo '<a href="user.php">User</a> &#8594; ';
echo ' '.$row['user_name'].'';
echo '</div>';
echo '<div class="user_body">';
    echo '<div class="user_container">';
        echo '<div class="profile_info">';
            echo '<div class="user_avatar">';
                if($row['user_pic']){
                    echo '<div class="user_image" style="background-
image:url('.$row['user_pic'].')"></div>';
                }
                else{
                    echo '';
                }

            echo '</div>';
            echo '<ul>';
                echo '<li id="profile_name">'.$row['user_name'].'</li>';
                if($usergroup == 1){
                    echo '<form class="group_change_form"
action="user_commit.php?id='.$id.'">';
                        echo '<select name="group_id">';
                            while($grow = mysqli_fetch_assoc($gresult)){
                                if($grow['user_id'] == $id){
                                    $selected = 'selected="selected"';
                                }
                                else{
                                    $selected = '';
                                }

```

```

        echo '<option value="'. $grow['group_id']. '"
        '$selected.'>'.' $grow['group_name']. '</option>';

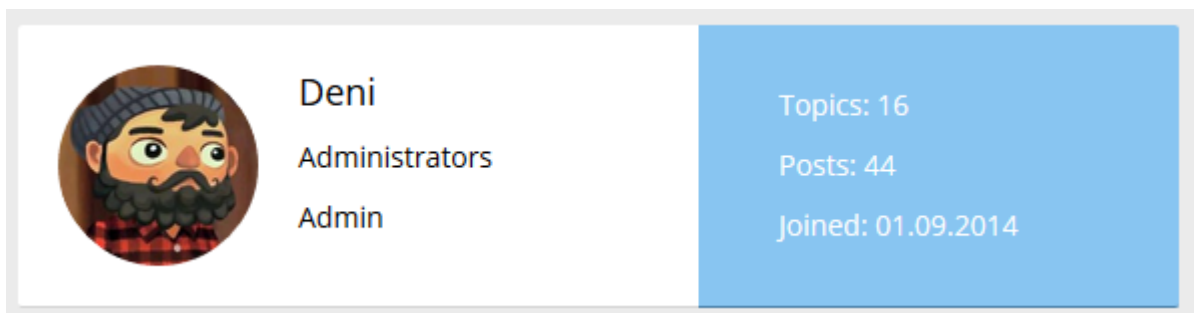
    }
    echo '</select>';
    echo '<input type="submit" value="Save"/>';
    echo '</form>';
}

else{
    echo '<li id="profile_group">'.' $row['group_name']. '</li>';
}

    echo '<li
id="profile_title">'.' $row['group_user_title']. '</li>';
    echo '</ul>';
    echo '</div>';
    echo '<div class="post_info">';
    echo '<ul>';
        echo '<li>Topics: ' . $topic_row_cnt. '</li>';
        echo '<li>Posts: ' . $post_row_cnt. '</li>';
        echo '<li>Joined:
'. date('d.m.Y', strtotime($row['user_date'])) . '</li>';
    echo '</ul>';
    echo '</div>';

```

Slika 11: Korisnički profil



Izvor: Autor

#### 6.6.4.1. Izmjena e-pošte i potpisa

Na korisničkom se profilu nalazi obrazac za promjenu e-pošte. Unutar tekstualnog polja već je upisana postojeća e-pošta korisnika koja se može izmijeniti. Pritiskom na tipku „Update“, jQuery-em šalje se uneseni email i identifikacijski broj korisnika `user_commit.php` datoteci unutar koje se jednostavnim upitom izmjenjuje korisnikov email u bazi podataka. Nakon uspješnog ažuriranja obavještava se korisnika. Izmjena potpisa vrši se na isti način kao i izmjena emaila.

Isječak 15: jQuery Ajax izmjena emaila

```
$(document).on('submit', '.email_form', function(e){
    e.preventDefault();
    $.ajax({
        type      : "POST",
        cache     : false,
        url       : $(this).attr('action'),
        data      : $(this).serialize(),
        success: function(){
            $('#.alert').text('User email
updated').fadeIn().delay(3000).fadeOut();
        }
    });
});
```

#### 6.6.4.2. Izmjena profilne slike

Kako bi korisnik izmijenio profilnu sliku, potreban je obrazac sa atributom `enctype="multipart/form-data"` te posebnim input poljem tipa atributa `type="file"` koje prikazuje gumb „Browse“. Klikom na gumb korisniku se omogućuje odabir lokalne slike. Nakon što je slika odabrana, korisnik mora pritisnuti gumb naziva „Upload“ koja šalje ispunjene podatke te identifikacijski broj korisnika datoteci `upload.php`.

Isječak 16: obrazac za upload profilne slike

```
echo '<form class="upload_form" enctype="multipart/form-data" method="post"
action="upload.php?id='.$_GET['id'].'">
    <div class="img_up_row">
        <label for="fileToUpload">Upload a profile picture (max 2MB, .jpg
        .jpeg .gif .png only)</label><br />
        <input type="file" name="fileToUpload" id="fileToUpload"/>
    </div>
    <div class="img_up_row">
        <input class="submit_avatar" type="submit" value="UPLOAD" />
    </div>
</form>';
```

Kako korisnik ne bi mogao na server staviti nešto što nije slika, unutar `upload.php` datoteke limitira se učitavanje na samo datoteke ekstenzija `.jpg`, `.jpeg`, `.gif` i `.png`. Također se definira maksimalna veličina slike na 2 megabajta. Pomoću `ImageManipulator` klase spremljene u `ImageManipulator.php` datoteci, odabranoj slici korisnika mijenja se veličina te se obrezuje. Trenutačno vrijeme se dodjeljuje nazivu slike kako ne bi bilo konflikta između datoteka sa istim nazivom.

Isječak 17: `upload.php`

```
$id = mysqli_real_escape_string($conn, $_GET['id']);
$valid_extensions = array('.jpg', '.jpeg', '.gif', '.png');
$file_extension = strrchr($_FILES['fileToUpload']['name'], ".");
$maxsize = 2097152;

if (in_array($file_extension, $valid_extensions)) {
    if(($_FILES['fileToUpload']['size'] >= $maxsize) ||
    ($_FILES['fileToUpload']['size'] == 0)) {
        echo 'File too large. File must be less than 2 megabytes.';
    }
    else {
        $newNamePrefix = time() . '_';
        $manipulator = new
        ImageManipulator($_FILES['fileToUpload']['tmp_name']);
```

```

        $newImage = $manipulator->resample(150, 150, true);
        $manipulator->save('uploads/' . $newNamePrefix .
$_FILES['fileToUpload']['name']);
        mysqli_query($conn, "UPDATE users SET user_pic = 'uploads/'.
$newNamePrefix . $_FILES['fileToUpload']['name'].'" WHERE user_id = $id");
        $_SESSION['user_pic'] = 'uploads/' . $newNamePrefix .
$_FILES['fileToUpload']['name'];
        header("Location: user.php?id=$id");

    }

}

else {
    echo 'You must upload an image...';
}

```

### 6.6.5. Unos kategorija

Svaki forum mora imati kategorije koje korisnicima daju do znanja kakav sadržaj će moći pronaći unutar tog foruma. Svaka kategorija obično ima niz potkategorija koje dalje filtriraju taj sadržaj. Primjerice, unutar kategorije naziva „Film“ mogu se nalaziti potkategorije naziva „Akcijski“, „Komedija“, „Horor“ i slično.

Pregled kategorija foruma predstavlja se na početnoj stranici foruma tako da budu odmah dostupne korisniku, ali prije toga potrebno je izraditi obrazac koji dozvoljava administratoru foruma da kreira nove kategorije.

Kod se upisuje unutar `create_cat.php` datoteke. `If...else` izjavom provjerava se da li je korisnik pripadnik administratorske grupe. Ukoliko nije, korisnik je obavješten da nema pristup te se obrazac ne prikazuje.

Ukoliko korisnik ima pristup, pomoću još jedne `if...else` izjave utvrđuje se da li stranica sadrži POST metodu. Ukoliko ne sadrži, prikazuje se obrazac, u suprotnom se izvršava upit u bazu podataka.

Razlog `if...else` izjave je što se u ovom slučaju ne koristi jQuery ili Ajax za validaciju, te nakon što korisnik pošalje ispunjen obrazac, kojemu `action` atribut nema vrijednosti, POST metodom se ti podaci šalju nazad na istu stranicu s obrascem koja sadrži i upit u bazu.

Upit u bazu je jednostavan. Podaci iz obrasca, koji sadrže naziv kategorije i kratak opis, upisuju se u tablicu `categories` u bazi podataka. Pomoću `mysqli_insert_id()` funkcije dobavlja se identifikator novokreirane kategorije. Također se šalje upit koji dobavlja podatke o grupama ukoliko postoje. While petljom te `mysqli_fetch_assoc()` funkcijom, spremaju se podaci o svakoj dobavljenoj grupi u PHP varijable te se ti podaci, zajedno sa identifikatorom kategorije, spremaju u tablicu `cat_perms` koja sadrži podatke o odobrenjima svake grupe za svaki pojedini forum prema postavkama administratora. Ukoliko dođe do bilo kakve greške, prikazuje se upozorenje.

Programski kod unutar `create_cat.php` datoteke:

Isječak 18: `create_cat.php`

```
<?php
//create_cat.php
include 'connect.php';
include 'header.php';
if($_SESSION['signed_in'] == false | $_SESSION['user_group'] != 1 )
{
echo 'Sorry, you do not have sufficient rights to access this page.';
}
else
{
    if($_SERVER['REQUEST_METHOD'] != 'POST')
    {
        echo '<div class="sign_in_body">
        <div class="sign_in_container">
        <div class="sign_in_title">Create a category</div>
        <form method="post" action="">
        <input class="cat_name" placeholder="name" type="text" name="cat_name" />
        <textarea class="cat_description" placeholder="description"
        name="cat_description"></textarea>
        <input class="submit" type="submit" value="CREATE" />
```

```

        </form></div></div>';
    }
    else
    {
        $sql = "INSERT INTO categories(cat_name, cat_description)
        VALUES('" . mysqli_real_escape_string($conn, $_POST['cat_name']) . "',
        '" . mysqli_real_escape_string($conn, $_POST['cat_description']) .
        "')";
        $result = mysqli_query($conn, $sql);
        $cat_id = mysqli_insert_id($conn);
        $gsql = "SELECT * FROM groups";
        $gresult = mysqli_query($conn, $gsql);
        while($row = mysqli_fetch_assoc($gresult)){
            $gid = $row['group_id'];
            $read_forum = $row['group_read_board'];
            $post_replies = $row['group_post_replies'];
            $post_topics = $row['group_post_topics'];

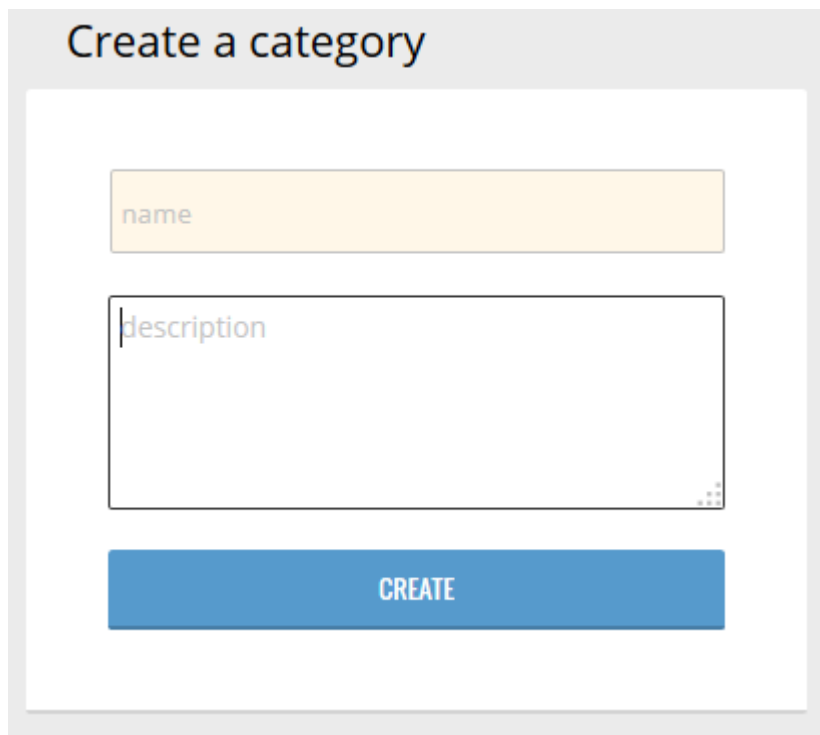
            mysqli_query($conn, "INSERT INTO cat_perms (group_id, cat_id, read_forum,
            post_replies, post_topics) VALUES('$gid', '$cat_id', '$read_forum',
            '$post_replies', '$post_topics')") or die(mysqli_error($conn));
        }

        if(!$result || !$gresult)
        {
            echo 'Error' . mysqli_error($conn);
        }
        else
        {
            header("Location: acp_forums.php");
            die();
        }
    }
}
include 'footer.php';
?>

```



Slika 12: Obrazac za kreiranje kategorije



The image shows a web form titled "Create a category". It contains two input fields: a single-line text box for "name" and a multi-line text area for "description". Below these fields is a prominent blue button with the text "CREATE" in white capital letters. The form is set against a light gray background.

Izvor: Autor

#### 6.6.6. Unos potkategorija

Način kreiranja potkategorija, sličan je načinu kreiranja kategorija. `create_subcat.php` sadrži kod za obrazac u koji se upisuju željeni naziv potkategorije i opis te se sa padajuće liste odabire kategorija kojoj se želi pridružiti potkategorija. Nazivi i identifikatori kategorija za padajuću listu dobavljaju se upitom u bazu podataka. Ukoliko ne postoji niti jedna kategorija, korisnika se obavještava i obrazac se ne prikazuje.

Kao i kod `create_cat.php` koda, provjerava se da li korisnik ima pristup te se provjerava POST metoda prije nego što se prikaže obrazac. Nakon slanja obrasca, odabrani podaci upisuju se u tablicu `subcategories`. Upitima se dobavljaju identifikator potkategorije i podaci o grupama te se upisuju u tablicu `subcat_perms` koja sadrži podatke o odobrenjima svake grupe za svaku pojedinu potkategoriju prema postavkama administratora. Ukoliko dođe do greške, prikazuje se upozorenje.

Isječak 19: create\_subcat.php

```
<?php
//create_subcat.php
include 'connect.php';
include 'header.php';
if($_SESSION['signed_in'] == false | $_SESSION['user_group'] != 1 )
{
    echo 'Sorry, you do not have sufficient rights to access this page.';
}
else{
if($_SERVER['REQUEST_METHOD'] != 'POST'){
    $sql = "SELECT cat_id, cat_name, cat_description
            FROM categories";

    $result = mysqli_query($conn, $sql);

if(!$result){
    echo 'Error while selecting from database. Please try again later.';
}
else{
if(mysqli_num_rows($result) == 0){
    if($_SESSION['user_group'] == 1) {
        echo 'You have not created categories yet.';
    }
    else {
        echo 'Before you can create a subcategory, you must create some
categories.';
    }
}
}

    echo '<div class="sign_in_body"><div class="sign_in_container">
    <div class="sign_in_title">Create a subcategory</div>
    <form method="post" action="">
    <input class="cat_name" placeholder="name" type="text" name="subcat_name"
/>
```

```

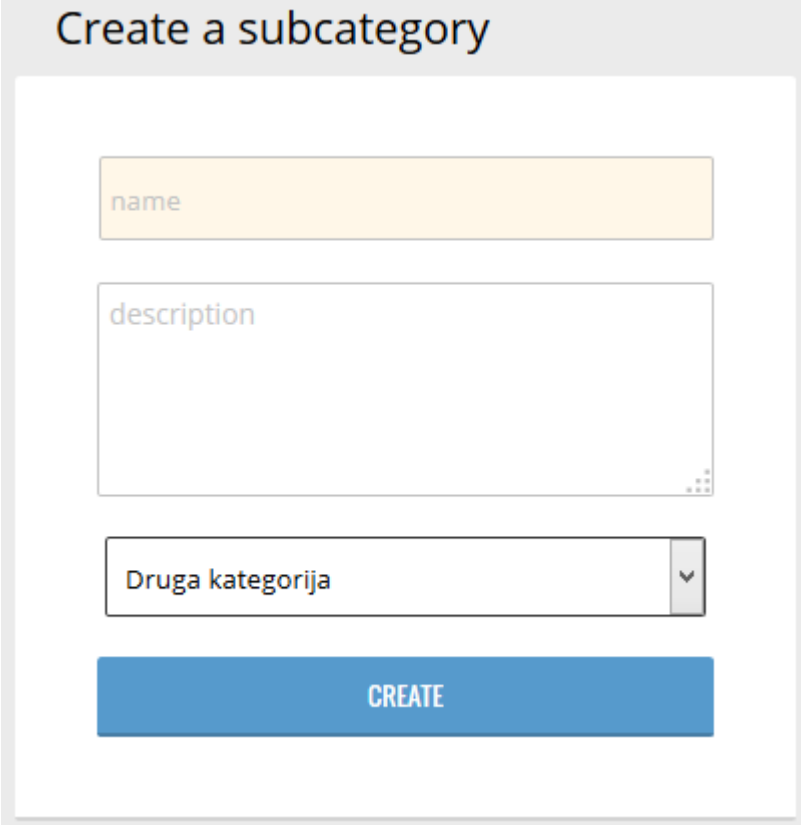
        <textarea class="cat_description" placeholder="description"
name="subcat_description"></textarea>

<select class="cat_name" name="subcat_cat">';
    while($row = mysqli_fetch_assoc($result))
    {
        echo '<option value="' . $row['cat_id'] . '">' . $row['cat_name'] .
'</option>';
    }
    echo '</select><input class="submit" type="submit" value="CREATE"
/></form></div></div>';
}
else{
    $sql = "INSERT INTO subcategories(subcat_name, subcat_description,
subcat_cat)
VALUES ('" . mysqli_real_escape_string($conn, $_POST['subcat_name']) . '",
        '" . mysqli_real_escape_string($conn,
$_POST['subcat_description']) . "',
        '" . mysqli_real_escape_string($conn,
$_POST['subcat_cat']) . "')";
    $result = mysqli_query($conn, $sql);
    $subcat_id = mysqli_insert_id($conn);
    $gsql = "SELECT * FROM groups";
    $gresult = mysqli_query($conn, $gsql);
    while($row = mysqli_fetch_assoc($gresult)){
        $gid = $row['group_id'];
        $read_forum = $row['group_read_board'];
        $post_replies = $row['group_post_replies'];
        $post_topics = $row['group_post_topics'];
        mysqli_query($conn, "INSERT INTO subcat_perms (group_id, subcat_id,
read_forum, post_replies, post_topics) VALUES('$gid', '$subcat_id',
'$read_forum', '$post_replies', '$post_topics')") or
die(mysqli_error($conn));}
    if(!$result || !$gresult){
        echo 'Error' . mysqli_error($conn);
    }
    else{

```

```
header("Location: acp_forums.php");  
die();  
}}}  
include 'footer.php';  
?>
```

Slika 13: Obrazac za kreiranje potkategorije



The image shows a web form titled "Create a subcategory". It contains three input fields: a text field for "name", a text area for "description", and a dropdown menu currently showing "Druga kategorija". A blue "CREATE" button is at the bottom.

Izvor: Autor

### 6.6.7. Početna stranica

Početna stranica je prva stranica koju posjetitelj vidi kada posjeti web sjedište. Osnovni elementi koje početna stranica foruma obično sadrži su: naziv foruma, navigacija, hiperveze za prijavu i registraciju te hijerarhijski popis kategorija i potkategorija.

Kod potreban za prikaz početne stranice upisan je u `index.php` datoteku koja se automatski otvara kada netko posjeti web sjedište foruma.

Na samom početku datoteke deklarirana je funkcija naziva `relative_date`. Ova funkcija uspoređuje datum dobavljen iz baze podataka sa trenutnim datumom, te ovisno o rezultatu, umjesto brojeva u datumu, prikazuje riječ „Today“, „Tomorrow“, „Yesterday“ ili brojeve prikazuje u jednom od standardnih oblika datuma. Djeluje na način da od dobavljenog datuma iz baze, koji je poslan funkciji u UNIX formatu pomoću `strtotime()` PHP funkcije, oduzima trenutni datum UNIX formata te rezultat dijeli sa brojem 86400 koji predstavlja jedan dan. Ovisno o rezultatu `if...else` izjave funkcije, ispisuje se odgovarajuća riječ. Ova se funkcija koristi prilikom prikaza datuma najnovije teme svake potkategorije.

Isječak 20: `relative_date` funkcija

```
function relative_date($time){
    $today = strtotime(date('M j, Y'));
    $reldays = ($time - $today)/86400;
    if ($reldays >= 0 && $reldays < 1) {
        return 'Today, ' . date('h:i A', $time);
    }
    else if ($reldays >= 1 && $reldays < 2) {
        return 'Tomorrow, ' . date('h:i A', $time);
    }
    else if ($reldays >= -1 && $reldays < 0) {
        return 'Yesterday, ' . date('h:i A', $time);
    }
    if (abs($reldays) < 182) {
        return date('d-m-Y, h:i A', $time);
    }
}
```

Slijedi upit u bazu kojom se dobivljaju podaci o kategorijama te se za svaku kategoriju dobivljaju dozvole grupe u kojoj se posjetitelj nalazi. Ukoliko posjetitelj nije prijavljen, smatra se članom grupe „Guest“. Rezultat upita upisuje se u varijablu `$result`.

Isječak 21: upit u bazu za prikaz kategorija i potkategorija na index.php

```
$sql = "SELECT categories.*, cat_perms.*
        FROM categories
        LEFT JOIN cat_perms
        ON categories.cat_id = cat_perms.cat_id
        WHERE cat_perms.group_id = '$usergroup'
        GROUP BY categories.cat_name, categories.cat_description,
categories.cat_id
        ORDER BY cat_order DESC";

$result = mysqli_query($conn, $sql);
```

If...else izjavama, provjerava se rezultat upita. Ukoliko nema rezultata ili je došlo do greške sa upitom, ispisuje se odgovarajuća poruka, u suprotnom se odvija kod potreban za prikaz sadržaja.

Za prikaz tog sadržaja, pomoću while petlje prolazi se kroz rezultat upita. Unutar petlje, provjeravaju se postavljene dozvole za prikaz sadržaja te ako dozvola za prikaz to ne dozvoljava, određeni sadržaj se ne prikazuje. Petlja ispisuje kategorije te za svaku kategoriju šalje upit u bazu kojom dohvaća njene potkategorije.

Isječak 22: while petlja za prikaz kategorija

```
while($row = mysqli_fetch_assoc($result))
{
    if($row['cat_status_hidden'] != 1){
        if($row['read_forum'] == 1 )
        {
            echo '<div class="cat_title">';
            echo '<div class="pad20">';
            echo '<a href="category.php?id=' . $row['cat_id'] . '"> ' .
$row['cat_name'] . '</a>';
            echo '</div>';
            echo '<div class="last_post">';
```

```

echo 'Latest topic';
echo '</div>';
echo '</div>';

$subcatsql = "SELECT subcategories.*, subcat_perms.*
               FROM subcategories
               LEFT JOIN subcat_perms
               ON subcategories.subcat_id = subcat_perms.subcat_id
               WHERE subcat_cat = '" . $row['cat_id'] . "'
               AND subcat_perms.group_id = '$usergroup'
               GROUP BY subcat_name
               ORDER BY subcat_order DESC";

$subcatresult = mysqli_query($conn, $subcatsql);

```

Rezultat se sprema u varijablu `$subcatresult` koja se koristi u drugoj `while` petlji kako bi se ispisale sve potkategorije za pojedinu kategoriju. Nazivi potkategorija su ujedno i hiperveze koje vode na `subcategory.php` datoteku s odgovarajućim identifikatorom potkategorije. Za svaku se potkategoriju šalje upit u bazu kojom se dobavljaju podaci o najnovijoj temi unutar te potkategorije te podaci o autoru teme. Ukoliko nema rezultata ili je došlo do greške, ispisuju se prigodne poruke.

Isječak 23: `while` petlja za prikaz potkategorija

```

while($subcatrow = mysqli_fetch_assoc($subcatresult))
{
    if($row['cat_status_subcat'] != 1){
        if($subcatrow['subcat_status_hidden'] != 1){
            if($subcatrow['read_forum'] == 1){

                echo '<div class="subcat_row">';
                echo '</img>';
                echo '<div class="subcat_data">';
                echo '<ul>';

```

```

        echo '<li><a href="subcategory.php?id=' . $subcatrow['subcat_id'] .
'>' . $subcatrow['subcat_name'] . '</a></li>';
        echo '<li
class="subcat_desc">'. $subcatrow['subcat_description']. '</li>';
        echo '</ul>';
        echo '</div>';
$topicsql = "SELECT topic_id, topic_subject, topic_date, topic_by,
users.user_name, users.user_pic
        FROM topics
        LEFT JOIN users
        ON topics.topic_by = users.user_id
        WHERE topic_subcat = ".$subcatrow['subcat_id'].
        ORDER BY topic_date DESC
        LIMIT 1
        ";

$topicsresult = mysqli_query($conn, $topicsql);

```

Osim hiperveze i naslova teme, prikazuje se korisničko ime i profilna slika korisnika koji je kreirao temu, te datum kreiranja teme koji je pomoću `relative_date` funkcije izmijenjen.

Također se za svaku potkategoriju šalju upiti koji broje koliko pojedina kategorija sadrži tema i odgovora. Ukoliko postoje i nema grešaka, rezultati se upisuju u varijable te se prikazuju.

Isječak 24: upit kojim se izračunava broj tema i odgovora unutar potkategorije

```

$sql_topic_count = "SELECT COUNT(topic_id)
        FROM topics
        WHERE topic_subcat = " . $subcatrow['subcat_id'];

$result_topic_count = mysqli_query($conn, $sql_topic_count);
$row_topic_count = mysqli_fetch_row($result_topic_count);
$total_topics = $row_topic_count[0];
$sql_post_count = "SELECT COUNT(post_id)
        FROM posts

```










```

        LEFT JOIN topics
        ON posts.post_topic = topics.topic_id
WHERE topic_subcat = " . $subcatrow['subcat_id'];
$result_post_count = mysqli_query($conn, $sql_post_count);
$row_post_count = mysqli_fetch_row($result_post_count);
$total_posts = $row_post_count[0];

echo '<div class="subcat_count">';
echo '<li>'.$total_topics.' topics</li>';
echo '<li>'.$total_posts.' posts</li>';
echo '</div>';
echo '</div>';

```

Slika 14: Početna stranica web sjedišta

Film		Latest topic
 <b>Komedija</b> Haha	2 topics 1 posts	 <a href="#">Najbolje komedije</a> Today, 07:31 AM by <a href="#">Deni</a>
 <b>Akcija</b> Eksplodiraj!	1 topics 0 posts	 <a href="#">Rambo 8?</a> Today, 07:37 AM by <a href="#">Korra</a>
Recepti		Latest topic
 <b>Glavna jela</b> Mmmm	1 topics 0 posts	 <a href="#">Boeuf Bourguignon Recipe</a> Today, 07:40 AM by <a href="#">JuliaChild</a>
 <b>Deserti</b> Mljaci	0 topics 0 posts	There are no topics

Izvor: Autor

### 6.6.8. Potkategorije

Potkategorije prikazane na početnoj stranici su ujedno i hiperveze. Te hiperveze vode na `subcategory.php` datoteku te je svakoj hipervezi dodijeljen jedinstveni identifikator potkategorije koju predstavlja. Taj se identifikator šalje GET metodom preko URL-a u obliku indeksa. Identifikator se dodjeljuje hipervezi na način da mu se dodijeli ime te vrijednost npr.

`subcategory.php?id="1"` gdje `id` predstavlja naziv indeksa, a broj `1` predstavlja njegovu vrijednost.

Nakon što korisnik klikne na hipervezu potkategorije, otvara se `subcategory.php` datoteka. Na početku te datoteke, varijabli `$id` dodjeljuje se vrijednost `id` iz GET metode. Potkategorije obično sadrže velik broj tema te je zbog toga teme dijele na „stranice“.

Stranica na kojoj se korisnik nalazi, prepoznaje se preko drugog indeksa naziva `page`. On se šalje URL-om koristeći hiperveze paginacije. U funkciju naziva `pagination`, spremaju se potrebni elementi i PHP kod potreban za prikaz paginacije. Funkcija radi na način da ukoliko postoji `page` indeks u URL-u, sprema se njegova vrijednost u `$page`, `$pageprev` te `$pagenext` varijable te ukoliko ne postoji, dodjeljuje im se vrijednost broj `1`. U varijablu `$ppp` spremljen je željeni broj tema po stranici te upitom u bazu dobavlja se ukupan broj tema u trenutnoj potkategoriji. Ukupan se broj tema dijeli sa željenim brojem tema po stranici te se time dobiva ukupan broj stranica koji se zaokružuje na slijedeći cijeli broj i sprema u varijablu `$total_pages`.

`If...else` izjavama provjerava se vrijednost varijabli koje predstavljaju stranice. Ukoliko postoji samo jedna stranica, paginacija se ne prikazuje. Ako je vrijednost `$pageprev` varijable veća od `1`, njoj se oduzima vrijednost za `1` te se ispisuje kao gumb za pomicanje na prijašnju stranicu. Primjer: Korisnik se nalazi na stranici broj `4`. U `$pageprev` sprema se vrijednost kao broj `4` te mu se oduzima `1`. `$pageprev` sada sadrži vrijednost broj `3` te ispisuje gumb sa hipervezom koji sadrži tu vrijednost kao vrijednost `page` indeksa. Kada korisnik pritisne gumb, odlazi na stranicu broj `3` te se cijeli proces ponavlja. Na isti se način izvodi proces za gumb koji vodi na slijedeću stranicu, samo što se dodaje broj `1` varijabli `$pagenext` umjesto da se oduzima. Osim tih gumbova, prikazuju se i gumbi sa popisom svih stranica. Ako postoji `3` ili manje stranice, sve se stranice ispisuju. Ako postoji više od `5` stranica, prikazuju se prva i zadnja stranica te samo nekoliko stranica prije i poslije trenutne stranice.

Isječak 25: pagination funkcija

```
function pagination() {  
    global $id;
```

```

global $conn;
if (isset($_GET["page"])){
    $page = $_GET["page"];
    $pageprev = $_GET["page"];
    $pagenext = $_GET["page"];
}else{
    $page = 1;
    $pageprev = 1;
    $pagenext = 1;
}
$ppp = 10;
$sql_count = "SELECT COUNT(topic_id)
              FROM topics
              WHERE topic_subcat = " . $id;
$result_count = mysqli_query($conn, $sql_count);
$row_count = mysqli_fetch_row($result_count);
$total = $row_count[0];
$total_pages = ceil($total/$ppp);

echo '<div class="pagination">';
if($total_pages == 1){}
else{
    if($pageprev > 1){
        echo '<a href="subcategory.php?id=' . $id . '&page=' . --$pageprev
        . '">';

        echo '<div class="topic_page_num"><</div>';
        echo '</a>';
    }
    if($total_pages <= 3){
        for($i = 1; $i <= $total_pages; $i++){
            echo '<a href="subcategory.php?id=' . $id . '&page=' . $i . '">';
            echo '<div class="topic_page_num">' . $i . '</div>';
            echo '</a>';
        }
    }
}

```

```

else{
    echo '<a href="subcategory.php?id='. $id .'&page=1">';
    echo '<div class="topic_page_num">1</div>';
    echo '</a>';
    if($page >= 5){
        echo '<div class="topic_page_num_dots">...</div>';
    }
    for($i = $page - 2; $i <= $page + 2; $i++){
        if($i <= 1){
        }
        else if($i == $total_pages){
            break;
        }
        else{
            echo '<a href="subcategory.php?id='. $id .'&page='. $i
.'">';

            echo '<div class="topic_page_num">'. $i . '</div>';
            echo '</a>';

        }
    }
    if($page <= $total_pages - 4){
        echo '<div class="topic_page_num_dots">...</div>';
    }
    echo '<a href="subcategory.php?id='. $id .'&page='. $total_pages
.'">';

    echo '<div class="topic_page_num">'. $total_pages . '</div>';
    echo '</a>';

    }
}

if($pagenext < $total_pages){
    echo '<a href="subcategory.php?id='. $id .'&page='. ++$pagenext
.'">';

    echo '<div class="topic_page_num">></div>';
    echo '</a>';    }
echo '</div>';
}

```

Slika 15: Paginacija



Izvor: Autor

Kako bi se prikazale sve teme odabrane potkategorije, šalje se upit u bazu podataka koji dobavlja podatke o potkategoriji, kategoriji u kojoj se nalazi, te dozvole postavljene na tu potkategoriju i kategoriju. `While` petljom i `If...else` izjavama provjerava se dozvola pregledavanja sadržaja grupe trenutnog korisnika, te se sadržaj potkategorije prikazuje ili ne prikazuje s obzirom na te dozvole. Također se provjerava dozvola za stvaranje novih tema. Ukoliko korisnik nema dozvolu, gumb za novu temu se ne prikazuje. Ispisuje se i takozvana breadcrumbs navigacija koja prikazuje put do trenutne pozicije na web sjedištu.

Isječak 26: breadcrumbs navigacija

```
while($perm_row = mysqli_fetch_assoc($perm_rs)){
    if($perm_row['cat_status_hidden'] != 1){
        if($perm_row['cat_read'] == 1){
            if($perm_row['cat_status_subcat'] != 1){
                if($perm_row['subcat_status_hidden'] != 1){
                    if($perm_row['read_forum'] == 1){

while($row = mysqli_fetch_assoc($rs)){
    echo '<div class="breadcrumbs"><a href="index.php">Home</a> &#8594; ';
        echo '
                                ' <a
href="category.php?id='.$row['subcat_cat'].'">'.$row['cat_name'].'</a>
&#8594; ';
        echo ' '.$row['subcat_name'].'';
    echo '</div>';
}

if($perm_row['cat_topics'] == 1){
    if($perm_row['post_topics'] == 1){
```

```

        echo '<a href="create_topic.php?id='.$id.'"><div id="create_topic"
class="create_topic">START A TOPIC</div></a>';
    }
}

echo''.pagination().'';

```

Dalje se šalje upit u bazu kojom se dobavljaju teme trenutačne kategorije te informacije o korisnicima koji su započeli teme. Ispisuju se pomoću `while` petlje. U varijabli `$start_from` izračunava se broj koji govori upitu od kojeg reda u popisu dobavljenih tema da započne s prikazom s obzirom na trenutačnu stranicu paginacije. Od trenutačne vrijednosti stranice, oduzima se vrijednost za 1 te se dobiveni broj pomnoži sa brojem iz `$ppp` varijable. Primjer: korisnik se nalazi na stranici broj 3 te je vrijednost `$ppp = 10`. Rezultat varijable sa zadanom formulom u ovom slučaju iznosi 20 što znači da se na stranici broj 3 prikazuje 10 tema počevši od teme čiji je redni broj 20 s obzirom na sortiranje tema prema datumu zadnjeg odgovora, datumu početka teme te statusu teme.

Isječak 27: upit za prikaz određen broj tema po stranici

```

$start_from = ($page-1) * $ppp;

$subcatsql = "SELECT topics.*, users.user_name, users.user_pic
FROM topics
LEFT JOIN users
ON topics.topic_by = users.user_id
LEFT JOIN posts
ON topics.topic_id = posts.post_topic
WHERE topic_subcat = '" . $row['subcat_id'] . "'
GROUP BY topics.topic_id
ORDER BY topic_status_pinned DESC, last_post_date DESC
LIMIT ". $start_from . ", ". $ppp . ";
$subcatresult = mysqli_query($conn, $subcatsql);

```

S obzirom na status teme, uz naslov teme ispisuje se i oznaka koja prikazuje taj status.

Isječak 28: programski kod koji prikazuje status teme

```
echo '<div class="topic_row">';
echo '</img>';
echo '<div class="topic_data">';
if($subcatrow['topic_status_locked'] == 1){
echo'<span class="locked">Locked</span>';
}
else{
echo'<span class="unlocked">Locked</span>';
}
if($subcatrow['topic_status_pinned'] == 1){
echo'<span class="pinned">Pinned</span>';
}
else{
echo'<span class="unpinned">Pinned</span>';
}
echo '<a href="topic.php?id=' . $subcatrow['topic_id'] . '">';
$subcatrow['topic_subject'] . '</a>';
echo '</div>';
```

Uz status teme, još se i ispisuje paginacija koja prikazuje broj stranica po temi, upit i kod su slični kao i kod prijašnje paginacije, jedino što se podaci dobivljaju iz tablice o odgovorima te nisu potrebni gumbići za prijašnju ili slijedeću stranicu. Prikazuju se prve tri stranice te zadnja stranica.

Isječak 29: upit za prikaz mini paginacije pokraj naziva teme

```
$posts_per_page = 10;
$sql_count_posts = "SELECT COUNT(post_id)
                    FROM posts
```

```

WHERE post_topic = " . $subcatrow['topic_id'];

$result_count_posts = mysqli_query($conn, $sql_count_posts);
$row_count_posts = mysqli_fetch_row($result_count_posts);
$total_posts = $row_count_posts[0];
$total_post_pages = ceil($total_posts/$posts_per_page);

$topic_timestamp = $subcatrow['topic_date'];
$topic_relative_time = relative_date(strtotime($topic_timestamp));

if($total_post_pages == 1){}
else{
if($total_post_pages <= 3){
for($i = 1; $i <= $total_post_pages; $i++){
echo '<a href="topic.php?id=' . $subcatrow['topic_id'] . '&page=' . $i . '">
<div class="page_num">' . $i . '</div></a>';
}
}
else{
for($i = 1; $i <= 3; $i++){
echo '<a href="topic.php?id=' . $subcatrow['topic_id'] . '&page=' . $i . '">
<div class="page_num">' . $i . '</div></a>';}
echo '<div class="page_num">...</div>';
echo '    <a href="topic.php?id=' . $subcatrow['topic_id'] . '&page=' .
$total_post_pages . '">
<div class="page_num">' . $total_post_pages . '</div></a>';}}

```

Za svaku se temu novim upitom dobavljaju podaci o zadnjem odgovoru unutar te teme. Podaci o zadnjem odgovoru ispisuju se u istom redu kao i naziv teme. Ti podaci sadrže ime autora odgovora, sliku autora, datum odgovora te hipervezu koja vodi na točnu lokaciju odgovora. Datum odgovora se pomoću funkcije `relative_date`, kao i kod `index.php` datoteke, pretvara u relativni datum. Kod svake se teme također prikazuje broj pogleda teme te broj odgovora.

Broj pogleda teme spremljen je u tablici sa podacima o temi. Programski kod potreban za izračun pogleda nalazi se u `topic.php` datoteci. Programski kod radi na slijedeći način: pomoću



GET metode, dobavlja se identifikator teme koju je korisnik posjetio. Unutar `$sample_rate` varijable, spremljena je vrijednost broj 10. Varijabla `$random` sadrži nasumičnu vrijednost između brojeva 9 i 11 koji se generira prilikom pokretanja datoteke. Unutar uvjeta `if` izjave generira se nasumični broj između broja 1 i `$sample_rate` vrijednosti. Ukoliko je taj nasumični broj jednak vrijednosti broj 1, vrijednost iz `$random` varijable upisuje se u bazu podataka za temu u kojoj se korisnik nalazi. Razlog korištenja nasumičnih brojeva je taj kako bi se smanjio broj upisa u bazu sa svakim posjetiteljem teme. Na ovaj način, statistički će svaki deseti posjetitelj teme upisati vrijednost između 9 i 11 kao broj pregleda teme.

Isječak 30: programski kod za generiranja nasumičnog broja pogleda teme

```
$id = mysqli_real_escape_string($conn, $_GET['id']);
$sample_rate = 10;
$random = mt_rand(9,11);

if(mt_rand(1,$sample_rate) == 1) {
    $query = "UPDATE topics
              SET topic_views = topic_views + ".$random."
              WHERE topic_id = '".$id."' ";
    mysqli_query($conn, $query);
}
```

### 6.6.9. Nova tema

Unutar `subcategory.php` datoteke, nalazi se gumb koji kada se pritisne, vodi na `create_topic.php` datoteku, zajedno sa identifikatorom potkategorije na kojoj je pritisnut.

Ova datoteka sadrži obrazac te upite potrebne za kreiranje nove teme. Na početku datoteke, šalje se upit koji dohvaća dozvole kao i kod potkategorija. Ukoliko korisnik nema pristup određenoj kategoriji ili potkategoriji, ne bi trebao ni moći pristupiti gumbu koji vodi do ove datoteke, ali je zato moguće direktno upisati URL te pristupiti obrascu, zbog toga je potrebno provjeriti dozvole i na ovom obrascu, kako ih korisnici ne bi mogli zaobići.

Kreiranje tema slično je kao i kreiranje kategorija i potkategorija. Provjerava se da li postoji POST metoda poslana sa same datoteke, te ukoliko ne postoji, prikazuje se obrazac. U obrazac se unosi naslov teme te sadržaj. Potkategorija u kojoj će se kreirati ova nova tema, dobavljena je GET metodom iz URL-a te ju nije potrebno ručno odabrati.

Za razliku od opisa kategorija i potkategorija, tekstualno polje sa sadržajem teme nije obično tekstualno polje. Koristi se poseban jQuery plugin naziva CKEDITOR koji pretvara odabrano tekstualno polje u WYSIWYG editor. On omogućuje izmjenu fontova, boja, unos tablica, lista, stvaranje hiperveza, prilaganje slika, video, emoticoni i slično.

Kako bi se tekstualno polje pretvorilo u WYSIWYG editor, nakon koda za tekstualno polje upisuje se skripta koja dodjeljuje editor polju sa određenom CSS klasom.

Isječak 31: kod za zamjenu tekstualnog polja s WYSIWYG editorom

```
<form method="post" action="">
<input class="topic_name" placeholder="subject" type="text"
name="topic_subject" required>

<textarea class="topic_content" placeholder="Topic content"
name="post_content"></textarea>
<script>
    CKEDITOR.replace("post_content");
</script>
<input class="submit_topic" type="submit" value="CREATE" />
</form>
```

Nakon što korisnik pošalje ispunjen obrazac, POST metodom podaci se šalju nazad na istu stranicu te se upitom podaci upisuju u bazu podataka. Ukoliko dođe do greške prilikom upisa, korisnika se obavještava. Nakon uspješnog unosa, dobavlja se novokreirani identifikator teme te se korisniku prikazuje poruka o uspješnom unosu zajedno sa hipervezom na svoju novu temu.

Isječak 32: upit kojim se unosi sadržaj nove teme u bazu

```
$sql = "INSERT INTO
    topics(topic_subject,
    topic_content,
    topic_date,
    last_post_date,
    topic_subcat,
    topic_by)
    VALUES('" . mysqli_real_escape_string($conn, $_POST['topic_subject']) .
    "',
    '" . mysqli_real_escape_string($conn, $_POST['post_content']) . "',
    NOW(),
    NOW(),
    '" . mysqli_real_escape_string($conn, $_GET['id']) . "',
    '" . mysqli_real_escape_string($conn, $_SESSION['user_id']) . "
    ");

$result = mysqli_query($conn, $sql);
if(!$result)
{
    echo 'An error occurred while inserting your data. Please try again later.<br
    /><br />' . mysqli_error($conn);
}

else
{
    $topicid = mysqli_insert_id($conn);
    echo 'You have successfully created <a href="topic.php?id=' . $topicid .
    '">your new topic</a>.';
}
```

Slika 16: Obrazac za kreiranje nove teme

Post a **Glavna jela** topic in **Recepti**

subject

Rich text editor toolbar: Bold (B), Italic (I), Strikethrough (ABC), Text color (I\_x), Bulleted list, Numbered list, Link, Unlink, Image, Table, Undo, Redo, Source, Styles, Normal, Font, Size, A, A, ?

body p

CREATE

Izvor: Autor

### 6.6.10. Tema

Pokretanjem `topic.php` izvršava se kod koji u bazu upisuje broj pregleda pojedine teme. Također se i izvodi upit koji provjerava dozvole korisnika te ovisno o tome prikazuje ili ne prikazuje temu. Kao i `subcategory.php` datoteka, `topic.php` datoteka sadrži funkciju za izračunavanje relativnog vremena, funkciju za ispis paginacije te breadcrumbs navigaciju.

Upitom u bazu podataka, dobivljaju se podaci o temi, autoru teme, kategoriji i potkategoriji. Identifikacijski broj teme iz URL-a spremljen je u `$id` varijablu koja služi za odabir točno potrebnih podataka.

Isječak 33: upit kojim se dobavljaju podaci potrebni za prikaz teme

```
$sql = "SELECT topics.*, subcategories.*, categories.cat_name, users.*,
groups.group_user_title
        FROM topics
        LEFT JOIN subcategories
        ON topics.topic_subcat = subcategories.subcat_id
LEFT JOIN categories
ON subcategories.subcat_cat = categories.cat_id
        LEFT JOIN users
        ON topics.topic_by = users.user_id
        LEFT JOIN groups
        ON users.user_group = groups.group_id
        WHERE topics.topic_id = " . $id;

$result = mysqli_query($conn, $sql);
$rs = mysqli_query($conn, $sql);
```

Podaci dobavljeni iz upita se prikazuju korisniku. Podaci su prikazani u pravokutnom elementu. Iznad elementa nalazi se naslov teme te broj odgovora na temu. Na lijevoj se strani elementa nalaze informacije o autoru, točnije slika, ime i grupa kojoj pripada. Gornji dio elementa prikazuje datum kreiranja teme te redni broj teme koji je uvijek #1 kako bi korisnik znao da je na samom početku. Donji dio elementa prikazuje autorov potpis te ukoliko je sadržaj teme izmijenjen, prikazuje se datum izmjene te ime korisnika koji je napravio izmjenu. Također se u donjem dijelu, ovisno o dozvolama korisnika, nalaze gumbi za odgovor sa citatom, izmjenu sadržaja i brisanje teme. U samoj se sredini elementa nalazi sadržaj teme. Ispod elementa koji prikazuje informacije o temi, nalazi se gumb za odgovor na temu, ukoliko korisnik nije prijavljen, prikazuje se drugi gumb koji služi za prijavu.

Isječak 34: programski kod za prikaz elementa koji sadrži podatke o temi

```
echo '<div id="op" class="post_container">';
echo '<div class="post_userinfo">';
if ($row['user_pic'])
```

```

{
echo '<div class="topic_user_image" style="background-
image:url('.$row['user_pic'].')"></div>';

        }
        else{
                echo '</img>';
        }
echo '<div class="post_username"> <a
href="user.php?id='.$row['user_id'].'">'. $row['user_name'] . '</a></div>';
echo '<div class="post_user_title">'. $row['group_user_title'] . '</div>';
echo '</div>';

echo '<div class="post_body">';
echo '<div class="post_date">'. $my_relative_time . '</div>';
echo '<div class="post_num" id="post1">';
echo '<a href="#post1">#1</a>';
echo '</div>';
echo '<div class="topic_content" id="'.$row['topic_id'].'" >
        '.
htmlspecialchars_decode(stripslashes($row['topic_content'])) .'
</div>';

if($row['topic_edit_by']){echo '<div class="edit_by"><i>Last edited by:
'.$row['topic_edit_by'].' - '.$relative_edit_time.'</i></div>';}

if(isset($_SESSION['user_id']) == $row['topic_by'] && $edittopic == 1 ||
$modpriv == 1) {
echo '<button class="edit_topic">Edit</button>';
}
if(isset($_SESSION['user_id']) == $row['topic_by'] && $deletetopic == 1 ||
$modpriv == 1){
echo '<button class="delete_topic">Delete</button>';
}
if($per_row['cat_reply'] == 1){
        if($per_row['post_replies'] == 1){

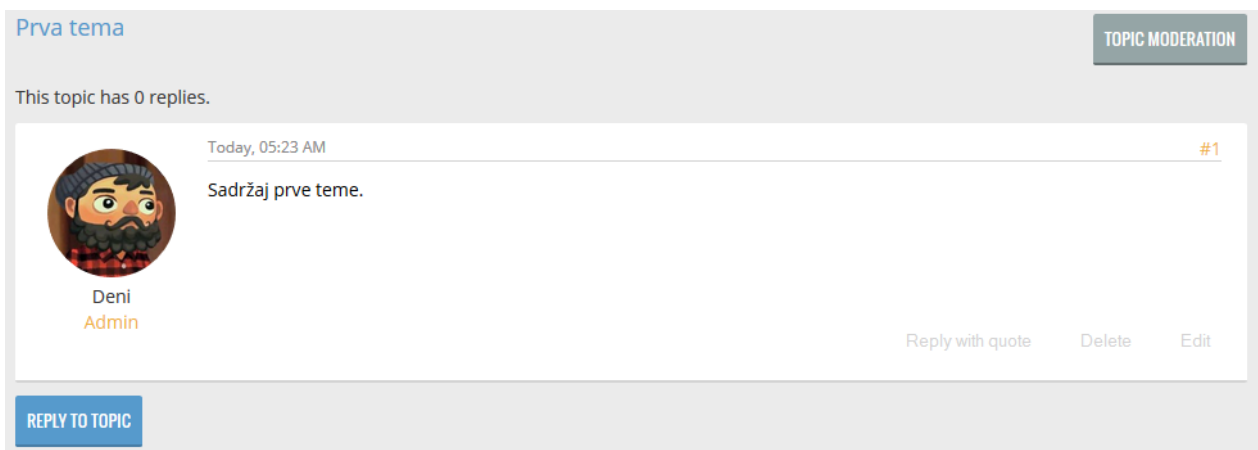
```

```

echo '<button class="reply_topic_quote">Reply with quote</button>';
    }}
if($row['user_signature'])
{
echo '<div class="user_signature">'.$row['user_signature'].'</div>';
}echo '</div>';
echo '</div>';
}

```

Slika 17: Prikaz objavljene teme



Izvor: Autor

Ukoliko autor teme želi izmijeniti ili izbrisati sadržaj teme, može to učiniti putem gumbića unutar elementa.

Izmjena teme vrši se korištenjem jQuerya i Ajaxa. Nakon što korisnik pritisne „Edit“ gumb, pokreće se jQuery skripta. U toj se skripti registrira klik miša na gumbu te se pokreće funkcija koja dobavlja trenutni URL, te iz njega izdvaja identifikacijski broj teme. Skripta zatim pronalazi najbliži element klase „topic\_content“, tj. element koji drži sadržaj teme, te pomoću jQuery Ajax metode `.load()`, šalje identifikacijski broj teme datoteci `edit_topic.php`. Skripta zatim sakriva sadašnji sadržaj te prikazuje sadržaj `edit_topic.php` stranice. Također sakriva sve ostale gumbiće.

Isječak 35: jQuery skripta za izmjenu sadržaja teme

```
$(document).ready(function() {
    $('.edit_topic').click(function() {
        var url = window.location.href;
        var hash = url.substring(url.indexOf('id'));
        var splitHash = hash.split('#');
        var getID = splitHash[0];
        $(this).siblings('.topic_content').load('edit_topic.php?' + getID,
function() {
            $(this).closest('.topic_content').hide().fadeIn(800);
        });
        $('.edit_post').fadeOut('slow');
        $('.edit_topic').fadeOut('slow');
        $('.delete_post').fadeOut('slow');
        $('.delete_topic').fadeOut('slow');
    });
});
```

Unutar `edit_topic.php` vrši se provjera dozvola korisnika koji joj pristupa te se ispisuje obrazac. Unutar tekstualnog polja obrasca ispisuje se sadržaj teme koji se upitom dobavlja iz baze podataka pomoću identifikatora teme poslanim jQuery-em. Sve se događa „in-line“ tako da korisnik uopće ni u jednom trenutku ne napušta stranicu. Također se ispisuju dva nova gumbića, „Save“ i „Cancel“ koji služe za spremanje ili poništenje promjena.

Isječak 36: obrazac koji prikazuje WYSIWYG editor prilikom ažuriranja sadržaja teme

```
echo '<form class = "topicform" method="post"
action="save_topic_edit.php?id=' . mysqli_real_escape_string($conn,
$_GET['id']) . '">
    <textarea id="edit2" name="post-edit">'.
htmlspecialchars_decode(stripslashes($row['topic_content'])) .
'</textarea><br />
    <script>
        CKEDITOR.replace("edit2");
```

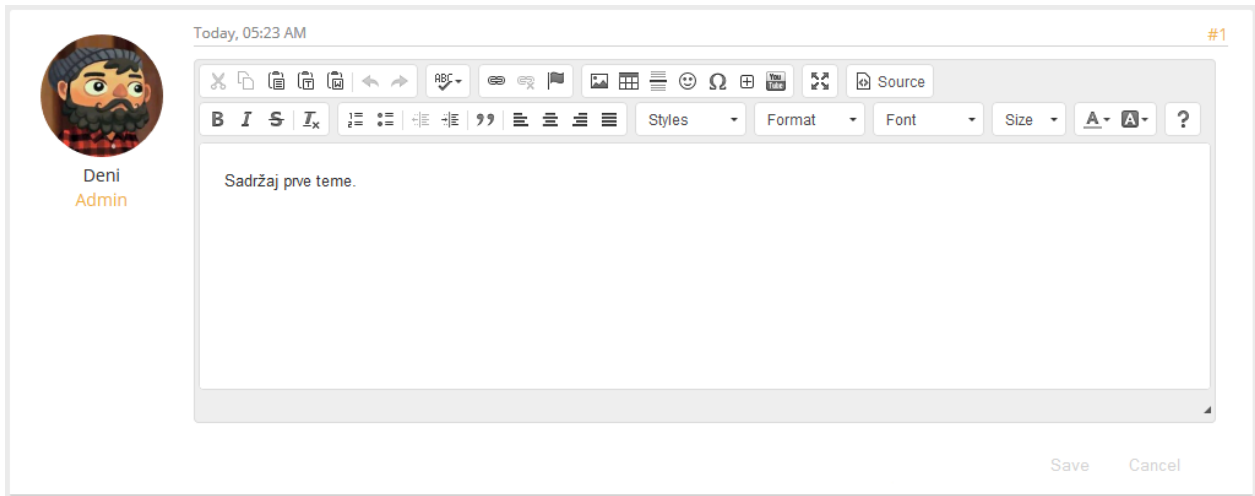


```

</script>
<button class="cancel_topic" type="button">Cancel</button><input
class="save_btn" type="submit" value="Save" />
</form>;

```

Slika 18: Primjer obrasca za ažuriranje sadržaja teme



Izvor: Autor

Ukoliko korisnik pritisne „Cancel“ gumb, pokreće se nova, slična skripta. U ovoj se skripti na isti način kao i u prijašnjoj, dobavlja identifikator iz URL-a te se sadržaj datoteke `.load()` metodom zamjenjuje na stranici, jedino što se ovog puta identifikator šalje `topic_content.php` datoteci te se originalni gumbi ponovno prikazuju.

Isječak 37: jQuery skripta za poništenje ažuriranja teme

```

$(document).on('click', '.cancel_topic', function(){
    var url = window.location.href;
    var hash = url.substring(url.indexOf('#'));
    var splitHash = hash.split('#');
    var getID = splitHash[0];
    var post = $(this).closest('.topic_content').attr('id');
    $(this).parents('.topic_content').load('topic_content.php?'+getID,
function(){

```

```

        $(this).closest('.topic_content').hide().fadeIn(800);
    });

    $('.edit_post').fadeIn('slow');
    $('.edit_topic').fadeIn('slow');
    $('.delete_post').fadeIn('slow');
    $('.delete_topic').fadeIn('slow');
});

```

Unutar `topic_content.php` datoteke se jednostavno šalje upit u bazu koji samo vraća stari neizmijenjeni sadržaj teme.

Isječak 38: `topic_content.php`

```

$sql2 = "SELECT topic_content
        FROM topics
        WHERE topic_id='".mysqli_real_escape_string($conn, $_GET['id']).'"
        ";

$result2 = mysqli_query($conn, $sql2);
while($row2 = mysqli_fetch_assoc($result2))
{
    echo ''. htmlspecialchars_decode(stripslashes($row2['topic_content'])) . ' ';
}

```

Ukoliko korisnik pritisne „Save“ gumb, pokreće se jQuery skripta. U ovoj se skripti detektira pritisak gumba „Save“ koji je dio obrasca. Iz URL-a se dobavlja identifikator teme te se pomoću `.ajax()` metode serijalizacijom šalju podaci iz obrasca te URL iz `action` atributa obrasca. URL sadrži adresu `save_topic_edit.php` datoteke. Ukoliko se upit unutar `save_topic_edit.php` datoteke uspješno izvrši, skripta pokreće `topic_content.php` datoteku kao i kod „Cancel“ gumba. Ako je došlo do greške, korisnika se obavještava.

Isječak 39: jQuery skripta sa spremanje izmijenjenog sadržaja teme

```
$(document).on('submit', '.topicform', function(e){
    e.preventDefault();
    var url = window.location.href;
    var hash = url.substring(url.indexOf('#'));
    var splitHash = hash.split('#');
    var getID = splitHash[0];
    $.ajax({
        type      : "POST",
        cache     : false,
        url       : $(this).attr('action'),
        data      : $(this).serialize(),
        success   : function(html){
            if(jQuery.trim(html) == 'Error'){
                $('.alert').text('There was an
error').fadeIn().delay(3000).fadeOut()
            }
            else{
                $('.alert').text('Topic edit
saved').fadeIn().delay(3000).fadeOut();

                $('.topicform').closest('.topic_content').load('topic_content.php?'+getID,
function(){
    $(this).closest('.topic_content').hide().fadeIn(800);
});
                $('.edit_post').fadeIn('slow');
                $('.edit_topic').fadeIn('slow');
                $('.delete_post').fadeIn('slow');
                $('.delete_topic').fadeIn('slow');
            }
        }
    });
});
```

Unutar `save_topic_edit.php` datoteke vrši se provjera dozvola te se izvodi upit u bazu koji zamjenjuje sadržaj teme te upisuje datum i ime korisnika koji je napravio izmjenu.

Isječak 40: `save_topic_edit.php`

```
$sql1 = "SELECT topic_by
        FROM topics
        WHERE topic_id='".mysqli_real_escape_string($conn,
$_GET['id']).'";
$result1 = mysqli_query($conn, $sql1);

while($row = mysqli_fetch_assoc($result1)){
    if(isset($_SESSION['user_id']) == $row['topic_by'] && $edittopic == 1 ||
$modpriv == 1){
        $sql = "UPDATE topics
                SET topic_content = '".mysqli_real_escape_string($conn,
$_POST['post-edit']).',
                topic_edit_date = NOW(),
                topic_edit_by = '".mysqli_real_escape_string($conn,
$_SESSION['user_name']).'
                WHERE topic_id = "'.mysqli_real_escape_string($conn,
$_GET['id']).';
$result = mysqli_query($conn, $sql)or die('Error');}
else{
    echo 'Wrong turn space cowboy';
}}
```

„Delete“ gumb pokreće jQuery skriptu kojom se briše tema. Prvo se korisnika upita potvrdim prozorom ako je siguran da želi obrisati temu. Ukoliko je siguran, identifikacijski broj teme šalje se `delete_topic.php` datoteci. Nakon upita unutar datoteke, korisnika se obavještava o rezultatu.

Isječak 41: jQuery skripta za brisanje teme

```
$(document).ready(function(){
    $('.delete_topic').click(function(){
        var topic = $(this).siblings('.topic_content').attr('id');
        var firm = confirm("Are you sure you want to DELETE this
TOPIC?");
        if (firm == true){
            $.ajax({
                type: "POST",
                url: 'delete_topic.php?id='+topic,
                success: function(html){
                    if(jQuery.trim(html) == 'Error'){
                        $('.alert').text('There was an
error').fadeIn().delay(3000).fadeOut()
                    }
                }
            })
        }
        else{
            $('.alert').text('Topic
deleted').fadeIn().delay(3000).fadeOut();
            $('.post_container').hide();
            $('.reply_container').hide();
            $('.pagination').hide();
        }
    });
});
```

Isječak 42: delete\_topic.php

```
if(isset($_SESSION['user_id']) == $row['topic_by'] && $deletetopic == 1 ||
$modpriv == 1){
    $sql = "DELETE FROM topics
            WHERE      topic_id      =      '".mysqli_real_escape_string($conn,
$_GET['id'])."'";

    $result = mysqli_query($conn, $sql) or die("Error");
}
else{
    echo 'Wrong turn space cowboy';
}
```

### 6.6.11.      Odgovor

Svaka tema može sadržavati beskrajn broj odgovora. Odgovori su poredani od najstarijeg do najnovijeg ispod sadržaja teme te se prikazuju na isti način kao i sadržaj teme, osim što se podaci dobivljaju iz druge tablice u bazi podataka. Ukoliko korisnik želi odgovoriti na neku temu, mora biti prijavljen u sustav foruma.


Unutar neke teme, nalazi se gumb „Reply to topic“ koji omogućava korisniku da odgovori na temu. Pritiskom tog gumba pokreće se jQuery funkcija kojom se prikazuje obrazac za ostavljanje odgovora. Unutar funkcije jQuery `.slideToggle()` metodom prikazuje se ili sakriva obrazac na način da se animira klizanje po visini elementa. Nakon završetka animacije, pozicija u web pregledniku pomiče se na sam obrazac.

Isječak 43: jQuery skripta za prikaz obrasca za kreiranje novog odgovora

```
$(document).ready(function() {
    $('.reply_content').hide();
    $('.reply_to_topic_btn').click(function() {
        $('.reply_content').slideToggle(800, function() {
            $('html, body').animate({
                scrollTop: $("#reply_content").offset().top
            });
        });
    });
});
```

Slika 19: Primjer obrasca za kreiranje novog odgovora

04 September 2014, 05:23 AM #1


 Deni Admin

Sadržaj prve teme.

*Last edited by: Deni - 04 September 2014, 06:35 AM*

Reply with quote Delete Edit

REPLY TO TOPIC

 Deni

Rich text editor toolbar: Bold, Italic, Strikethrough, Link, Unlink, List, Quote, etc.

SUBMIT REPLY

Izvor: Autor

Nakon što korisnik unese sadržaj te pritisne na gumb „Submit reply“, pokreće se jQuery skripta kojom se pomoću Ajax šalju uneseni podaci datoteci `reply.php`. Unutar te datoteke, `if...else` izjavama provjerava se da li je sadržaj obrasca prazan, ukoliko nije izvršava se upit kojim se upisuju podaci u bazu podataka te se ispisuje novi identifikacijski broj odgovora. Ako dođe do greške prilikom upisa, ispisuje se upozorenje.

Datoteka vraća rezultat te jQuery skripta nastavlja s izvršavanjem. Ukoliko je rezultat prazan, korisnika se obavještava da ispunjeno polje obrasca ne smije biti prazno, ako je rezultat „Error“, korisnik je obavješten da je došlo do greške prilikom upisa. U slučaju da datoteka vraća rezultat u obliku novog identifikacijskog broja odgovora, jQuery skripta šalje taj podatak `reply_content.php` datoteci.

Unutar `reply_content.php` datoteke, pomoću identifikacijskog se broja vrši upit u bazu te se dobavljaju podaci o odgovoru koji se ispisuju u istom obliku kao i ostali odgovori unutar teme.

Skripta dobavlja sadržaj datoteke te nakon zadnjeg odgovora unutar teme, prikazuje novostvoreni odgovor. Sadržaj unutar obrasca se briše te se obrazac sakriva.

Isječak 44: jQuery skripta za slanje sadržaja odgovora

```
$(document).on('submit', '.reply_form', function(e){
    e.preventDefault();
    var closestdiv =
$($('.reply_container').siblings('.post_container').last());
    var postNum =
$(closestdiv).children('.post_body').children('.post_num').attr('id');
    var splitNum = postNum.split('post');
    var idNum = parseInt(splitNum[1], 10);
    $.ajax({
        type      : "POST",
        cache     : false,
        url       : $(this).attr('action'),
        data      : $(this).serialize(),
        success   : function(data){
            var newid = $.trim(data);
            if(newid == ''){
                $($('.alert').text('The field can not be
empty')).fadeIn().delay(3000).fadeOut()
            }
            else if(newid == 'Error'){
                $($('.alert').text('There was a problem, try
again')).fadeIn().delay(3000).fadeOut()
            }
            else{
                $.get('reply_content.php?id='+newid, function(content){
```



```

        var showContent =
$($('.reply_container').siblings('.post_container').last().after(content);
        $('#'+newid).closest('.post_container').hide().slideDown(800);
        $(showContent).siblings('.post_container').last().children('.post_body
').children('.post_num').append(idNum+++1).attr('id', 'post'+idNum);
        $('.reply_content').slideUp(800)
        $('.alert').text('Reply
posted').fadeIn().delay(3000).fadeOut();
        });    }
for (instance in CKEDITOR.instances){
CKEDITOR.instances[instance].setData("");
}} });});

```

Isječak 45: reply.php

```

$id = mysqli_real_escape_string($conn, $_GET['id']);

if($_POST['reply-content'] == ''){
}
else{
    mysqli_query($conn, "UPDATE topics SET last_post_date = NOW() WHERE
topic_id = $id") or die("Error");
    $sql = "INSERT INTO
        posts(post_content,
            post_date,
            post_topic,
            post_by)
        VALUES ('" . mysqli_real_escape_string($conn, $_POST['reply-
content']) . "',
            NOW(),
            $id,
            " . $_SESSION['user_id'] . ")";
    $result = mysqli_query($conn, $sql);
    if(!$result){
        echo 'Error';
    }
}

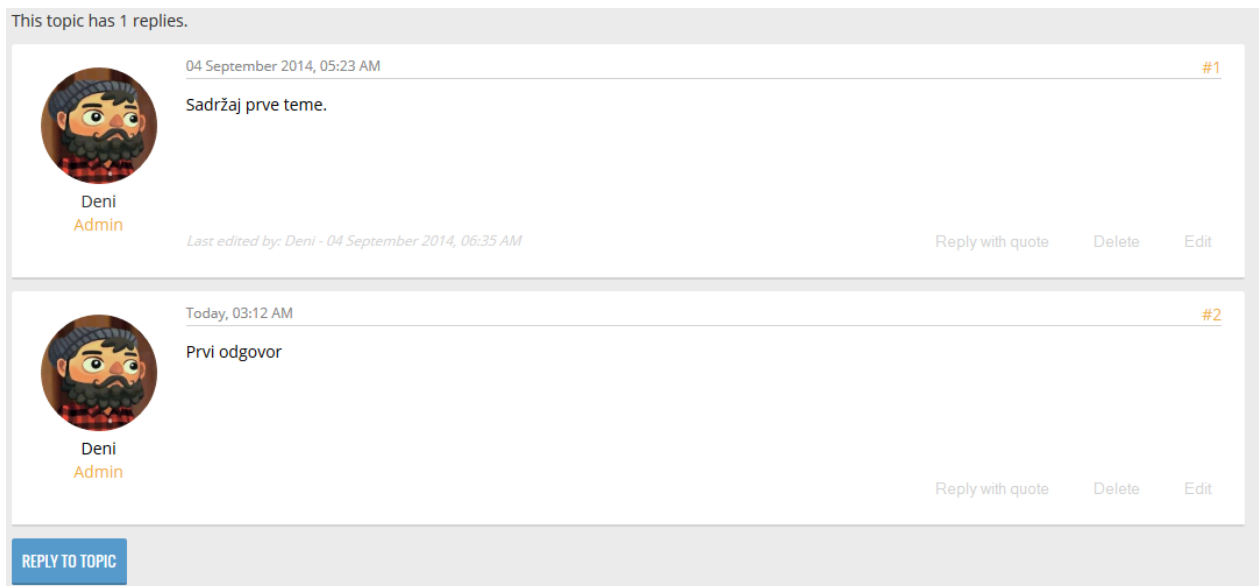
```

```

        else{
            $postid = mysqli_insert_id($conn);
            echo ''. $postid.'';
        }
    }
}

```

Slika 20: Primjer objavljenog odgovora na temu



Izvor: Autor

Unutar svake teme i odgovora prikazan je gumb „Reply with quote“ koji nudi korisniku da odgovori uz citat. Pritiskom na gumb pokreće se jQuery skripta koja šalje identifikacijski broj teme i odgovora `reply_post_quote_content.php` datoteci te se prikazuje sadržaj datoteke u obliku obrasca. Taj obrazac djeluje na isti način kao i prilikom stvaranja običnog odgovora samo što sadrži sadržaj odgovora ili teme koji se citira zajedno sa imenom autora i hipervezom na citiran sadržaj.

Isječak 46: jQuery skripta za odgovor uz citat

```
$(document).on('click', '.reply_post_quote', function(){
    var url = window.location.href;
    var hash = url.substring(url.indexOf('id'));
    var splitHash = hash.split('#');
    var getID = splitHash[0];
    var post = $(this).siblings('.post_content').attr('id');
    $('.reply_content').slideDown(800, function(){
        $('html, body').animate({
            scrollTop: $("#reply_content").offset().top
        });
    });
    $('.reply_form_wrap').load('reply_post_quote_content.php?'+getID+'&post='+post);
});
});
```

Isječak 47: reply\_post\_quote\_content.php

```
if(isset($_GET['page'])){
    $page = mysqli_real_escape_string($conn, $_GET['page']);
}
else{
    $page = 1;
}
$id = mysqli_real_escape_string($conn, $_GET['id']);
$post = mysqli_real_escape_string($conn, $_GET['post']);

$sql = "SELECT post_content, post_id, users.user_name
        FROM posts
        LEFT JOIN users
        ON posts.post_by = users.user_id
        WHERE post_id='".$post."'
        ";


$result = mysqli_query($conn, $sql);
while($row = mysqli_fetch_assoc($result))
{
```

```

    echo '<form class="reply_form" method="post" action="reply.php?id=' . $id
    . '">';
    echo '<textarea id="edit1" name="reply-content"><blockquote><div
class="quote_by">Originally posted by ' . $row['user_name'] . '<a
href="topic.php?id=' . $id . '&page=' . $page . '#' . $post . '"> &#10140;</a>
</div>'.
htmlspecialchars_decode(stripslashes($row['post_content'])) .
'</blockquote></textarea><br />';
    echo '<script>CKEDITOR.replace("edit1");</script>';
    echo '<input class="reply_btn" type="submit" value="SUBMIT REPLY" />
    ';
}

```

Slika 21: Primjer objavljenog odgovora uz citat



Deni

Admin

04 September 2014, 05:23 AM

#1


Sadržaj prve teme.

Last edited by: Deni - 04 September 2014, 06:35 AM

Reply with quote

Delete

Edit



Deni

Admin

Today, 03:12 AM


#2

Prvi odgovor

Reply with quote

Delete

Edit



Deni

Admin

Today, 05:55 PM

#3

Originally posted by Deni →

Prvi odgovor

Odgovor uz citat

Reply with quote

Delete

Edit

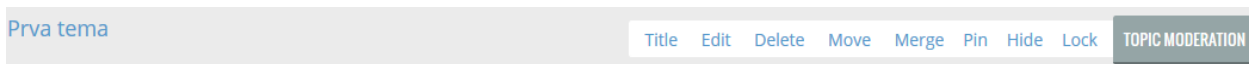
REPLY TO TOPIC

Izvor: Autor

## 6.6.12. Moderiranje foruma

Korisnik koji pripada grupi s moderatorskim privilegijama ima dodatne mogućnosti unutar tema. Unutar neke teme, za razliku od običnim korisnicima, prikazan mu je gumb naziva „Topic moderation“. Pritiskom na gumb, prikazuju se slijedeće dodatne opcije: izmjena naslova, izmjena sadržaja, brisanje, micanje teme u drugu potkategoriju, spajanje, pribadanje, sakrivanje te zaključavanje teme u obliku gumbova „Title“, „Edit“, „Delete“, „Move“, „Merge“, „Pin“, „Hide“ i „Lock“.

Slika 22: Gumb za moderiranje sa opcijama



Izvor: Autor

### 6.6.12.1. Izmjena naslova

Pritiskom „Title“ gumba, jQuery skriptom prikazuje se sadržaj `topic_title.php` datoteke koja sadrži obrazac s tekstualnim poljem unutar kojeg je unesen sam naslov teme.

Isječak 48: jQuery skripta za prikaz obrasca za izmjenu naslova teme

```
$(document).on('click', '#mod_title', function(){
    $('.topic_subject').load('topic_title.php?'+justID);
    $('.topic_moderation_content').delay(500).animate({width:
'toggle'});
});
```

Isječak 49: `topic_title.php`

```
$id = mysqli_real_escape_string($conn, $_GET['id']);
$sql = "SELECT topic_subject
FROM topics
WHERE topic_id = ".$id;
$result = mysqli_query($conn, $sql);
```

```

while($row = mysqli_fetch_assoc($result)){
    echo '<form class="change_topic_subject" method="post" action="">
        <input
                                id="change_topic_subject"
placeholder="'. $row['topic_subject']. '"      value="'. $row['topic_subject']. '"
type="text" name="merge_url" required>
        <input id="save_subject_btn" class="save_subject_btn" type="submit"
value="SAVE" />
    </form>

    ';
}

```

Slika 23: Primjer izmjene naslova teme

Izvor: Autor

Nakon što korisnik pritisne „Save“ gumb unutar obrasca, izmijenjeni naslov šalje se `save_subject_edit.php` datoteci koja jednostavno izmjenjuje podatak u bazi podataka. Također se pokreće jQuery skripta koja sakriva obrazac te prikazuje novi naslov.

Isječak 50: jQuery skripta za slanje izmijenjenog naslova teme

```

$(document).on('click', '.save_subject_btn', function(e){
    e.preventDefault();
    var subject = $('#change_topic_subject').val()
    $.ajax({
        type: "POST",
        url: 'save_subject_edit.php?'+justID+'&subject='+subject,
        success: function(html){
            if(jQuery.trim(html) == 'Error'){
                $('#.alert').text('There was an
                    error').fadeIn().delay(3000).fadeOut()
            }
            else{

```

```

        $('.alert').text('Title
changed').fadeIn().delay(3000).fadeOut();

        $('.topic_subject').html('<a
href="'+url+'"'+subject+'</a>');

    });});});});

```

### 6.6.12.2. Izmjena i brisanje

„Edit“ i „Delete“ gumbi unutar opcija moderiranja djeluju na isti način kao i gumbi unutar samog elementa teme koji su ranije objašnjeni.

### 6.6.12.3. Micanje teme

Pritiskom „Move“ gumba pokreće se jQuery skripta kojom se prikazuje novi prozor unutar kojeg se nalazi obrazac iz `move_topic.php` datoteke. Taj obrazac sadrži tekstualno polje i „Move“ gumb.

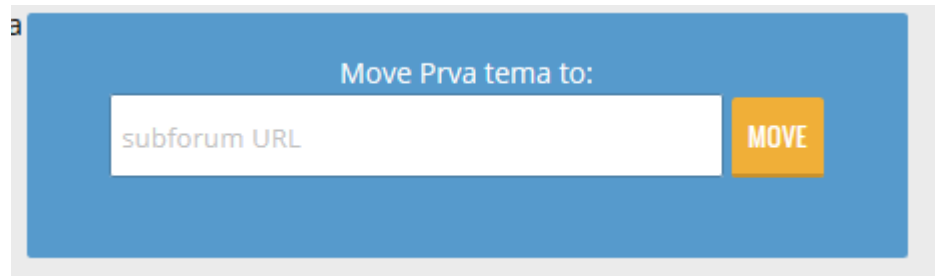
Isječak 51: jQuery skripta za prikaz obrasca za micanje teme

```

$(document).on('click', '#mod_move', function(){
    $('.alert').fadeIn().load('move_topic.php?'+justID);
    $('.topic_moderation_content').delay(500).animate({width:
'toggle'});
});

```

Slika 24: Primjer obrasca za micanja teme u drugu potkategoriju



Izvor: Autor

Unutar polja potrebno je upisati URL potkategorije u koju se želi tema pomaknuti. Pritiskom na gumb „Move“ pokreće se druga jQuery skripta koja iz unesenog URL-a izvlači identifikacijski broj potkategorije te šalje taj podatak `move_commit.php` datoteci koja jednostavno upitom u bazu izmjenjuje taj podatak za odabranu temu.

Isječak 52: jQuery skripta za izvlačenje identifikacijskog broja iz URLa

```
$(document).on('click', '.move_btn', function(e) {
    e.preventDefault();
    var moveURL = $('#move_url').val();
    var removeLocation = moveURL.split('id=');
    var removePage = removeLocation[1].split('&');
    var removeHash = removePage[0].split('#');
    var moveID = removeHash[0];
    $.ajax({
        type: "POST",
        url: 'move_commit.php?'+justID+'&subforum='+moveID,
        success: function() {
            $('#alert').text('Topic
moved').delay(1500).fadeOut(function() {
                location.reload();
            });});});});
```



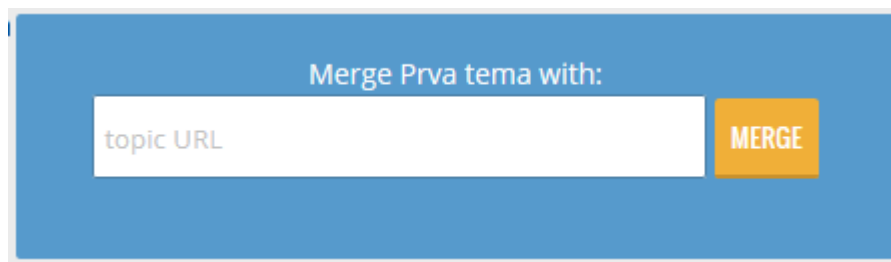
#### 6.6.12.4. Spajanje tema

Pritiskom na gumb „Merge“ pokreće se jQuery skripta koja prikazuje obrazac koji se nalazi unutar `merge_topic.php` datoteke. Obrazac sadrži tekstualno polje te „Merge“ gumb.

Isječak 53: jQuery skripta za prikaz obrasca za spajanje tema

```
$(document).on('click', '#mod_merge', function(){
    $('.alert').fadeIn().load('merge_topic.php?'+justID);
    $('.topic_moderation_content').delay(500).animate({width:
'toggle'});
});
```

Slika 25: Primjer obrasca za spajanje teme s drugom temom



Izvor: Autor

Unutar polja potrebno je upisati URL teme s kojom korisnik želi spojiti temu na kojoj se nalazi. Nakon što pritisne „Merge“ gumb, pokreće se druga jQuery skripta koja izvlači identifikacijski broj teme iz URL-a te šalje taj podatak `merge_commit.php` datoteci. Unutar datoteke vrše se upiti u bazu koji upisuju podatke o temi kao odgovor nove teme te se svim odgovorima unutar stare teme mijenja vanjski ključ teme kako bi se sad prikazali unutar odabrane teme iz URL-a. Stara se tema briše.

Isječak 54: jQuery skripta za izvlačenje identifikatora iz URLa

```
$(document).on('click', '.merge_btn', function(e){
    e.preventDefault();
```

```

var mergeURL = $('#merge_url').val();
var removeLocation = mergeURL.split('id=');
var removePage = removeLocation[1].split('&');
var removeHash = removePage[0].split('#');
var mergeID = removeHash[0];
$.ajax({
    type: "POST",
    url: 'merge_commit.php?'+justID+'&topic='+mergeID,
    success: function(){
        window.location.replace('topic.php?id='+mergeID);
    } }); });

```

Isječak 55: merge\_commit.php

```

$topic = mysqli_real_escape_string($conn, $_GET['topic']); //target topic
$id = mysqli_real_escape_string($conn, $_GET['id']); //current topic
$sql_topic = "SELECT * FROM topics WHERE topic_id = ".$id;
$result_topic = mysqli_query($conn, $sql_topic);
while($topic_row = mysqli_fetch_assoc($result_topic)){
    $insert_topic = "INSERT INTO posts(post_content,
        post_date,
        post_topic,
        post_by)
        VALUES ('".$topic_row['topic_content'].",
        '".$topic_row['topic_date'].",
        '".$topic_row['topic'].",
        '".$topic_row['topic_by'].")";
    $result = mysqli_query($conn, $insert_topic);}
$sql_posts = "SELECT *
        FROM posts
        WHERE post_topic = ".$id;
$result_posts = mysqli_query($conn, $sql_posts);
while($posts_row = mysqli_fetch_assoc($result_posts)){
    $insert_posts = "INSERT INTO
        posts(post_content,
        post_date,

```

```

        post_topic,
        post_by,
        post_edit_date,
        post_edit_by)
VALUES ('".$posts_row['post_content']."',
        '".$posts_row['post_date']."',
        '".$topic."',
        '".$posts_row['post_by']."',
        '".$posts_row['post_edit_date']."',
        '".$posts_row['post_edit_by']."'");

$result = mysqli_query($conn, $insert_posts);}
if($result != '' && !$tresult != ''){
$sql_delete_topic = "DELETE FROM topics
                    WHERE topic_id = ".$id;
mysqli_query($conn, $sql_delete_topic);
}

```

#### 6.6.12.5. Pribadanje, sakrivanje i zaključavanje

Sva tri gumba „Pin“, „Hide“ i „Lock“ rade na istom principu, samo što se u bazu upisuju vrijednosti u različite kolumne prilikom aktiviranja gumba. Način na koji rade sva tri gumba biti će prikazan kroz primjer gumba „Pin“.

Nakon što se gumb pritisne, pokreće se jQuery skripta koja šalje identifikacijski broj teme i indeks „pin“ s vrijednošću 1 `pin_topic.php` datoteci. Unutar datoteke vrši se upit u bazu koji mijenja podatak kolumne „topic status pinned“ iz 0 u 1. jQuery skripta nastavlja sa izvršavanjem te se pokraj naslova teme prikazuje novi element koji označava temu kao „Pinned“. Takva će tema uvijek biti na vrhu popisa tema unutar potkategorije u kojoj se nalazi. Skripta također mijenja gumb „Pin“ u gumb „Unpin“ koji onda radi na isti način kao i „Pin“ gumb samo što mijenja vrijednost kolumne iz 1 u 0.

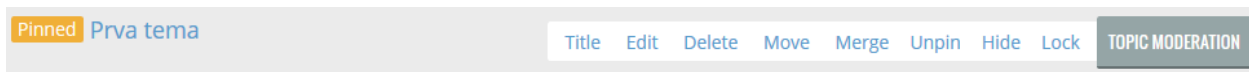
Isječak 56: jQuery skripta za pribadanje teme

```
$(document).on('click', '#mod_pin', function(){
    $.ajax({
        type: "POST",
        url: 'pin_topic.php?'+getID+'&pin=1',
        success: function(){
            $('.alert').text('Topic
pinned').fadeIn().delay(3000).fadeOut();
            $('.unpinned').addClass('pinned').fadeIn();
            $('#mod_pin').attr('id', 'mod_unpin').html('Unpin');
            $('.topic_moderation_content').delay(500).animate({width:
'toggle'}); } }); });
```

Isječak 57: pin\_topic.php

```
$sql = "UPDATE topics
    SET topic_status_pinned = '".mysqli_real_escape_string($conn,
$_GET['pin'])."'
    WHERE topic_id = '".mysqli_real_escape_string($conn, $_GET['id'])."'";
$result = mysqli_query($conn, $sql);
```

Slika 26: Primjer "Pinned" teme



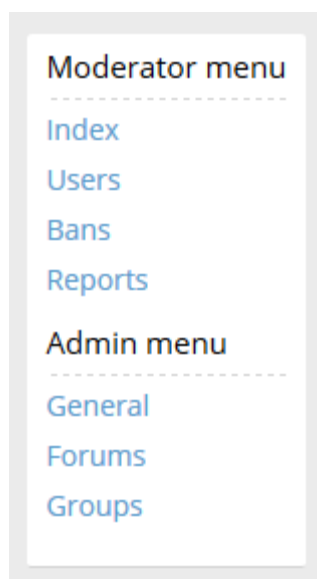
Izvor: Autor

### 6.6.13. Administracija sustava

Administratori i moderatori sustava Forum, u header dijelu web sjedišta imaju prikazanu hipervezu naziva „Control panel“. Ona vodi do posebnog dijela sustava Forum koji služi za upravljanje samim sustavom. Tamo je moguće pretraživati i ažurirati članove foruma te dodavati i ažurirati kategorije, potkategorije i grupe. Klik na „Control panel“ vodi korisnika na izbornik koji

je podijeljen u dva dijela, „Moderator menu“, kojem pristup imaju moderatori i administratori te „Admin menu“ kojem pristup imaju samo administratori.

Slika 27: Admin/moderator izbornik



Izvor: Autor

#### 6.6.13.1. Pregled i ažuriranje korisnika

Opcija „Users“ na moderatorskom dijelu control panela vodi korisnika do `acp_users.php` datoteke. Unutar datoteke prikazano je samo polje za unos korisničkog imena te „Search“ gumb. Nakon što korisnik klikne unutar polja za unos, automatski se pomoću jQuery-a generira popis svih korisnika te se svakim novim unešenim slovom lista filtrira na ona imena koja sadrže unesena slova. Lista je sortirana prema grupama.

Tražilica radi na način da kada korisnik pritisne neku tipku unutar polja za unos, automatski se poziva jQuery skripta koja uzima vrijednost iz polja te šalje tu vrijednost `user_search.php` datoteci. Unutar datoteke vrši se upit u bazu te se ispisuje rezultat upita u obliku liste korisnika koja se vraća nazad jQuery skripti te se prikazuje korisniku. Za svakog je korisnika ispisano korisničko ime, broj odgovora i tema te grupa kojoj pripada. Također su ponuđena dva gumba

pomoću kojih je moguće ažurirati profil korisnika ili ga izbrisati na slične načine kao i kod tema i odgovora.

Isječak 58: jQuery skripta za automatsko traženje korisnika

```
function searchq(){
    var searchText = $("input[name='search']").val();
    $.post("user_search.php", {searchVal: searchText}, function(output){
        $('#output').html(output);
    });
}
```

Isječak 59: user\_search.php

```
$output = '';
if(isset($_POST['searchVal'])){
    $searchq = $_POST['searchVal'];
    $searchq = preg_replace("#[^0-9a-z]#i","", $searchq);
    $sql = mysqli_query($conn, "SELECT users.*, groups.group_name FROM users
LEFT JOIN groups ON users.user_group = groups.group_id WHERE user_name LIKE
'%" . $searchq . "%'") or die("Error while searching");
    $count = mysqli_num_rows($sql);
    if($count == 0){
        $output = 'There are no search results';
    }
    else{
        while($row = mysqli_fetch_array($sql)){
            $name = $row['user_name'];
            $id = $row['user_id'];
            $group = $row['group_name'];
            $psql = mysqli_query($conn, "SELECT * FROM posts WHERE post_by =
'" . $id . "'");
            $pcount = mysqli_num_rows($psql);
            $tsql = mysqli_query($conn, "SELECT * FROM topics WHERE topic_by
= '" . $id . "'");
```

```

        $tcount = mysqli_num_rows($tsql);
        echo '<div class="acp_forums_row">
            <div class="acp_forums_row_data">
                <span class="user_search_row">'. $name. '</span>
                <span id="grey" class="user_search_row"> Posts:
'. $pcount. '</span>
                <span id="grey" class="user_search_row"> Topics:
'. $tcount. '</span>
                <span id="grey" class="user_search_row"> Group:
'. $group. '</span>
            <div class="inline_row">
                <span></span>
                <button class="forum_options">&#9998;</button>
                <button class="delete_subforum">&#10005;</button>
            </div>
        </div>
    </div>';
}}}

```

Na isti način djeluje i „Bans“ opcija izbornika control panela, jedino što tražilica pretražuje samo korisnike kojima je status u bazi zabilježen kao „banned“.

### 6.6.13.2. Pregled i ažuriranje kategorija i potkategorija

„Forums“ opcija admin izbornika control panela vodi korisnika do `acp_forums.php` datoteke. Ta datoteka prikazuje popis kategorija i potkategorija poput indeks stranice foruma. Za razliku od popisa na indeks stranici, `acp_forums.php` uz svaku kategoriju i potkategoriju prikazuje posebne gumbiće za ažuriranje istih. Prva dva gumba u obliku strijelica služe za pomicanjekategorije ili potkategorije za jedan nivo gore ili dolje kako bi se izmijenio redoslijed prikaza. Treći gumb u obliku olovke vodi na stranicu sa opcijama kategorije ili potkategorijedok dok četvrti gumb služi za brisanje. Osim tih gumbića, iznad popisa se nalaze dva gumba za dodavanje novih kategorija i potkategorija.

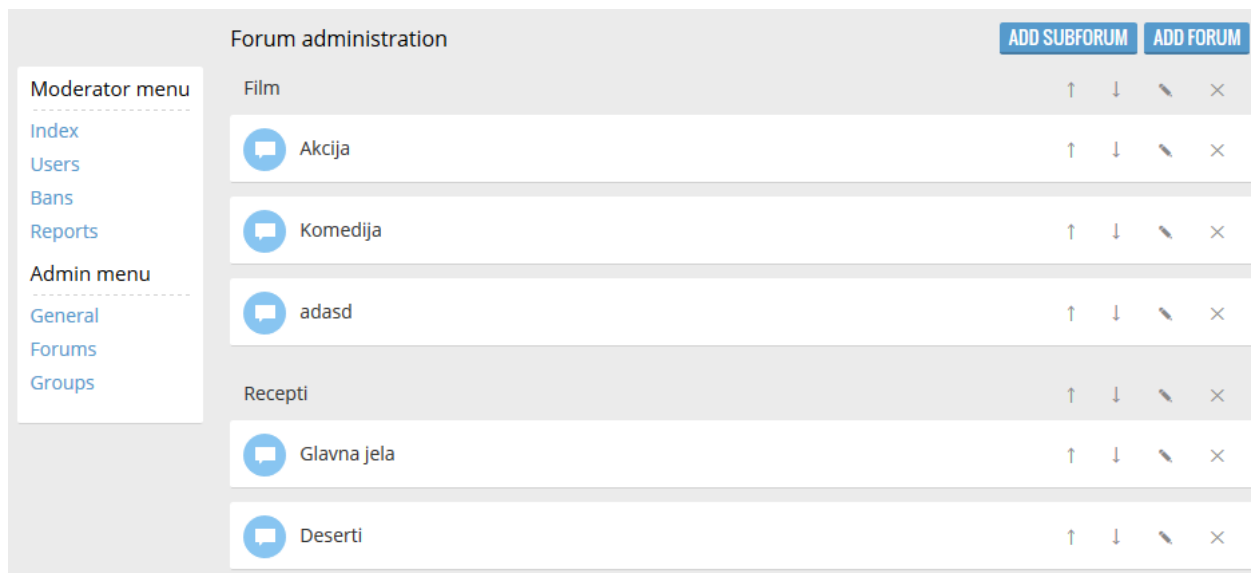
Pritiskom na jednu od gumba sa strijelicom poziva `forum_order.php` datoteku. Unutar datoteke vrši se upit s obzirom koja je strijelica stisnuta. Upit u bazu ažurira kolumnu `cat_order` ili `subcat_order` unutar tablica `categories` i `subcategories` u bazi podataka. U primjeru kada se stisne strijelica prema gore za neku potkategoriju, upit u bazu dobavlja trenutnu `subcat_order` vrijednost odabrane potkategorije, sprema tu vrijednost u varijablu te toj varijabli dodaje +1. Nakon toga slijedi upit koji povećava vrijednost `subcat_order` za +1 u bazi podataka te svim ostalim potkategorijama koje imaju jednaku vrijednost kao i odabrana potkategorija, oduzima vrijednost za -1.

Isječak 60: `forum_order.php`

```
if(isset($_GET['supid'])) {
    $forumID = mysqli_real_escape_string($conn, $_GET['supid']);
    $sql = "SELECT subcat_order FROM subcategories WHERE
subcat_id='$forumID'";
    $result = mysqli_query($conn, $sql);
    while($row = mysqli_fetch_assoc($result)) {
        $item = $row['subcat_order']+1;
        mysqli_query ($conn, "UPDATE subcategories SET subcat_order =
subcat_order +1 WHERE subcat_id = '$forumID'");
        mysqli_query ($conn, "UPDATE subcategories SET subcat_order =
subcat_order -1 WHERE subcat_order = '$item' AND subcat_id != '$forumID'");
    }
}
```



Slika 28: Popis kategorija i potkategorija u admin control panel-u



Izvor: Autor

Pritiskom gumba u obliku olovke vodi na `subforum_options.php` ili `forum_options.php` datoteku. Unutar tih datoteka prikazuju se informacije poput naziva i opisa te opcije sakrivanja kategorija ili potkategorija. Kod kategorija se nudi opcija za sakrivanje svih potkategorija dok se kod potkategorija nudi opcija premještanja potkategorije u drugu kategoriju.

Osim osnovnih postavki, prikazane su i dozvole kategorija i potkategorija koje se mogu ažurirati checkboxovima za pojedine grupe. Postoje dozvole o čitanju, stvaranju novih teme stvaranju novih odgovora unutar odabrane kategorije ili potkategorije. Izmjenom i prihvaćanjem promjena dozvola, šalju se podaci o označenim checkboxevima datoteci `subforum_perms_commit.php` ili `forums_perms_commit.php` datoteci u obliku niza. U toj se datoteci vrijednosti niza provjeravaju te ukoliko postoje, dodjeljuje im se vrijednost 1, ukoliko nisu pronađene (jer vrijednosti neoznačenih checkboxeva se ne šalju), dodjeljuje im se vrijednost 0 te se upisuju u bazu podataka.

```
function array_map_callback($a){
    global $conn;
    return mysqli_real_escape_string($conn, $a);
}
$read = 0;
$reply = 0;
$post = 0;
if(isset($_POST['read_forum'])){
    $read = array_map('array_map_callback', $_POST['read_forum']);
}
if(isset($_POST['reply_forum'])){
    $reply = array_map('array_map_callback', $_POST['reply_forum']);
}
if(isset($_POST['post_forum'])){
    $post = array_map('array_map_callback', $_POST['post_forum']);
}
$sql = "SELECT * FROM groups";
$result = mysqli_query($conn, $sql);
while($row = mysqli_fetch_assoc($result)){
    $group_id = $row['group_id'];
    $a = get_defined_vars();
    if($read && array_key_exists(''.$group_id.'', $_POST['read_forum'])){
        $read_forum = 1;    }
    else{
        $read_forum = 0;    }
    if($reply && array_key_exists(''.$group_id.'', $_POST['reply_forum'])){
        $post_replies = 1;    }
    else{
        $post_replies = 0;    }
    if($post && array_key_exists(''.$group_id.'', $_POST['post_forum'])){
        $post_topics = 1;    }
    else{
        $post_topics = 0;    }
```

Slika 29: Primjer ažuriranja potkategorije

Moderator menu  
[Index](#)  
[Users](#)  
[Bans](#)  
[Reports](#)  
Admin menu  
[General](#)  
[Forums](#)  
[Groups](#)

### Akcija - settings

Subforum name:
← change the forum name

Description:
← change the forum description

Subforum hidden:
☐ Yes ☒ No
 ← hide/show forum

Subforum category:

SUBMIT

### Akcija - permissions

	Read forum	Post replies	Post topics
Administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Moderators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Guests	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Members	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

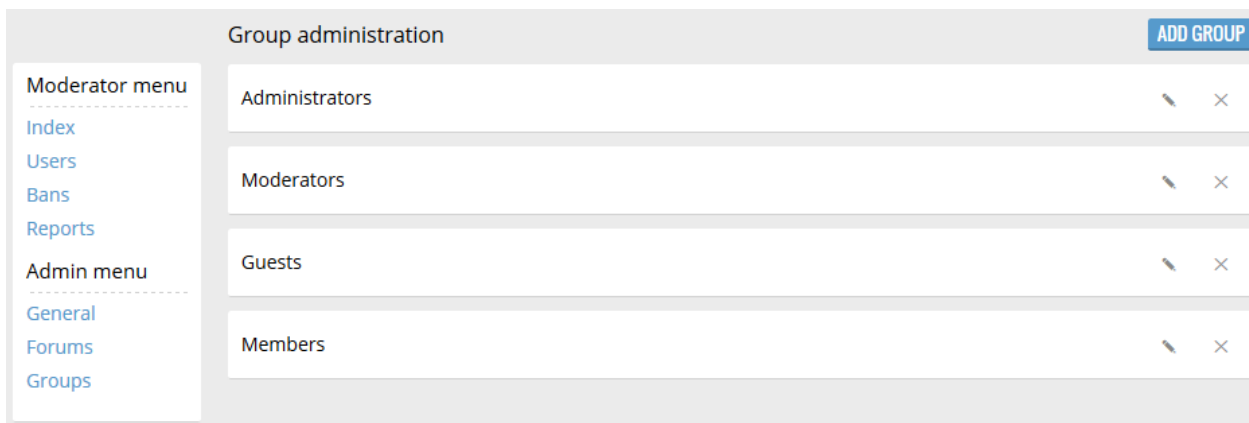
SAVE

Slika Izvor: Autor

### 6.6.13.3. Pregled i ažuriranje grupa

Groups opcija admin control panela prikazuje `acp_groups.php` datoteku unutar koje je prikazan popis grupa, gumb za izmjenu podataka grupe, brisanje grupe i gumb za stvaranje nove grupe. Obrasci za izmjenu i stvaranje novih grupa gotovo su isti, osim što jedan obrazac šalje upit za izmjenu postojećih podataka dok drugi te iste podatke prvi put upisuje u bazu.

Slika 30: Primjer administracije grupa



Izvor: Autor

Klik na gumb za ažuriranje grupe vodi korisnika na `edit_group.php` datoteku. Unutar datoteke prikazuje se obrazac za izmjenu naziva grupe, naslova članova grupe te dozvola grupe poput moderatorskih prava, čitanje foruma, stvaranje tema i odgovora te mogućnosti ažuriranja i brisanja tema i odgovora. Prihvaćanjem promjena i slanjem ispunjenog obrasca, podaci se šalju datoteci `edit_group_commit.php` koja jednostavno primljene podatke izmjenjuje u bazi podataka za odgovarajuću grupu.

Isječak 62: `edit_group_commit.php`

```
$id = mysqli_real_escape_string($conn, $_GET['id']);
$group_title = mysqli_real_escape_string($conn, $_POST['group_title']);
$user_title = mysqli_real_escape_string($conn, $_POST['user_title']);
$moderator_priv = mysqli_real_escape_string($conn,
$_POST['moderator_priv']);
$read_board = mysqli_real_escape_string($conn, $_POST['read_board']);
$post_replies = mysqli_real_escape_string($conn, $_POST['post_replies']);
$post_topics = mysqli_real_escape_string($conn, $_POST['post_topics']);
$edit_posts = mysqli_real_escape_string($conn, $_POST['edit_posts']);
$edit_topics = mysqli_real_escape_string($conn, $_POST['edit_topics']);
$delete_posts = mysqli_real_escape_string($conn, $_POST['delete_posts']);
$delete_topics = mysqli_real_escape_string($conn, $_POST['delete_topics']);
```

```

mysqli_query($conn, "UPDATE groups SET group_name = '$group_title',
group_user_title = '$user_title', group_moderator_priv = '$moderator_priv',
group_read_board = '$read_board', group_post_replies = '$post_replies',
group_post_topics = '$post_topics', group_edit_posts = '$edit_posts',
group_delete_posts = '$delete_posts', group_delete_topics =
'$delete_topics', group_edit_topics = '$edit_topics'
WHERE group_id = '$id'")
or die(mysqli_error($conn));

```

Slika 31: Primjer ažuriranja postavki grupe

**Moderator menu**  
[Index](#)  
[Users](#)  
[Bans](#)  
[Reports](#)  
**Admin menu**  
[General](#)  
[Forums](#)  
[Groups](#)

### Edit group Moderators

Group title:	<input type="text" value="Moderators"/>	
User title:	<input type="text" value="Mods"/>	
Allow moderator privileges:	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow moderator privileges to users in this group
Read board:	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to read the board
Post replies	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to reply to topics
Post topics	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to post new topics
Edit posts	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to edit their own posts
Delete posts	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to delete their own posts
Edit topics	<input type="radio"/> Yes <input checked="" type="radio"/> No	← allow users to edit their own topics
Delete topics	<input checked="" type="radio"/> Yes <input type="radio"/> No	← allow users to delete their own topics
		<input type="button" value="SUBMIT"/>

Izvor: Autor

## 7. Zaključak

Jedan od najboljih izbora za izradu web aplikacije je PHP programski jezik. Kombinirajući PHP sa JavaScriptom, jQuery-em, Ajaxom i mogućnostima HTML 5, prilikom izrade web aplikacije gotovo da ne postoje stvari koje nije moguće učiniti. Cijeli proces izrade aplikacije postaje lakši, te barem jedna od korištenih tehnologija uvijek ima rješenje za određeni problem. Korištene tehnologije uče se vrlo brzo, jednostavne su za primjenu, open-source su i potpuno besplatne te sadrže bogate izvore dokumentacije.

Prikazan proces izgradnje sustava internetskog foruma demonstrira kako je uz relativno malo količine programskog koda i pomoću besplatnih tehnologija, moguće izraditi dinamičnu, brzu i user-friendly web aplikaciju koja je potpuno funkcionalna, spremna i sigurna za korištenje. Korisnici aplikacije gotovo trenutačno primaju nenametljive povratne informacije o svojim postupcima unutar sustava, bez potrebe konstantnog osvježivanja stranica ili vraćanja unatrag prilikom greške. Pruženo korisničko sučelje dovoljno je jednostavno za korištenje ali i ima dovoljno funkcija kako bi korisnici mogli uživati u diskusijama i nadgledavanju istih.

## Literatura

1. Solid IT, DB-Engines Ranking, 2014., <http://db-engines.com/en/ranking> (28.08.2014)
2. The jQuery Foundation, What is jQuery?, 2010., <http://jquery.com/> (28.08.2014)
3. The PHP Group, History of PHP, 2008., <http://php.net/manual/en/history.php.php> (28.08.2014)
4. vins, Supported Database, 2009., [http://leiron.be/infodocs/supported\\_database.html](http://leiron.be/infodocs/supported_database.html) (28.08.2014)
5. W3C, Same Origin Policy, 2010., [http://www.w3.org/Security/wiki/Same\\_Origin\\_Policy](http://www.w3.org/Security/wiki/Same_Origin_Policy) (28.08.2014)

## Popis slika

Slika 1. ER model .....	11
Slika 2. Dijagram slučaja korištenja .....	12
Slika 3: Relacijski model baze podataka .....	13
Slika 4: Struktura web sjedišta .....	15
Slika 5: Primjer hijerarhijske struktura kategorija korisnika .....	17
Slika 6: Primjer hijerarhijske strukture kategorija tema .....	18
Slika 7: Dizajn stranica .....	19
Slika 8: Header stranice .....	21
Slika 9: Obrazac za registraciju .....	31
Slika 10: Obrazac za prijavu .....	34
Slika 11: Korisnički profil .....	36
Slika 12: Obrazac za kreiranje kategorije .....	42
Slika 13: Obrazac za kreiranje potkategorije .....	45
Slika 14: Početna stranica web sjedišta .....	50
Slika 15: Paginacija .....	54
Slika 16: Obrazac za kreiranje nove teme .....	61
Slika 17: Prikaz objavljene teme .....	64
Slika 18: Primjer obrasca za ažuriranje sadržaja teme .....	66
Slika 19: Primjer obrasca za kreiranje novog odgovora .....	72
Slika 20: Primjer objavljenog odgovora na temu .....	75
Slika 21: Primjer objavljenog odgovora uz citat .....	77
Slika 22: Gumb za moderiranje sa opcijama .....	78
Slika 23: Primjer izmjene naslova teme .....	79
Slika 24: Primjer obrasca za micanja teme u drugu potkategoriju .....	81
Slika 25: Primjer obrasca za spajanje teme s drugom temom .....	82
Slika 26: Primjer "Pinned" teme .....	85
Slika 27: Admin/moderator izbornik .....	86
Slika 28: Popis kategorija i potkategorija u admin control panel-u .....	90



Slika 29: Primjer ažuriranja potkategorije.....	92
Slika 30: Primjer administracije grupa .....	93
Slika 31: Primjer ažuriranja postavki grupe .....	94

## Popis tablica

Tablica 1. PHP podržane baze podataka .....	7
---	---

## Popis isječaka

Isječak 1: header.php .....	19
Isječak 2: include session.php .....	21
Isječak 3: session.php .....	22
Isječak 4: footer.php .....	23
Isječak 5: body primjer .....	23
Isječak 6: connect.php .....	24
Isječak 7: signup.php .....	25
Isječak 8: jQuery password validacija .....	27
Isječak 9: jQuery 1. dio slanje obrasca registracije .....	27
Isječak 10: register.php .....	28
Isječak 11: jQuery 2. dio obavijest registracije .....	30
Isječak 12: signin.php .....	32
Isječak 13: jQuery slanje i validacija signin obrasca.....	33
Isječak 14: user.php .....	34
Isječak 15: jQuery Ajax izmjena emaila.....	37
Isječak 16: obrazac za upload profilne slike.....	38
Isječak 17: upload.php .....	38
Isječak 18: create_cat.php.....	40
Isječak 19: create_subcat.php .....	43
Isječak 20: relative_date funkcija .....	46

Isječak 21: upit u bazu za prikaz kategorija i potkategorija na index.php.....	47
Isječak 22: while petlja za prikaz kategorija .....	47
Isječak 23: while petlja za prikaz potkategorija .....	48
Isječak 24: upit kojim se izračunava broj tema i odgovora unutar potkategorije.....	49
Isječak 25: pagination funkcija.....	51
Isječak 26: breadcrumbs navigacija.....	54
Isječak 27: upit za prikaz određen broj tema po stranici .....	55
Isječak 28: programski kod koji prikazuje status teme.....	56
Isječak 29: upit za prikaz mini paginacije pokraj naziva teme.....	56
Isječak 30: programski kod za generiranja nasumičnog broja pogleda teme .....	58
Isječak 31: kod za zamjenu tekstualnog polja s WYSIWYG editorom .....	59
Isječak 32: upit kojim se unosi sadržaj nove teme u bazu.....	60
Isječak 33: upit kojim se dobavljaju podaci potrebni za prikaz teme.....	62
Isječak 34: programski kod za prikaz elementa koji sadrži podatke o temi .....	62
Isječak 35: jQuery skripta za izmjenu sadržaja teme .....	65
Isječak 36: obrazac koji prikazuje WYSIWYG editor prilikom ažuriranja sadržaja teme .....	65
Isječak 37: jQuery skripta za poništenje ažuriranja teme .....	66
Isječak 38: topic_content.php .....	67
Isječak 39: jQuery skripta sa spremanje izmijenjenog sadržaja teme .....	68
Isječak 40: save_topic_edit.php.....	69
Isječak 41: jQuery skripta za brisanje teme.....	70
Isječak 42: delete_topic.php .....	70
Isječak 43: jQuery skripta za prikaz obrasca za kreiranje novog odgovora .....	71
Isječak 44: jQuery skripta za slanje sadržaja odgovora.....	73
Isječak 45: reply.php.....	74
Isječak 46: jQuery skripta za odgovor uz citat .....	76
Isječak 47: reply_post_quote_content.php .....	76
Isječak 48: jQuery skripta za prikaz obrasca za izmjenu naslova teme.....	78
Isječak 49: topic_title.php.....	78
Isječak 50: jQuery skripta za slanje izmijenjenog naslova teme .....	79

Isječak 51: jQuery skripta za prikaz obrasca za micanje teme .....	80
Isječak 52: jQuery skripta za izvlačenje identifikacijskog broja iz URLa .....	81
Isječak 53: jQuery skripta za prikaz obrasca za spajanje tema.....	82
Isječak 54: jQuery skripta za izvlačenje identifikatora iz URLa.....	82
Isječak 55: merge_commit.php.....	83
Isječak 56: jQuery skripta za pribadanje teme.....	85
Isječak 57: pin_topic.php.....	85
Isječak 58: jQuery skripta za automatsko traženje korisnika .....	87
Isječak 59: user_search.php.....	87
Isječak 60: forum_order.php.....	89
Isječak 61: subforum_perms_commit.php .....	91
Isječak 62: edit_group_commit.php .....	93

## Popis kratica

1. HTML – HyperText Markup Language
2. CSS – Cascading Style Sheets
3. PHP/FI – Personal Home Page/Forms Interpreter
4. Ajax – Asynchronous JavaScript and XML
5. XML – Extensible Markup Language
6. JSON – JavaScript Object Notation