

Filip Minić 138/17

## Dokumentacija

<https://mediumsocialnetwork.000webhostapp.com/index.php?page=pocetna>

Smer: Internet Tehnologije

Modul: Web programiranje

Predmet: Praktikum PHP

Beograd, Jun 2019

## Sadržaj

1.	Uvod: .....	5
1.1.	Korišćeni programski jezici .....	5
1.2.	Opis funkcionalnosti .....	5
1.3.	Templates .....	5
2.	Organizacija .....	6
2.1.	Organizaciona šema .....	6
2.2.	Sitemap.....	7
2.3.	Sema Baze .....	9
2.4.	Editor Slike .....	10
2.5.	Opis stranica sajta .....	14
	Index.php.....	14
	Page=pocetna .....	16
	Page=profilUser .....	17
	Page= editProfile .....	19
	Page=editPost.....	20
	Page=comment.....	21
	Page=adminPanel .....	22
	Page=autor .....	23
3.	Kodovi .....	24
	Index.php.....	24
3.1.	config .....	25
	.env .....	25
	Config.php .....	25
	Connection.php .....	25
3.2.	Models .....	27
3.2.1.	adminPanel.....	27
	Functions.php .....	27
3.2.2.	comments.....	30
	deleteComm.php.....	30
	Functions.php .....	30
	getComments.php.....	31
	getOneComment.php.....	32
	insertComment.php .....	32
	updateComment.php .....	33
3.2.3.	posts .....	35

deltePost.php .....	35
Functions.php .....	35
getAllCategories.php .....	37
getAllPosts.php.....	37
getAllPostsProfile.php .....	38
getAllPostsUser.php .....	38
getPost.php.....	38
insertPost.php .....	39
searchPost.php .....	41
updatePost.php .....	41
3.2.4 users .....	44
deleteUser.php.....	44
exportExcel.php.....	44
exportWord.php.....	45
Functions.php .....	46
Login.php .....	49
Logout.php .....	51
Registration.php .....	51
searchUser.php.....	52
updateBackgroundPhoto.php .....	53
updatePassword.php.....	54
updateProfile.php.....	55
updateProfilePhoto.php.....	56
updateRole.php .....	58
zabranaUser.php .....	60
3.3. Views .....	61
3.3.1 Fixed .....	61
Footer.php.....	61
Head.php .....	61
Nav.php .....	62
3.3.2. Pages.....	63
Admin.php.....	63
Autor.php .....	66
Comments.php .....	67
editPost.php .....	67
editProfile.php.....	69

loginRegister.php.....	74
Postovi.php.....	75
Postovi2.php.....	76
Profile_details.php .....	77
Profile_info.php.....	78
Profile_user_info.php.....	80
3.4 JavaScript.....	82

# 1. Uvod:

## 1.1. Korišćeni programski jezici

Za izradu sajta korišćeni su sledeći programski jezici:

- PHP
- PhpMyAdmin
- HTML
- CSS
- JavaScript
- jQuery
- Toastr
- Bootstrap

Kod je pisan u PHPStorm-u.

## 1.2. Opis funkcionalnosti

Sajt sadrži sledeće funkcionalnosti:

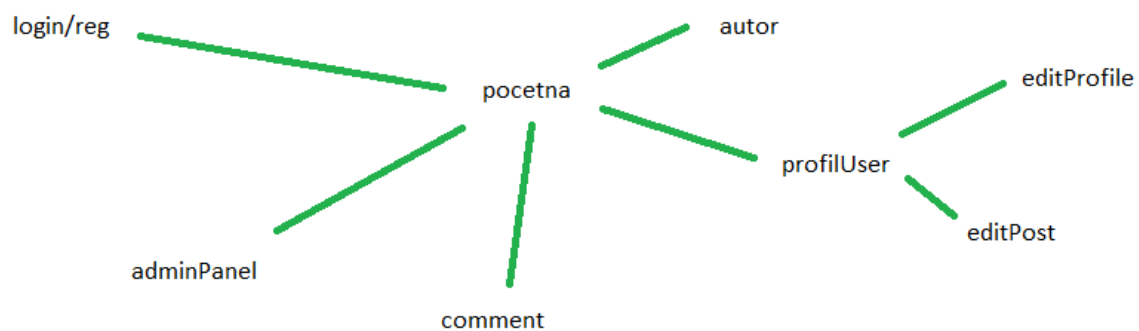
- Vise obkila filtriranja(search, kategorije)
- CRUD za korisnike(AJAX)
- CRUD za postove
- CRUD za komentare postova(AJAX)
- Paginaciju koja je radjena AJAX-om prilikom dolaska na kraj stranice
- Serverska i klijentska provera
- Login/Register korisnika
- Slanje maila korisniku ako neko pogresno ukuca sifru prilikom logovanja
- Upisivanje gresaka nad bazom u fajl
- Export u Word i Excel
- Racunanje aktivnosti korisnika u poslednja 24h u procentima
- Zapisivanje broja i podataka o aktivnim korisnicima
- Resize slike pomocu koda
- Zabrana prisupa preko url admin panelu i profilu korisnika(licni podaci)

## 1.3. Templates

Template je koricen na celom sajtu.

## 2. Organizacija

### 2.1. Organizaciona šema



## 2.2. Sitemap

```
3. <?xml version="1.0" encoding="UTF-8" ?>
  <urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
    <url>
      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.95</priority>
    </url>
    <url>

      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=pocetna</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.95</priority>
    </url>
    <url>

      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=profilUser</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.75</priority>
    </url>
    <url>

      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=editProfile</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.75</priority>
    </url>
    <url>

      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=editPost</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.75</priority>
    </url>
    <url>

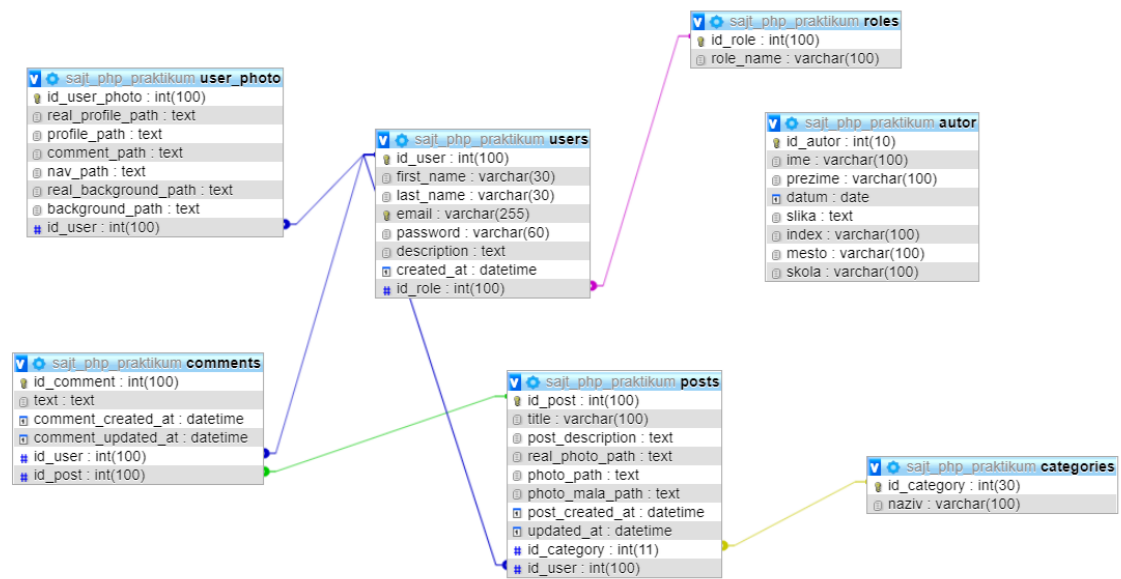
      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=comment</loc>
      <lastmod>2019-14-06</lastmod>
      <changefreq>monthly</changefreq>
      <priority>0.75</priority>
    </url>
    <url>

      <loc>https://mediumsocialnetwork.000webhostapp.com/index.php?page=autor</loc>
```

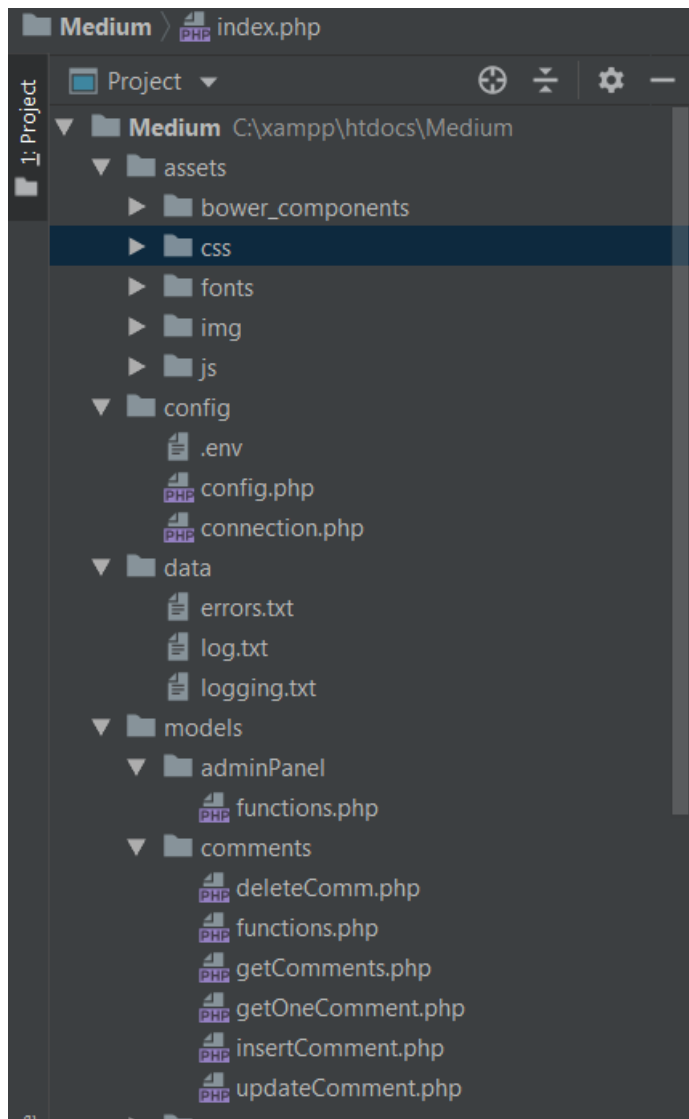
```
/loc>  
  <lastmod>2019-14-06</lastmod>  
  <changefreq>monthly</changefreq>  
  <priority>0.75</priority>  
</url>  
  
</urlset>
```

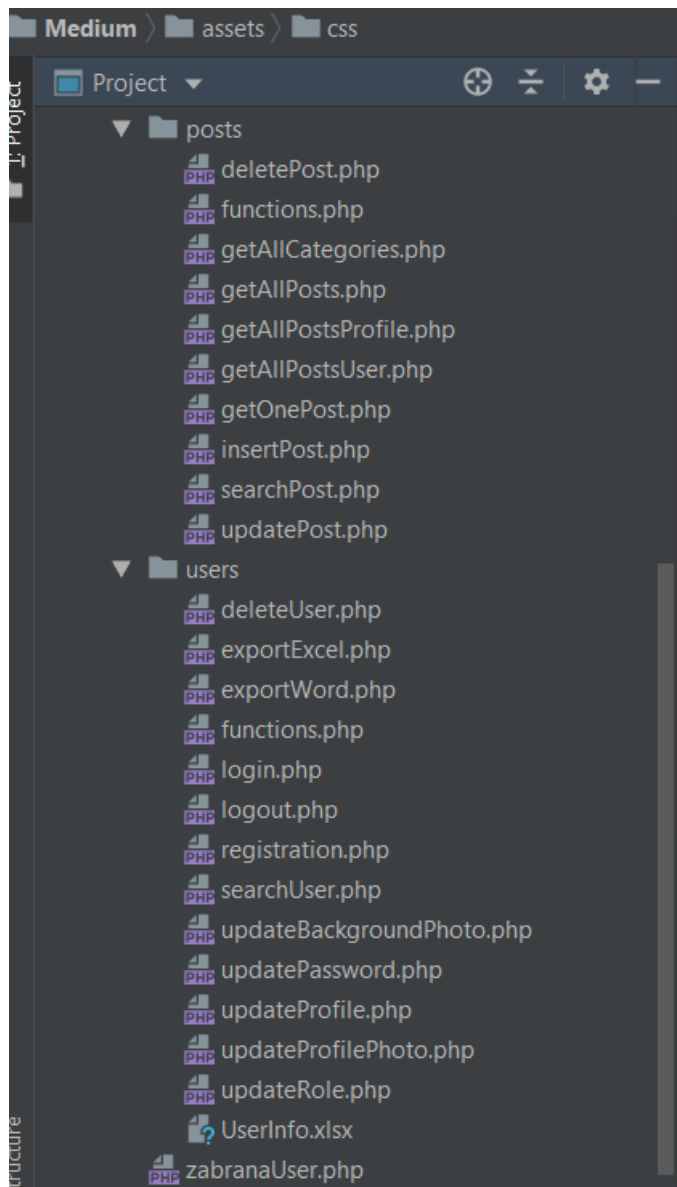


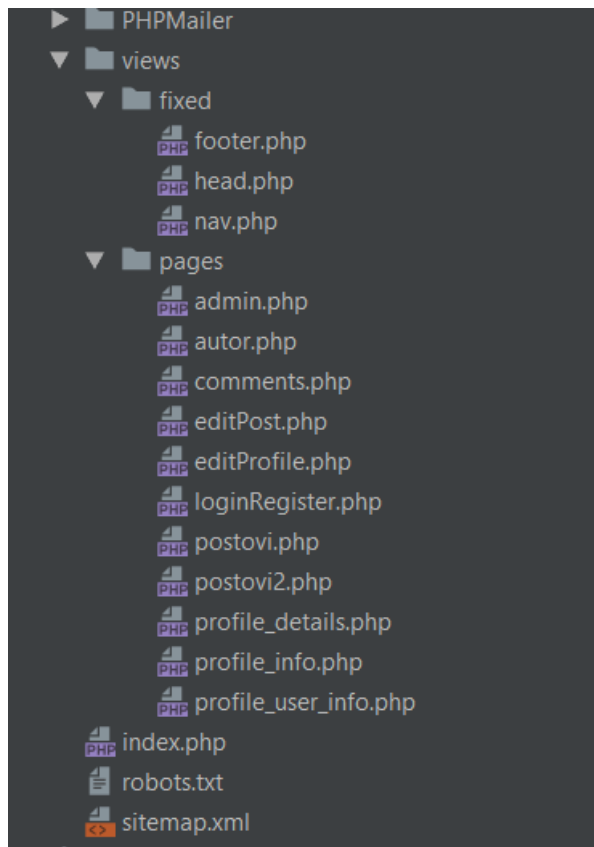
## 2.3. Sema Baze

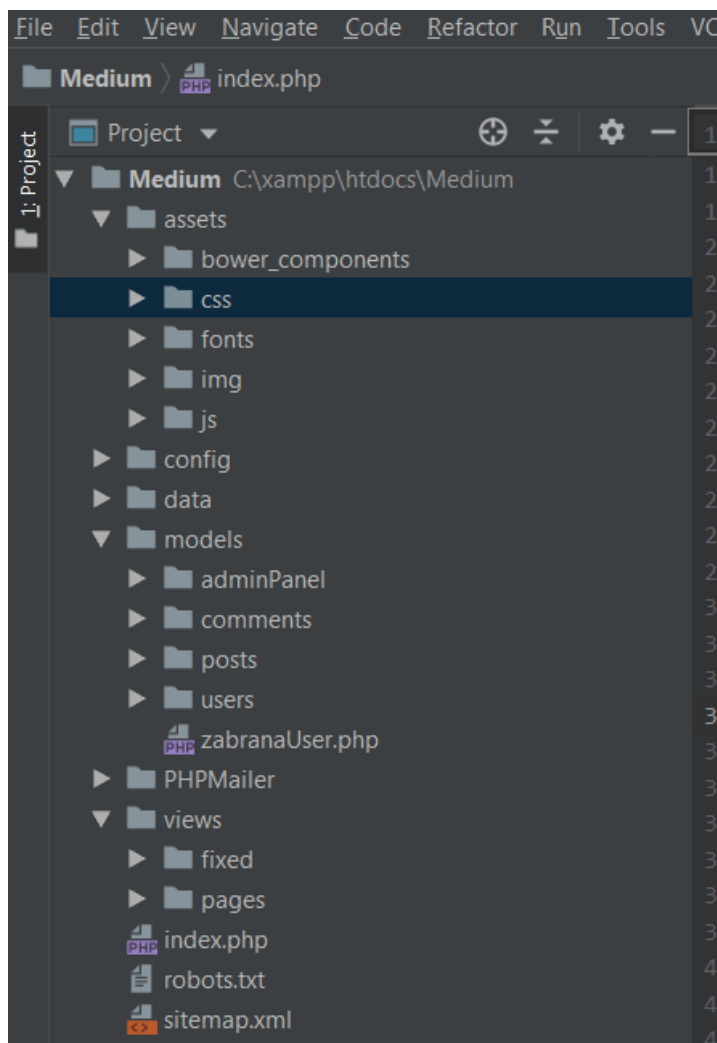


## 2.4. Editor Slike



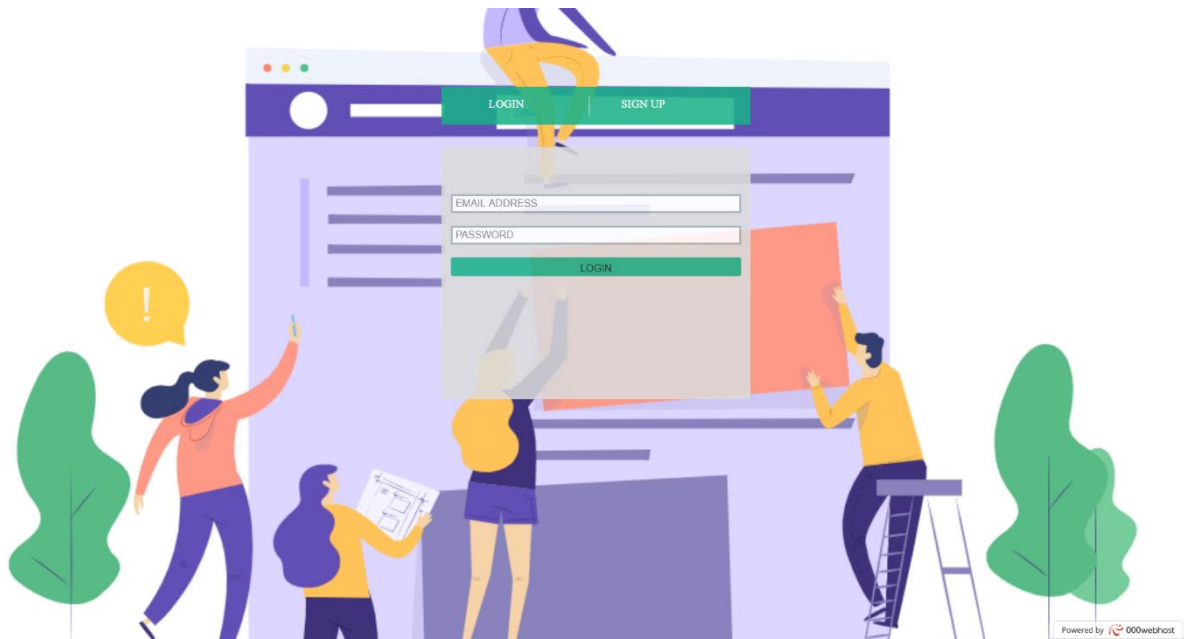






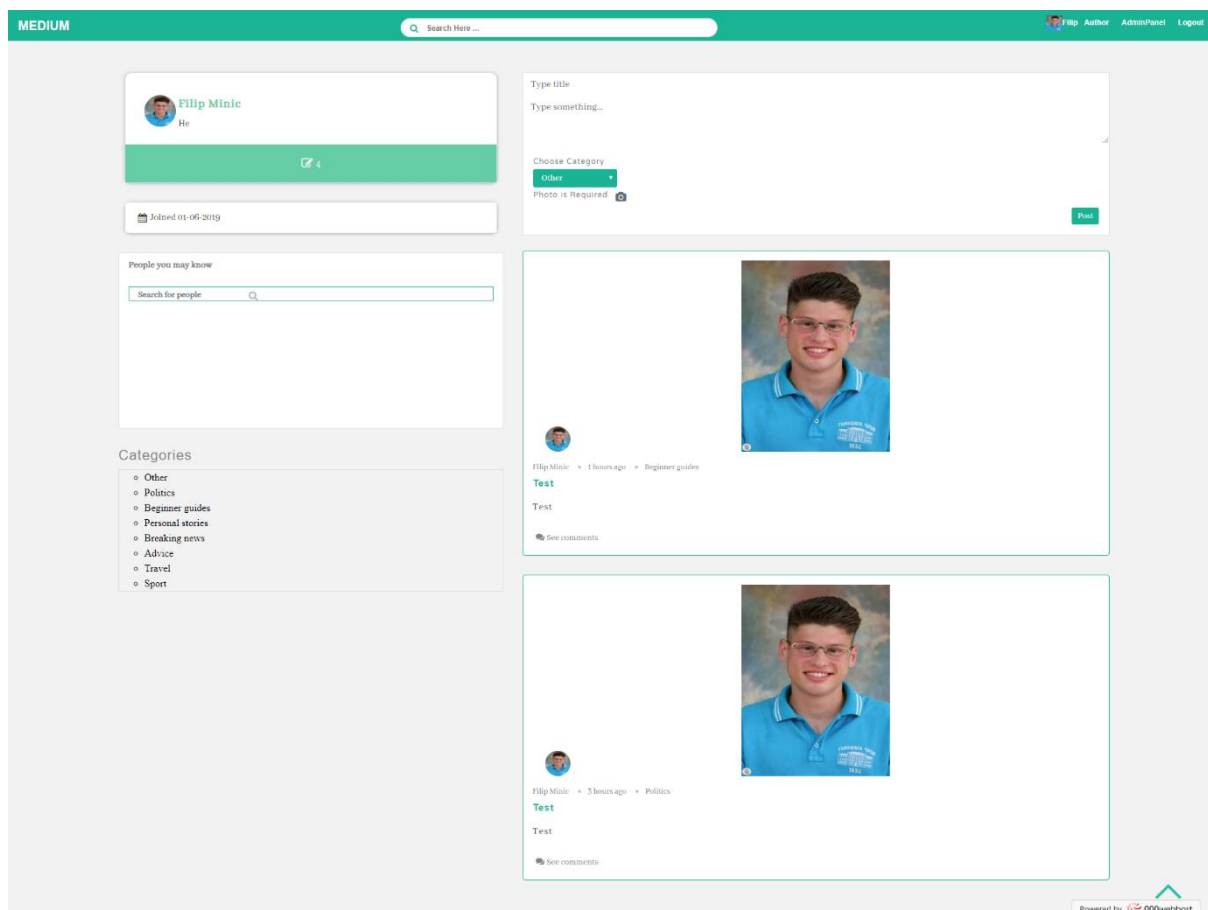
## 2.5. Opis stranica sajta

Index.php



Pocetna stranica sajta sa koje se korisnik loguje ili kreira nalog za pristup sadrzaju sajta.

Page=pocetna



Strana sajta na kojoj su izistani svi postovi koji su na sajtu kreirani. Nakon spustanja na kraj strane ucitavaju se nova 3 posta i tako dok se ne dodje do poslednjg. Search u navigaciji služi za pretragu postova na osnovu imena korisnika, ili ključnih reci iz naslova ili opisa posta. Sa leve strane se nalaze podaci o ulogovanom korisniku i koliko je ukupno on postova kreirao. Takođe se nalazi pretraga korisnika sajta. Ispod toga se nalazi filtriranje na osnovu kategorija.



Page=profilUser


MEDIUM

Filip

Author

AdminPanel

Logout



Filip

Filip Minic

User Role: ADMIN

Edit Profile

Edit Posts

4

Total Posts

About

He

Full Name : Filip Minic

Filip Minic

1 hours ago

Beginner guides

Test

Test

See comments

Filip Minic

3 hours ago

Politics

Test

Test

See comments

Filip Minic

1 days ago

Personal stories

ffa

sfa:sa

See comments

END

Powered by 000webhost

17

Svaki korisnik ima svoj profil na kome se nalaze samo postovi koje je on okacio. Takodje korisnik moze menjati svoju porfilnu i naslovnu sliku, kao i osnovne podatke o sebi. Moze i brisati svoje postove ili ih editovati.

## Page= editProfile

MEDIUM

Filip Author AdminPanel Logout

Change Personal Info

First Name

Filip

Last Name

Minic

Mail

filip.minic98@gmail.com

Description

He

Save

Change Password

Password

Confirm Password

Save

Change User Role

admin

Save

Change Background Photo

Change Background Photo

Choose profile photo

Save

Change Profile Photo

Change Profile Photo

Choose profile photo

Save


DELETE USER

Powered by 000webhost

Strana na kojoj korisnik menja podatke o sebi i samo on ima pristup ovim podacima.

## Page=editPost

MEDIUM

 [Filep](#) [Author](#) [AdminPanel](#) [Logout](#)

Edit


Title

Description




Choose Category


Other

Photo is required



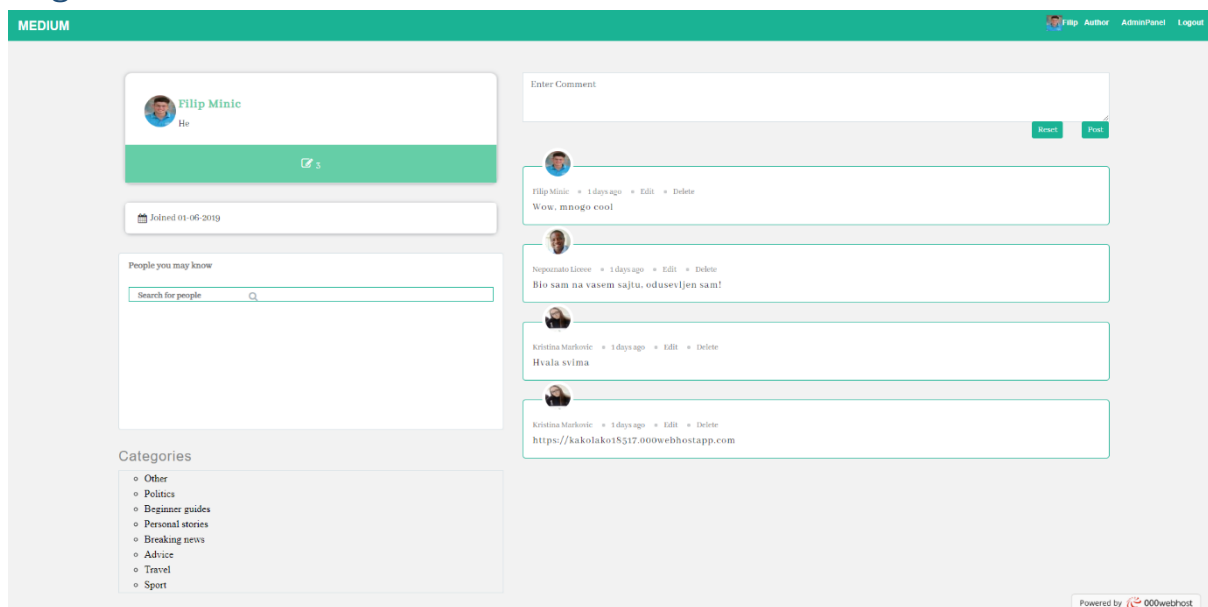
Save

Photo	Title	Description	Category		
	Kica	das da das	Beginner guides	Update	Delete
	Test	Test	Politics	Update	Delete
	Test	Test	Beginner guides	Update	Delete

Powered by  000webhost

Strana na kojoj korisnik edituje ili brise svoje postove, samo on ima pristup ovoj strani.

## Page=comment



Strana na kojoj se nalaze komentari za post na kom smo kliknuli na „See comments“. Korisnik moze editovati ili brisati svoje komentare.

## Page=adminPanel

**MEDIUM**

Filip Author AdminPanel Logout

Usefull links

[Export Excel](#) [Export Word](#) [Documentation](#)

Attendance in the last 24 hours in %

Login/Reg	Index	User Profile	Edit Profile	Edit Post	Comment	Admin Panel	Author
20.73	31.71	15.85	3.66	2.44	2.44	17.07	6.1

Active members: 1

No.	First Name	Last Name	Role
1	Filip	Minic	admin

Powered by  000webhost

Adminpanel je strana na kojoj se nalazi Export u excel korisnika sajta i export podataka o autoru u word. Statistika pristupa stranicama u poslednja 24 casa izrazena u %. Kao i Broj trenutno ulogovanih korisnika, imena i uloga korisnika.

Admin ima pristup svim stranicama sajta i manipulacijom sadrzaja na sajtu.

Moze editovati, brisati korisnike, kao i svacije postove i komentare.

## Page=autor


**MEDIUM**

Filip Author AdminPanel Logout

Author's details

First Name	Last Name	Birthday	Faculty	ID	Born
Filip	Minic	1998-04-04	ICT College	138/17	Cacak, Serbia

Author's Photo



Powered by 000webhost

Slika autora sajta i podaci iz baze o njemu.

### 3. Kodovi

#### Index.php

```
<?php
    include "config/connection.php";
    include "models/users/functions.php";
    include "models/posts/functions.php";
    include "models/comments/functions.php";
    include "models/adminPanel/functions.php";
    require_once "views/fixed/head.php";

// moras staviti !isset kada budeš završio sajt!!!
if(!isset($_SESSION['user'])){
    include "views/pages/loginRegister.php";
}else{
    include "views/fixed/nav.php";
    if(isset($_GET['page'])){
        switch($_GET['page'])
        {
            case 'pocetna':
                include "views/pages/profile_info.php";
                include "views/pages/postovi.php";
                break;
            case 'profilUser':
                include "views/pages/profile_user_info.php";
                include "views/pages/profile_details.php";
                include "views/pages/postovi2.php";
                break;
            case 'editProfile':
                include "views/pages/editProfile.php";
                break;
            case 'editPost':
                include "views/pages/editPost.php";
                break;
            case 'comment':
                include "views/pages/profile_info.php";
                include "views/pages/comments.php";
                break;
            case 'adminPanel':
                include "views/pages/admin.php";
                break;
            case 'autor':
                include "views/pages/autor.php";
                break;
            default:
                include "views/pages/profile_info.php";
                include "views/pages/postovi.php";
                break;
        }
    } else{
        include "views/pages/profile_info.php";
        include "views/pages/postovi.php";
    }
}
```



```

    }
    include "views/fixed/footer.php";

?>

```

### 3.1. config

.env

```

DBNAME=sajt_php_praktikum
SERVER=localhost
USERNAME=root
PASSWORD=

```

Config.php

```

<?php

// Osnovna podesavanja
define("ABSOLUTE_PATH", $_SERVER["DOCUMENT_ROOT"]."/Medium");
//define("RELATIVE_PATH", $_SERVER["DOCUMENT_ROOT"]."/labphp/termin7");

// Ostala podesavanja
define("ENV_FAJL", ABSOLUTE_PATH."/config/.env");
define("LOG_FAJL", ABSOLUTE_PATH."/data/log.txt");
define("ERROR_FILE", ABSOLUTE_PATH."/data/errors.txt");
define("LOGGING_FAJL", ABSOLUTE_PATH."/data/logging.txt");

// Podesavanja za bazu
define("SERVER", env("SERVER"));
define("DATABASE", env("DBNAME"));
define("USERNAME", env("USERNAME"));
define("PASSWORD", env("PASSWORD"));

function env($naziv){
    $podaci = file(ENV_FAJL);
    $vrednost = "";
    foreach($podaci as $key=>$value){
        $konfig = explode("=", $value);
        if($konfig[0]==$naziv){
            $vrednost = trim($konfig[1]); // trim() zbog \n
        }
    }
    return $vrednost;
}

```

Connection.php

```

<?php
@session_start();
require_once "config.php";

```

```

try {
    $conn = new PDO("mysql:host=".SERVER.";dbname=".DATABASE.";charset=utf8",
    USERNAME, PASSWORD);
    $conn->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
}
catch(PDOException $ex){
    echo $ex->getMessage();
}

function executeQuery($query){
    global $conn;
    return $conn->query($query)->fetchAll();
}

function zabeleziPristupStranici(){
    $open = fopen(LOG_FAJL, "a");
    if($open){
        $date = date('d-m-Y H:i:s');
        @fwrite($open,
        "{$_SERVER['REQUEST_URI']}\t{$date}\t{$_SERVER['REMOTE_ADDR']}\t
        {$_SESSION['user']->first_name}-{$_SESSION['user']->last_name}\n");
        fclose($open);
    }
}

function catchErrors($error){
    @$open = fopen(ERROR_FILE, "a");
    $unos = $error."\t".date('d-m-Y H:i:s'). "\n";
    @fwrite($open, $unos);
    @fclose($open);
}

```

## 3.2. Models

### 3.2.1.adminPanel

#### Functions.php

```
<?php
function pristumStranamaUprocentima(){
    $array = [];
    $ukupno = 0;
    $loginRegister = 0;
    $pocetna = 0;
    $profil = 0;
    $editProfil = 0;
    $editPost = 0;
    $comment = 0;
    $admin = 0;
    $autor = 0;

    $poslednja24h = strtotime("1 day ago");
    //var_dump($poslednja24h);
    @$file = file(LOG_FAJL);
    if(count($file)){
        foreach ($file as $f){
            //var_dump($f);
            $delovi = explode("\t", $f);
            $url = explode(".php", $delovi[0]);
            $page = explode("&", $url[1]);
            //var_dump($delovi);
            $time = trim(strtotime($delovi[1]));
            //var_dump($time);
            //print_r($delovi[3]);
            if($time >= $poslednja24h){
                switch ($page[0]){
                    case "":
                        $loginRegister++;$ukupno++;
                        break;
                    case "?page=pocetna":
                        $pocetna++;$ukupno++;
                        break;
                    case "?page=profilUser":
                        $profil++;$ukupno++;
                        break;
                    case "?page=editProfile":
                        $editProfil++;$ukupno++;
                        break;
                    case "?page=editPost":
                        $editPost++;$ukupno++;
                        break;
                    case "?page=comment":
                        $comment++;$ukupno++;
                        break;
                    case "?page=adminPanel":
                        $admin++;$ukupno++;
                        break;
                    case "?page=autor":
                        $autor++;$ukupno++;
                }
            }
        }
    }
}
```

```

                break;
            default:
                $pocetna++;$ukupno++;
                break;
        }
    }
}

if($ukupno >0){
    $array[] = round($loginRegister*100/$ukupno, 2);
    $array[] = round($pocetna*100/$ukupno, 2);
    $array[] = round($profil*100/$ukupno, 2);
    $array[] = round($editProfil*100/$ukupno, 2);
    $array[] = round($editPost*100/$ukupno, 2);
    $array[] = round($comment*100/$ukupno, 2);
    $array[] = round($admin*100/$ukupno, 2);
    $array[] = round($autor*100/$ukupno, 2);
}
}

return $array;
}

function loggingCount($id){
    @$open = fopen(LOGGING_FAJL, "a");
    $unos = $id. "\n";
    @fwrite($open, $unos);
    @fclose($open);
}

function loggedCount(){
    return count(file(LOGGING_FAJL));
}

function deleteLoggingCount($id){
    $id = (int)$id;
    $unos = "";

    @$file = file(LOGGING_FAJL);

    if(count($file)){
        foreach ($file as $f){
            $fId = trim((int)$f);
            if($fId != $id){
                $unos .= $fId. "\n";
            }
        }
    }

    @$open = fopen(LOGGING_FAJL, "w");
    @fwrite($open, $unos);
    @fclose($open);
}

function nameUser($id){

```

```

    global $conn;
    $upit = "SELECT u.first_name, u.last_name, r.role_name FROM users u JOIN
roles r ON u.id_role = r.id_role WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$id]);

    return $priprema->fetch();
}

function autor(){
    global $conn;
    $upit = "SELECT * FROM autor WHERE id_autor = 1";
    $rezultat = $conn->query($upit);

    return $rezultat->fetch();
}

```

### 3.2.2. comments

#### deleteComm.php

```
<?php
session_start();
if(isset($_POST['idComm'])){
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";

    $idComm = $_POST['idComm'];
    try{
        $rezultat = deleteComm($idComm);
        http_response_code(204);
        echo json_encode($rezultat);
    } catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(["Greska", $ex->getMessage()]);
        catchErrors("deleteComm.php ->".$ex->getMessage());
    }
}

}else{
    http_response_code(400);
}
```

#### Functions.php

```
<?php
function getAllComments($idPost){
    global $conn;
    $upit = "SELECT * FROM comments c JOIN users u ON c.id_user = u.id_user
JOIN user_photo up ON u.id_user = up.id_user WHERE c.id_post = ? ORDER BY
c.comment_created_at DESC";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idPost]);
    $rezultat = $priprema->fetchAll();

    return $rezultat;
}

function insertComment($comment, $vremeComm, $idUser, $idPost){
    global $conn;
    $upit = "INSERT INTO comments (id_comment, text, comment_created_at,
id_user, id_post) VALUES(NULL,?, ?, ?, ?)";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$comment, $vremeComm, $idUser, $idPost]);

    return $priprema;
}

function deleteComm($idComm){
    global $conn;
    $upit = "DELETE FROM comments WHERE id_comment = ?";
    $priprema = $conn->prepare($upit);
```

```

    $priprema-> execute([$idComm]);

    return $priprema;
}

function getOneComm($idComm){
    global $conn;
    $upit = "SELECT * FROM comments WHERE id_comment = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idComm]);

    return $priprema->fetch();
}

function updateComm($comment, $vremeComm, $idComm, $idPost){
    global $conn;
    $upit = "UPDATE comments SET text = ?, comment_updated_at = ? WHERE
id_comment = ? AND id_post = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$comment, $vremeComm, $idComm, $idPost]);

    return $priprema;
}

```

## getComments.php

```

<?php
header("Content-type: application/json");
if(isset($_POST['idPost'])){
    require "../config/connection.php";
    require "functions.php";

    $idComment = $_POST['idPost'];

    try{
        $select = getAllComments($idComment);

        http_response_code(200);
        echo json_encode($select);
    } catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(['greska'], $ex->getMessage());
        catchErrors("getComments.php ->".$ex->getMessage());
    }
}else{
    http_response_code(400);
}

```

## getOneComment.php

```
<?php
header("Content-type: application/json");

if(isset($_POST['idComm'])){
    require "../config/connection.php";
    require "functions.php";

    $idComm = $_POST['idComm'];
    try{
        $resenje = getOneComm($idComm);
        http_response_code(200);
        echo json_encode($resenje);
    } catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(['greska', 'Problem'. $ex->getMessage()]);
        catchErrors("getOneComment.php ->". $ex->getMessage());
    }
}
} else{
    http_response_code(400);
}
```

## insertComment.php

```
<?php
session_start();
header("Content-type: application/json");

if(isset($_POST['btnComment'])){
    require "../config/connection.php";
    require "functions.php";

    $idPost = $_POST['idPost'];
    $comment = $_POST['textOpis'];
    $idUser = $_SESSION['id_user'];
    $vremeComm = date("Y-m-d H-i-s", time());

    $reIdPost = "/^\d*$/";
    $reComment = "/^(.|\s)*[a-zA-Z]+(.\|\s)*$/";

    $greske = [];

    if(!preg_match($reIdPost, $idPost)){
        $greske[] = "Id mora biti unet";
    }
    if(!preg_match($reComment, $comment)){
        $greske[] = "Text komentara";
    }

    if(count($greske) > 0){
        http_response_code(422);
        echo json_encode($greske);
    } else{
```



```
try{
    $insert = insertComment($comment, $vremeComm, $idUser, $idPost);
    http_response_code(204);
    echo json_encode($insert);
}catch (PDOException $ex){
    http_response_code(500);
    echo json_encode(['Greska', $ex->getMessage()]);
    catchErrors("insertComment.php ->".$ex->getMessage());
}
}
}else{
    http_response_code(400);
}
```

updateComment.php

```
<?php
session_start();
header("Content-type: application/json");

if(isset($_POST['btnComment'])){
    require "../config/connection.php";
    require "functions.php";

    $idComm = $_POST['idComm'];
    $idPost = $_POST['idPost'];
    $comment = $_POST['textOpis'];
    $vremeComm = date("Y-m-d H-i-s", time());

    $reId = "/^\\d*$/";
    $reComment = "/^(.|\\s)*[a-zA-Z]+(.|\\s)*$/";

    $greske = [];

    if(!preg_match($reId, $idPost)){
        $greske[] = "Id mora biti unet";
    }
    if(!preg_match($reId, $idComm)){
        $greske[] = "Id mora biti unet";
    }
    if(!preg_match($reComment, $comment)){
        $greske[] = "Text komentara";
    }

    if(count($greske) > 0){
        http_response_code(422);
        echo json_encode($greske);
    }else{
        try{
            $insert = updateComm($comment, $vremeComm, $idComm, $idPost);
            http_response_code(204);
            echo json_encode($insert);
        }catch (PDOException $ex){
            http_response_code(500);
            echo json_encode(['Greska', $ex->getMessage()]);
        }
    }
}
```

```
        catchErrors("updateComment.php ->".$ex->getMessage());
    }
}
}else{
    http_response_code(400);
}
```

### 3.2.3. posts

#### deltePost.php

```
<?php
if(isset($_POST['idPost'])){
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";
    $idPost = $_POST['idPost'];
    try{
        $rezultat = deletePost($idPost);
        http_response_code(204);
        echo json_encode($rezultat);
    } catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(["Greska", $ex->getMessage()]);
        catchErrors("deletePost.php ->".$ex->getMessage());
    }
}
}else{
    http_response_code(400);
}
```

#### Functions.php

```
<?php

function getCategories(){
    return executeQuery("SELECT * FROM categories");
}

function insertNewPost($naslov, $opis, $putanja, $novaPutanja, $putanjaMala,
$datumKreiranja, $idCategory, $idUser){
    global $conn;
    $upit = "INSERT INTO posts (id_post, title, post_description,
real_photo_path, photo_path, photo_mala_path, post_created_at, id_category,
id_user) VALUES (NULL, ?, ?, ?, ?, ?, ?, ?, ?)";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$naslov, $opis, $putanja, $novaPutanja, $putanjaMala,
$datumKreiranja, $idCategory, $idUser]);

    return $priprema;
}

function getAllPosts($br){
    return executeQuery("SELECT * FROM user_photo up JOIN users u ON
up.id_user = u.id_user JOIN posts p ON u.id_user = p.id_user JOIN categories c
ON p.id_category = c.id_category ORDER BY p.post_created_at DESC LIMIT $br");
}

function getAllPostsCategory($category, $br){
    return executeQuery("SELECT * FROM user_photo up JOIN users u ON
up.id_user = u.id_user JOIN posts p ON u.id_user = p.id_user JOIN categories c
ON p.id_category = c.id_category WHERE c.id_category = $category ORDER BY
p.post_created_at DESC LIMIT $br");
}
```

```

//function getAllPostsUser($idUser){
//    global $conn;
//    $upit = "SELECT * FROM posts p JOIN users u ON p.id_user = u.id_user
JOIN user_photo up ON u.id_user = up.id_user WHERE p.id_user = ? ORDER BY
p.post_created_at DESC";
//    $priprema = $conn->prepare($upit);
//    $priprema->execute([$idUser]);
//    $rezultat = $priprema->fetchAll();
//
//    return $rezultat;
//}

function getAllPostsUser($idUser, $br2){
    return executeQuery("SELECT * FROM user_photo up JOIN users u ON
up.id_user = u.id_user JOIN posts p ON u.id_user = p.id_user JOIN categories c
ON p.id_category = c.id_category WHERE p.id_user = $idUser ORDER BY
p.post_created_at DESC LIMIT $br2");
}

function searchPosts($title){
    global $conn;
    $upit = "SELECT * FROM posts p JOIN users u ON p.id_user = u.id_user JOIN
user_photo up ON u.id_user = up.id_user WHERE LOWER(p.title) LIKE ? OR
LOWER(p.post_description) LIKE ? OR LOWER(u.first_name) LIKE ? ORDER BY
p.post_created_at DESC";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$title, $title, $title]);
    $rezultat = $priprema->fetchAll();

    return $rezultat;
}

function getAllPostsProfile($idUser){
    global $conn;
    $upit = "SELECT * FROM posts p JOIN categories c ON p.id_category =
c.id_category WHERE p.id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idUser]);
    $rezultat = $priprema->fetchAll();

    return $rezultat;
}

function getOnePost($idPost){
    global $conn;
    $upit = "SELECT * FROM posts WHERE id_post = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idPost]);
    $rezultat = $priprema->fetch();

    return $rezultat;
}

function updatePost($title, $desc, $realPhotoPath, $photoPath, $malaPath,

```

```

$update($idCat, $idUser){
    global $conn;
    $upit = "UPDATE posts SET title = ?, post_description = ?, real_photo_path
= ?, photo_path = ?, photo_mala_path = ?, updated_at = ?, id_category = ?
WHERE id_post = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$title, $desc, $realPhotoPath, $photoPath, $malaPath,
$update, $idCat, $idUser]);

    return $priprema;
}

function deletePost($idPost){
    global $conn;
    $upit = "DELETE FROM posts WHERE id_post = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idPost]);

    return $priprema;
}

```

## getAllCategories.php

```

<?php
header("Content-type: application/json");
require "../config/connection.php";
require "functions.php";

try {
    $categories = getCategories();
    http_response_code(200);
    echo json_encode($categories);
} catch (PDOException $ex){
    http_response_code(500);
    catchErrors("getAllCategories.php ->".$ex->getMessage());
}

```

## getAllPosts.php

```

<?php
@session_start();
header("Content-type: application/json");
require "../config/connection.php";
require "functions.php";
try {
    $br = $_POST['br'];
    if (isset($_SESSION['id_category'])) {
        $category = $_SESSION['id_category'];
        $posts = getAllPostsCategory($category, $br);
    } else {
        $posts = getAllPosts($br);
    }
    http_response_code(200);
    echo json_encode($posts);
}

```

```

}catch (PDOException $ex){
    http_response_code(500);
    catchErrors("getAllPosts.php ->".$ex->getMessage());
}

```

## getAllPostsProfile.php

```

<?php
session_start();
header("Content-type: application/json");
require "../config/connection.php";
require "functions.php";
try {
    $idUser = $_SESSION['userStrana'];
    $postovi = getAllPostsProfile($idUser);

    echo json_encode($postovi);
}catch (PDOException $ex){
    http_response_code(500);
    catchErrors("getAllPostsProfile.php ->".$ex->getMessage());
}

```

## getAllPostsUser.php

```

<?php
session_start();

header("Content-type: application/json");
require "../config/connection.php";
require "functions.php";
try {
    $idUserSession = $_SESSION['userStrana'];
    $br2 = $_POST['brojac'];
    $posts = getAllPostsUser($idUserSession, $br2);
    unset($idUserSession);
    http_response_code(200);
    echo json_encode($posts);
} catch (PDOException $ex){
    http_response_code(500);
    catchErrors("getAllPostsUser.php ->".$ex->getMessage());
}

```

## getPostOne.php

```

<?php
header("Content-type: application/json");

if(isset($_POST['idPost'])){
    require "../config/connection.php";
    require "functions.php";

    $idPost = $_POST['idPost'];
}

```

```

try{
    $resenje = getOnePost($idPost);
    http_response_code(200);
    echo json_encode($resenje);
} catch (PDOException $ex){
    http_response_code(500);
    echo json_encode(['greska', 'Problem'. $ex->getMessage()]);
    catchErrors("getOnePost.php ->". $ex->getMessage());
}

}

}else{
    http_response_code(400);
}

```

## insertPost.php

```

<?php
session_start();
if(isset($_POST['btnPost'])) {
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";

    $fajl_naziv = $_FILES['postPhoto']['name'];
    $fajl_tmpLokacija = $_FILES['postPhoto']['tmp_name'];
    $fajl_tip = $_FILES['postPhoto']['type'];
    $fajl_velicina = $_FILES['postPhoto']['size'];
    $naslov = $_POST['tbTitle'];
    $opis = $_POST['textOpis'];
    $category = $_POST['ddlCategories'];

    $greske = [];

    $dozvoljeni_tipovi = ['image/jpg', 'image/jpeg', 'image/png'];

    if(!in_array($fajl_tip, $dozvoljeni_tipovi)){
        array_push($greske, "Pogresan tip fajla. - Profil slika");
    }
    if($fajl_velicina > 3000000){
        array_push($greske, "Maksimalna velicina fajla je 3MB. - Profil slika");
    }
    if(strlen($opis) == 0){
        array_push($greske, "Mora se neki opis napisati");
    }
    if(strlen($naslov) == 0){
        array_push($greske, "Mora se neki opis napisati");
    }

    if(count($greske) == 0){
        list($sirina, $visina) = getimagesize($fajl_tmpLokacija);

        $postojecaSlika = null;
        switch($fajl_tip){
            case 'image/jpeg':

```

```

        $postojecaSlika = imagecreatefromjpeg($fajl_tmpLokacija);
        break;
    case 'image/png':
        $postojecaSlika = imagecreatefrompng($fajl_tmpLokacija);
        break;
    }

    $novaVisina = 303;
    $novaSirina = ($novaVisina/$visina) * $sirina;

    $novaMalaVisina = 50;
    $novaMalaSirina = 50;

    $novaSlika = imagecreatetruecolor($novaSirina, $novaVisina);
    $novaSlikaMala = imagecreatetruecolor($novaMalaSirina,
    $novaMalaVisina);

    imagecopyresampled($novaSlika, $postojecaSlika, 0, 0, 0, 0,
    $novaSirina, $novaVisina, $sirina, $visina);
    imagecopyresampled($novaSlikaMala, $postojecaSlika, 0, 0, 0, 0,
    $novaMalaSirina, $novaMalaVisina, $sirina, $visina);

    $naziv = time().$fajl_naziv;
    $putanjaNovaSlika = 'assets/img/posts/nova_'. $naziv;
    $putanjaNovaSlikaMala = 'assets/img/posts/novaMala_'. $naziv;

    switch($fajl_tip){
        case 'image/jpeg':
            imagejpeg($novaSlika, '../../'.$putanjaNovaSlika, 75);
            imagejpeg($novaSlikaMala, '../../'.$putanjaNovaSlikaMala, 75);
            break;
        case 'image/png':
            imagepng($novaSlika, '../../'.$putanjaNovaSlika);
            imagepng($novaSlikaMala, '../../'.$putanjaNovaSlikaMala);
            break;
    }

    $putanjaOriginalnaSlika = 'assets/img/posts/'. $naziv;

    if(move_uploaded_file($fajl_tmpLokacija,
    '../../'.$putanjaOriginalnaSlika)){

        $idUser = $_SESSION['id_user'];
        $vremeUnosa = date("Y-m-d H-i-s", time());
        try {
            $isInserted = insertNewPost($naslov, $opis,
            $putanjaOriginalnaSlika, $putanjaNovaSlika, $putanjaNovaSlikaMala,
            $vremeUnosa, $category, $idUser);

            header("Location: ../../index.php?page=pocetna");

        } catch(PDOException $ex){
            header("Location: ../../index.php?page=pocetna");
            echo $ex->getMessage();
            catchErrors("isertPost.php ->". $ex->getMessage());
        }
    }

```



```

    }
}

imagedestroy($postojecaSlika);
imagedestroy($novaSlika);
imagedestroy($novaSlikaMala);

} else {
    header("Location: ../../index.php?page=pocetna");
}
}

```

## searchPost.php

```

<?php
if(isset($_POST['post'])){
    header("Content-type: application/json");
    require "../../config/connection.php";
    require "functions.php";

    $post = "%".strtolower($_POST['post'])."%";
    try{
        $posts = searchPosts($post);
        http_response_code(200);
        echo json_encode($posts);
    }catch (PDOException $ex){
        http_response_code(500);
        catchErrors("searchPost.php ->".$ex->getMessage());
    }

    } else{
        http_response_code(400);
    }
}

```

## updatePost.php

```

<?php

if(isset($_POST['btnUpdatePost'])) {
    header("Content-type: application/json");
    require "../../config/connection.php";
    require "functions.php";

    $fajl_naziv = $_FILES['postPhoto']['name'];
    $fajl_tmpLokacija = $_FILES['postPhoto']['tmp_name'];
    $fajl_tip = $_FILES['postPhoto']['type'];
    $fajl_velicina = $_FILES['postPhoto']['size'];

    $naslov = $_POST['tbTitle'];
    $opis = $_POST['tbDescription'];
    $category = $_POST['ddlCategories'];
    $idPost = $_POST['hiddenPost'];
}

```

```

$greske = [];

$dozvoljeni_tipovi = ['image/jpg', 'image/jpeg', 'image/png'];

if(!in_array($fajl_tip, $dozvoljeni_tipovi)){
    array_push($greske, "Pogresan tip fajla. - Profil slika");
}
if($fajl_velicina > 3000000){
    array_push($greske, "Maksimalna velicina fajla je 3MB. - Profil slika");
}
if(strlen($opis) == 0){
    array_push($greske, "Mora se neki opis napisati");
}
if(strlen($naslov) == 0){
    array_push($greske, "Mora se neki opis napisati");
}

if(count($greske) == 0){
    list($sirina, $visina) = getimagesize($fajl_tmpLokacija);

    $postojecaSlika = null;
    switch($fajl_tip){
        case 'image/jpeg':
            $postojecaSlika = imagecreatefromjpeg($fajl_tmpLokacija);
            break;
        case 'image/png':
            $postojecaSlika = imagecreatefrompng($fajl_tmpLokacija);
            break;
    }

    $novaVisina = 303;
    $novaSirina = ($novaVisina/$visina) * $sirina;

    $novaMalaVisina = 50;
    $novaMalaSirina = 50;

    $novaSlika = imagecreatetruecolor($novaSirina, $novaVisina);
    $novaSlikaMala = imagecreatetruecolor($novaMalaSirina, $novaMalaVisina);

    imagecopyresampled($novaSlika, $postojecaSlika, 0, 0, 0, 0, $novaSirina, $novaVisina, $sirina, $visina);
    imagecopyresampled($novaSlikaMala, $postojecaSlika, 0, 0, 0, 0, $novaMalaSirina, $novaMalaVisina, $sirina, $visina);

    $naziv = time().$fajl_naziv;
    $putanjaNovaSlika = 'assets/img/posts/nova_'. $naziv;
    $putanjaNovaSlikaMala = 'assets/img/posts/novaMala_'. $naziv;

    switch($fajl_tip){
        case 'image/jpeg':
            imagejpeg($novaSlika, '../..' . $putanjaNovaSlika, 75);

```

```

        imagejpeg($novaSlikaMala, '../..' . $putanjaNovaSlikaMala, 75);
        break;
    case 'image/png':
        imagepng($novaSlika, '../..' . $putanjaNovaSlika);
        imagepng($novaSlikaMala, '../..' . $putanjaNovaSlikaMala);
        break;
    }

    $putanjaOriginalnaSlika = 'assets/img/posts/' . $naziv;

    if(move_uploaded_file($fajl_tmpLokacija,
        '../..' . $putanjaOriginalnaSlika)){

        $vremeUnosa = date("Y-m-d H-i-s", time());
        try {
            $isInserted = updatePost($naslov, $opis,
                $putanjaOriginalnaSlika, $putanjaNovaSlika, $putanjaNovaSlikaMala,
                $vremeUnosa, $category, $idPost);

            header("Location: ../../index.php?page=pocetna");

        } catch(PDOException $ex){
            header("Location: ../../index.php?page=pocetna");
            echo $ex->getMessage();
            catchErrors("updatePost.php ->" . $ex->getMessage());
        }
    }

    imagedestroy($postojecaSlika);
    imagedestroy($novaSlika);
    imagedestroy($novaSlikaMala);

} else {
    header("Location: ../../index.php?page=pocetna");
}
}

```

### 3.2.4 users

#### deleteUser.php

```
<?php
session_start();
if(isset($_POST['deleteUser'])){
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";

    $idUserDelete = $_SESSION['idUserDelete'];
    try{
        $delete = deleteUser($idUserDelete);
        http_response_code(204);
        header("Location:../../index.php?page=pocetna");
    }catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(['Nije obrisan user', $ex->getMessage()]);
        catchErrors("deleteUser.php ->".$ex->getMessage());
        header("Location:../../index.php?page=pocetna");
    }
} else{
    http_response_code(400);
    header("Location:../../index.php?page=pocetna");
}
unset($idUserDelete);
```

#### exportExcel.php

```
<?php

require "../config/connection.php";
require "functions.php";

$getUsers = getAllUsers();

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

$excel->Visible = 1;

$excel->DisplayAlerts = 1;

$workbook = $excel->Workbooks->Add();

$sheet = $workbook->Worksheets('Sheet1');
$sheet->activate;

$br = 1;
foreach ($getUsers as $user){
    $polje = $sheet->Range("A{$br}");
    $polje->activate;
    $polje->value = $user->id_user;
```

```

        $polje = $sheet->Range("B{$br}");
        $polje->activate;
        $polje->value = $user->first_name;

        $polje = $sheet->Range("C{$br}");
        $polje->activate;
        $polje->value = $user->last_name;

        $polje = $sheet->Range("D{$br}");
        $polje->activate;
        $polje->value = $user->email;

        $polje = $sheet->Range("E{$br}");
        $polje->activate;
        $polje->value = $user->description;

        $polje = $sheet->Range("F{$br}");
        $polje->activate;
        $polje->value = $user->created_at;

        $polje = $sheet->Range("G{$br}");
        $polje->activate;
        $polje->value = $user->role_name;
        $br++;
    }

    $workbook->_SaveAs("http://localhost/Medium/models/users/UserInfo.xlsx", -
4143);
    $workbook->Save();

    $workbook->Saved=true;
    $workbook->Close;

    $excel->Workbooks->Close();
    $excel->Quit();

    unset($sheet);
    unset($workbook);
    unset($excel);

    header("Location: ../../index.php?page=pocetna");

```

## exportWord.php

```

<?php

include '../../config/connection.php';
include "../adminPanel/functions.php";

$autor = autor();
//var_dump($autor);

$word = new COM("Word.Application");
$word->Visible = true;

$word->Documents->Add();

```

```

$word->Selection->TypeText("First Name: $autor->ime \n Last Name: $autor->prezime \n Birthday: $autor->datum \n Faculty: $autor->skola \n ID: $autor->index \n Born: $autor->mesto");

$word->Documents[1]->SaveAs("Author.doc");

header("Location: ../../index.php?page=autor");

```

## Functions.php

```

<?php

function userRregister($ime, $prezime, $mail, $password, $created_at){
    global $conn;
    $upitRegister = "INSERT INTO users (id_user, first_name, last_name, email, password, created_at, id_role) values (NULL, ?, ?, ?, ?, ?, 2)";
    $priprema = $conn->prepare($upitRegister);
    $priprema->execute([$ime, $prezime, $mail, $password, $created_at]);

    $lastInsertedIdUser = $conn->lastInsertId();
    userPhotoAddReister($lastInsertedIdUser);

    return $priprema;
}

function userPhotoAddReister($lastUserId){
    global $conn;
    $upit = "INSERT INTO user_photo VALUES (NULL, 'assets/img/users/1559432200profilna.jpg', 'assets/img/users/novaProfilna_1559432200profilna.jpg', 'assets/img/users/novaKomentar_1559432200profilna.jpg', 'assets/img/users/novaNavigacija_1559432200profilna.jpg', 'assets/img/usersBackground/1559432191backgoiund.jpg', 'assets/img/usersBackground/nova_1559432191backgoiund.jpg', ?)";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$lastUserId]);

    return $priprema;
}

function getMailUser($mail){
    global $conn;
    $upitMail = "SELECT * FROM users WHERE email = ?";
    $priprema = $conn->prepare($upitMail);
    $priprema->execute([$mail]);

    return $priprema;
}

function userLogin($email, $sifra){
    global $conn;
    $upit = "SELECT * FROM users WHERE email = ? AND password = ?";
    $priprema = $conn->prepare($upit);

```

```

    $priprema->execute([$email, $sifra]);

    return $priprema;
}

function userLoginMailer($email, $sifra){
    global $conn;
    $upit = "SELECT * FROM users WHERE email = ? AND password <> ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$email, $sifra]);
    $priprema->fetch();

    return $priprema;
}

function updateBackgroundPhoto($putanjaOriginalna, $putanjaNovaSlika,
$idUser){
    global $conn;
    $upit = "UPDATE user_photo SET real_background_path = ?, background_path =
? WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$putanjaOriginalna, $putanjaNovaSlika, $idUser]);

    return $priprema;
}

function updateProfilePhoto($putanjaOriginalna, $putanjaProfilna,
$putanjaKomentar, $putanjaNavigacija, $idUser){
    global $conn;
    $upit = "UPDATE user_photo SET real_profile_path = ?, profile_path = ?,
comment_path = ?, nav_path = ? WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$putanjaOriginalna, $putanjaProfilna, $putanjaKomentar,
$putanjaNavigacija, $idUser]);

    return $priprema;
}

function getUser ($idUser){
    global $conn;
    $upit = "SELECT * FROM users WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idUser]);

    return $priprema->fetch();
}

function getAllUsers(){
    return executeQuery("SELECT * FROM users u JOIN roles r ON u.id_role =
r.id_role");
}

function getUserPhoto($idUser){
    global $conn;
    $upit = "SELECT * FROM user_photo WHERE id_user = ?";

```

```

    $priprema = $conn->prepare($upit);
    $priprema->execute([$idUser]);

    return $priprema->fetch();
}

function updateUserProfile($ime, $prezime, $email, $description, $idUser){
    global $conn;
    $upit = "UPDATE users SET first_name = ?, last_name = ?, email = ?,
description = ? WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$ime, $prezime, $email, $description, $idUser]);

    return $priprema;
}

function updateUserPassword($password, $idUser){
    global $conn;
    $upit = "UPDATE users SET password = ? WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$password, $idUser]);

    return $priprema;
}

function searchUser($user){
    global $conn;
    $upit = "SELECT * FROM users u JOIN user_photo up ON u.id_user =
up.id_user WHERE LOWER(u.last_name) LIKE ? OR LOWER(u.first_name) LIKE ? ORDER
BY u.created_at ASC LIMIT 7 ";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$user, $user]);
    $rezultat = $priprema->fetchAll();

    return $rezultat;
}

function deleteUser($idUser){
    global $conn;
    $upit = "DELETE FROM users WHERE id_user = ?";
    $priprema = $conn->prepare($upit);
    $priprema->execute([$idUser]);

    return $priprema;
}

function getRoles(){
    return executeQuery("SELECT * FROM roles");
}

function updateRole($role, $idUser){
    global $conn;
    $upit = "UPDATE users SET id_role = ? WHERE id_user = ?";
    $piprema = $conn->prepare($upit);
    $piprema->execute([$role, $idUser]);

```



```

        return $piprema;
    }

    function userRole($idUser){
        global $conn;
        $upit = "SELECT r.role_name FROM roles r JOIN users u ON r.id_role =
u.id_role WHERE id_user = ?";
        $piprema = $conn->prepare($upit);
        $piprema->execute([$idUser]);

        return $piprema->fetch();
    }

```

## Login.php

```

<?php
@session_start();
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

if(isset($_POST['btnLogin'])){
    header("Content-type: application/json");
    ob_start();
    require "../config/connection.php";
    require "../adminPanel/functions.php";
    require "functions.php";
    // Ucitavanje mailera
    require ABSOLUTE_PATH.'/PHPMailer/src/PHPMailer.php';
    require ABSOLUTE_PATH.'/PHPMailer/src/SMTP.php';
    require ABSOLUTE_PATH.'/PHPMailer/src/Exception.php';

    $email = $_POST['loginMail'];
    $sifra = $_POST['loginPassword'];

    $greske = [];
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $greske[] = "Mail is not in good format!";
    }
    if (strlen($sifra) < 6) {
        $greske[] = "Password length must be over 6";
    }

    if(count($greske) > 0){
        http_response_code(422);
        echo json_encode($greske);
        header("Location:../index.php");
    } else{
        $siframd5 = md5($sifra);

        try{

```

```

        $login = userLogin($email, $siframd5);

        if($login->rowCount() == 1){
            $user = $login->fetch();
            http_response_code(200);
            echo json_encode($user);

            $_SESSION['id_user'] = $user->id_user;
            loggingCount($_SESSION['id_user']);
            $_SESSION['user'] = $user;
            $_SESSION['role'] = $user->id_role;
            header("Location:../../index.php?page=pocetna");
        } else{

            $loginMailer = userLoginMailer($email, $siframd5);

            if($loginMailer){
                $mail = new PHPMailer(true);

                $mail->SMTPDebug = 0;

                $mail->isSMTP();
                $mail->Host = 'smtp.gmail.com';

                $mail->SMTPAuth = true;
                $mail->Username = 'auditorne.php@gmail.com';
                $mail->Password = 'Sifra123';
                $mail->SMTPSecure = 'tls';
                $mail->Port = 587;

                $mail->setFrom('mediumteam44@gmail.com', 'Attention');
                $mail->addAddress($email);

                $mail->isHTML(true);

                $mail->Subject= "Attention";

                $mail->Body = "Someone has tried to login in your account,
please change password";

                $mail->send();
                header("Location:../../index.php?page=pocetna");
            }else{
                echo json_encode(["Greska" => "Wrong Mail AND Password"]);
                http_response_code(409);
                header("Location:../../index.php?page=pocetna");
            }
        }
    } catch (PDOException $ex){
        http_response_code(500);
        echo json_encode(['Greska' => "Greska". $ex->getMessage()]);
        catchErrors("login.php ->".$ex->getMessage());
    }
}

```

```

    }
} else{
    http_response_code(403);
    echo json_encode("Sta da radim sad!");
}

```

## Logout.php

```

<?php
session_start();
require "../adminPanel/functions.php";
require "../../config/connection.php";
if(isset($_SESSION['user'])){
    deleteLoggingCount($_SESSION['id_user']);
    unset($_SESSION['user']);
    header("Location: ../../index.php");
}

```

## Registration.php

```

<?php

if(isset($_POST['btnRegister'])) {
    header("Content-type: application/json");
    require "../../config/connection.php";
    require "functions.php";

    $ime = $_POST['first_name'];
    $prezime = $_POST['last_name'];
    $email = $_POST['tbMail'];
    $sifra = $_POST['tbPassword'];

    $reImePrezime = "/^[A-Z][a-z]{2,15}$/";

    $greske = [];

    if (!preg_match($reImePrezime, $ime)) {
        $greske[] = "First Name is not in good format!";
    }
    if (!preg_match($reImePrezime, $prezime)) {
        $greske[] = "Last Name is not in good format!";
    }
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $greske[] = "Mail is not in good format!";
    }
    if (strlen($sifra) < 6) {
        $greske[] = "Password length must be over 6";
    }

    if(count($greske) > 0){
        http_response_code(422);
        echo json_encode($greske);
    }
}

```

```

    } else{

        $proveraMail = getMailUser($email);
        //// provera maila mi nije potrebna zato sto sam to zastitio preko baze
        if($proveraMail->rowCount() > 0){
            http_response_code(409);
            echo json_encode("Email vec postoji");
        } else{
            $siframd5 = md5($sifra);
            $vremeRegistracije = date("Y-m-d H-i-s", time());
            try{
                $register = userRegister($ime, $prezime, $email,
                $siframd5,$vremeRegistracije );

                http_response_code(201);
                echo json_encode($register);
            } catch (PDOException $ex){
                /*http_response_code(409);*/
                http_response_code(500);
                echo json_encode(['Greska' => "BAZA". $ex->getMessage()]);
                catchErrors("registration.php ->".$ex->getMessage());
            }
        }
    }
} else{
    http_response_code(403);
}
}

```

## searchUser.php

```

<?php
if(isset($_POST['user'])){
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";
    try {
        $user = "%" . strtolower($_POST['user']) . "%";

        $users = searchUser($user);
        http_response_code(200);
        echo json_encode($users);
    } catch (PDOException $ex){
        http_response_code(500);
        catchErrors("searchUser.php ->".$ex->getMessage());
    }
} else{
    http_response_code(400);
}
}

```

## updateBackgroundPhoto.php

```
<?php
session_start();
if(isset($_POST['endUpdateBackground']))){

    require_once "../config/connection.php";
    require_once "functions.php";

    $fajl_naziv = $_FILES['backgroundPhoto']['name'];
    $fajl_tmpLokacija = $_FILES['backgroundPhoto']['tmp_name'];
    $fajl_tip = $_FILES['backgroundPhoto']['type'];
    $fajl_velicina = $_FILES['backgroundPhoto']['size'];
    $idUser = $idUser = $_POST['tbHidden3'];

    $greske = [];

    $dozvoljeni_tipovi = ['image/jpg', 'image/jpeg', 'image/png'];

    if(!in_array($fajl_tip, $dozvoljeni_tipovi)){
        array_push($greske, "Pogresan tip fajla. - Profil slika");
    }
    if($fajl_velicina > 3000000){
        array_push($greske, "Maksimalna velicina fajla je 3MB. - Profil slika");
    }

    if(count($greske) == 0){
        list($sirina, $visina) = getimagesize($fajl_tmpLokacija);

        $postojecaSlika = null;
        switch($fajl_tip){
            case 'image/jpeg':
                $postojecaSlika = imagecreatefromjpeg($fajl_tmpLokacija);
                break;
            case 'image/png':
                $postojecaSlika = imagecreatefrompng($fajl_tmpLokacija);
                break;
        }

        $novaVisina = 400;

        $novaSirina = ($novaVisina/$visina) * $sirina;

        $novaSlika = imagecreatetruecolor($novaSirina, $novaVisina);

        imagecopyresampled($novaSlika, $postojecaSlika, 0, 0, 0, 0,
        $novaSirina, $novaVisina, $sirina, $visina);

        $naziv = time().$fajl_naziv;
        $putanjaNovaSlika = 'assets/img/usersBackground/nova_'. $naziv;
```

```

        switch($fajl_tip){
        case 'image/jpeg':
            imagejpeg($novaSlika, '../..' . $putanjaNovaSlika, 75);
            break;
        case 'image/png':
            imagepng($novaSlika, '../..' . $putanjaNovaSlika);
            break;
        }

        $putanjaOriginalnaSlika = 'assets/img/usersBackground/' . $naziv;

        if(move_uploaded_file($fajl_tmpLokacija,
        '../..' . $putanjaOriginalnaSlika)){
            //echo "Slika je upload-ovana na server!";

            try {
                $isInserted = updateBackgroundPhoto($putanjaOriginalnaSlika,
                $putanjaNovaSlika, $idUser);

                if($isInserted){
                    //echo "Putanja do slike je upisana u bazu!";
                    header("Location: ../../index.php?page=profilUser&user=$idUser");
                }

            } catch(PDOException $ex){
                header("Location: ../../index.php?page=profilUser&user=$idUser");
                echo $ex->getMessage();
                catchErrors("updateBackgroundPhoto.php ->" . $ex->getMessage());
            }
        }

        imagedestroy($postojecaSlika);
        imagedestroy($novaSlika);

    } else {
        header("Location: ../../index.php?page=profilUser&user=$idUser");
    }
}

```

## updatePassword.php

```

<?php

session_start();
if (isset($_POST['btnUpdatePassword'])) {
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";

    $sifra = $_POST['tbPassword'];
    $idUser = $_POST['tbHidden2'];

    $greske = [];
}

```

```

    if (strlen($sifra) < 6) {
        $greske[] = "Password length must be over 6";
    }

    $data = null;
    if(count($greske) > 0){
        http_response_code(422);
        $data = $greske;
    } else{
        $md5Sifra = md5($sifra);
        try{
            $update = updateUserPassword($md5Sifra, $idUser);
            if($update){
                http_response_code(204);
                $data = $update;
            }
        } catch (PDOException $ex){
            http_response_code(500);
            $data = ['Greska' => 'Greska sa bazom'. $ex->getMessage()];
            catchErrors("updatePassword.php ->".$ex->getMessage());
        }
    }

    echo json_encode($data);
} else{
    http_response_code(400);
}

```

## updateProfile.php

```

<?php
session_start();
if(isset($_POST['btnUpdateProfile'])) {
    header("Content-type: application/json");
    require "../config/connection.php";
    require "functions.php";

    $ime = $_POST['tbFirstName'];
    $prezime = $_POST['tbLastName'];
    $email = $_POST['tbMail'];
    $opis = $_POST['tbDescription'];
    $idUser = $_POST['tbHidden1'];

    $reImePrezime = "/^[A-Z][a-z]{2,15}$/";
    $reOpis = "/^(.|\\s)*[a-zA-Z]+(.|\\s)*$/";

    $greske = [];

    if (!preg_match($reImePrezime, $ime)) {
        $greske[] = "First Name is not in good format!";
    }
    if (!preg_match($reImePrezime, $prezime)) {
        $greske[] = "Last Name is not in good format!";
    }
    if (!preg_match($reOpis, $opis)) {

```

```

        $greske[] = "Opis is not in good format!";
    }
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $greske[] = "Mail is not in good format!";
    }

    $data = null;

    if(count($greske) > 0){
        http_response_code(422);
        $data = $greske;
    } else{

        try{
            $update = updateUserProfile($ime, $prezime, $email, $opis,
$idUser);

            http_response_code(204);

        } catch (PDOException $ex){
            http_response_code(500);
            $data = ["Greska" =>"Update korisnika nije uspeo". $ex-
>getMessage()];
            catchErrors("updateProfile.php ->".$ex->getMessage());
        }
    }

    echo json_encode($data);
} else{
    http_response_code(400);
}

```

## updateProfilePhoto.php

```

<?php
session_start();
if(isset($_POST['endUpdateProfile'])){

    require_once "../config/connection.php";
    require_once "functions.php";

    // FAJL

    // $fajl = $_FILES['slika'];
    // var_dump($fajl);

    $fajl_naziv = $_FILES['profilePhoto']['name'];
    $fajl_tmpLokacija = $_FILES['profilePhoto']['tmp_name'];
    $fajl_tip = $_FILES['profilePhoto']['type'];
    $fajl_velicina = $_FILES['profilePhoto']['size'];
    $idUser = $_POST['tbHidden4'];

    $greske = [];

    $dozvoljeni_tipovi = ['image/jpg', 'image/jpeg', 'image/png'];
}

```



```

    if(!in_array($fajl_tip, $dozvoljeni_tipovi)){
        array_push($greske, "Pogresan tip fajla. - Profil slika");
    }
    if($fajl_velicina > 3000000){
        array_push($greske, "Maksimalna velicina fajla je 3MB. - Profil
slika");
    }

    if(count($greske) == 0){
        list($sirina, $visina) = getimagesize($fajl_tmpLokacija);

        $postojecaSlika = null;
        switch($fajl_tip){
            case 'image/jpeg':
                $postojecaSlika = imagecreatefromjpeg($fajl_tmpLokacija);
                break;
            case 'image/png':
                $postojecaSlika = imagecreatefrompng($fajl_tmpLokacija);
                break;
        }

        $sirinaProfile = 88;
        $visinaProfile = 88;

        $sirinaKomentar = 50;
        $visinaKomentar = 50;

        $sirinaNavigacija = 25;
        $visinaNavigacija = 25;

        $slikaProfile = imagecreatetruecolor($sirinaProfile, $visinaProfile);
        $slikaKomentar = imagecreatetruecolor($sirinaKomentar,
$visinaKomentar);
        $slikaNavigacija = imagecreatetruecolor($sirinaNavigacija,
$visinaNavigacija);

        imagecopyresampled($slikaProfile, $postojecaSlika, 0, 0, 0, 0,
$sirinaProfile, $visinaProfile, $sirina, $visina);
        imagecopyresampled($slikaKomentar, $postojecaSlika, 0, 0, 0, 0,
$sirinaKomentar, $visinaKomentar, $sirina, $visina);
        imagecopyresampled($slikaNavigacija, $postojecaSlika, 0, 0, 0, 0,
$sirinaNavigacija, $visinaNavigacija, $sirina, $visina);

        $naziv = time().$fajl_naziv;
        $putanjaNovaProfilna = 'assets/img/users/novaProfilna_'. $naziv;
        $putanjaNovaKomentar = 'assets/img/users/novaKomentar_'. $naziv;
        $putanjaNovaNavigacija = 'assets/img/users/novaNavigacija_'. $naziv;

        switch($fajl_tip){

```

```

        case 'image/jpeg':
            imagejpeg($slikaProfile, '../..' . $putanjaNovaProfilna, 75);
            imagejpeg($slikaKomentar, '../..' . $putanjaNovaKomentar, 75);
            imagejpeg($slikaNavigacija, '../..' . $putanjaNovaNavigacija,
75);

            break;
        case 'image/png':
            imagepng($slikaProfile, '../..' . $putanjaNovaProfilna);
            imagepng($slikaKomentar, '../..' . $putanjaNovaKomentar);
            imagepng($slikaNavigacija, '../..' . $putanjaNovaNavigacija);
            break;
    }

    $putanjaOriginalnaSlika = 'assets/img/users/' . $naziv;
    if(move_uploaded_file($fajl_tmpLokacija,
'../..' . $putanjaOriginalnaSlika)){
        //echo "Slika je upload-ovana na server!";
        try {
            $isInserted = updateProfilePhoto($putanjaOriginalnaSlika,
$putanjaNovaProfilna, $putanjaNovaKomentar, $putanjaNovaNavigacija, $idUser);

            if($isInserted){
                //echo "Putanja do slike je upisana u bazu!";
                header("Location:
../../index.php?page=profilUser&user=$idUser");
            }

        } catch(PDOException $ex){
            header("Location:
../../index.php?page=profilUser&user=$idUser");
            echo $ex->getMessage();
            catchErrors("updateProfilePhoto.php ->" . $ex->getMessage());
        }
    }

    imagedestroy($postojecaSlika);
    imagedestroy($slikaProfile);
    imagedestroy($slikaKomentar);
    imagedestroy($slikaNavigacija);

} else {
    header("Location: ../../index.php?page=profilUser&user=$idUser");
    //var_dump($greske);
}
}

```

## updateRole.php

```

<?php

if(isset($_POST['updateRole'])) {
    header("Content-type: application/json");
}

```

```

require "../config/connection.php";
require "functions.php";

$role = $_POST['ddlRole'];
$idUser = $_POST['hiddenIdUser'];

$reId = "/^\d*$/";

$greske = [];
if($role == 0){
    $greske[] = "Role is not in good format!";
}
if (!preg_match($reId, $idUser)) {
    $greske[] = "User id is not in good format!";
}

$data = null;

if(count($greske) > 0){
    http_response_code(422);
    $data = $greske;
    header("Location: ../../index.php?page=profilUser&user=$idUser");
} else{
    try{
        $update = updateRole($role, $idUser);

        http_response_code(204);
        header("Location: ../../index.php?page=profilUser&user=$idUser");
    } catch (PDOException $ex){
        http_response_code(500);
        $data = ["Greska" => "Update korisnikove uloge nije uspeo". $ex-
>getMessage()];
        catchErrors("UpdateRole.php ->". $ex->getMessage());
        header("Location: ../../index.php?page=profilUser&user=$idUser");
    }
}

echo json_encode($data);
} else{
    http_response_code(400);
    header("Location: ../../index.php?page=profilUser&user=$idUser");
}
}

```

## zabranaUser.php

```
<?php
ob_start();
@session_start();

$rolee = $_SESSION['user']->id_role; //id uloge korisnika koji je ulogovan
$idUserStranaa = $_SESSION['idUserDelete'];
$idUserr = $_SESSION['id_user'];

if($idUserr != $idUserStranaa){
    if($rolee != 1){
        header("Location:index.php?page=pocetna");
    }
}
```

## 3.3. Views

### 3.3.1 Fixed

#### Footer.php

```
</body>

<script src="assets/bower_components/jquery/dist/jquery.js"></script>

<!--<script
src="assets/bower_components/tether/dist/js/tether.min.js"></script>-->
<script src="assets/bower_components/bootstrap/dist/js/bootstrap.js"></script>

<script
src="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/js/toastr.min.js"
></script>
<script src="assets/js/medium.js"></script>

<script>
    const idUser = '<?php echo $_SESSION['id_user']?>';
    const idRole = '<?php echo $_SESSION['role'] ?>';
</script>

</html>
```

#### Head.php

```
<!DOCTYPE html>
<html lang="en">

<head>

    <!-- =====
    Title and Meta Tags
    ===== -->
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Medium - Social Network</title>
    <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="description" content="Add your business website description
here">
    <meta name="keywords" content="Add your, business, website, keywords,
here">
    <meta name="author" content="Add your business, website, author here">

    <!-- =====
    Favicons
    ===== -->
    <!-- <link rel="shortcut icon" href="assets/img/favicons/favicon.html">--
```

```

>
<!-- <link rel="apple-touch-icon" href="assets/img/favicons/apple-touch-
icon.html">-->
<!-- <link rel="apple-touch-icon" sizes="72x72"
href="assets/img/favicons/apple-touch-icon-72x72.html">-->
<!-- <link rel="apple-touch-icon" sizes="114x114"
href="assets/img/favicons/apple-touch-icon-114x114.html">-->
<link rel="shortcut icon" type="image/ico"
href="assets/img/favicons/ico.ico" />
<!-- TOASTR-->
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/css/toastr.min.c
ss">
<!--END TOASTR-->
<!-- =====
CSS
===== -->
<!-- Style-->
<link type="text/css"
href="assets/bower_components/bootstrap/dist/css/bootstrap.css"
rel="stylesheet" />
<link type="text/css" href="assets/css/app.css" rel="stylesheet" />

</head>

```

## Nav.php

```

<body>
<?php
ob_start();

    $idUser = $_SESSION['id_user'];
    $getUserPhoto = getUserPhoto($idUser);
    $getUser = getUser($idUser);
    $categories = getCategories();

?>
<!-- =====
Navigation Section
===== -->
<nav class="navbar navbar-light navbar-toggleable-sm bg-faded justify-content-
between" id="mainNav-2">
    <button class="navbar-toggler navbar-toggler-right" type="button" data-
toggle="collapse" data-target="#collapsingNavbar2">
        <span class="navbar-toggler-icon"></span>
    </button>
    <a href="index.php?page=pocetna" class="navbar-brand mr-0">
MEDIUM</a>&nbsp;  
    <div class="navbar-collapse collapse justify-content-between"
id="collapsingNavbar2">
        <div><!--placeholder to evenly space flexbox items and center links--

```

```

></div>
    <ul class="navbar-nav">
        <!-- Search in right of nav -->
        <li class="nav-item hidden-xs-down">
            <?php
                @$url = $_GET['page'];
                if($url == 'pocetna'):
                    <?>
                    <form class="top_search clearfix" method="post"
action="models/posts/searchPost.php">
                        <div class="top_search_con">
                            <input class="s" placeholder="Search Here ..."
type="text" id="searchPost">
                            <span class="top_search_icon"><i class="fa fa-
search"></i></span>
                            <input class="top_search_submit" type="submit">
                        </div>
                    </form>
                    <?php endif; ?>
                </li>
            <!-- Search Ends -->
        </ul>
        <ul id="slikaNav" class="nav navbar-nav">
            <a href="index.php?page=profilUser&user=<?=$idUser; ?>">
                <li><span class="avatar w-32">first_name;
?>"></span></li>
                </a><!-- /navbar-item -->
            <a href="index.php?page=profilUser&user=<?=$idUser; ?>"><li><?=$
getUser->first_name; ?></li></a>
            <?php
                $role = $_SESSION['user']->id_role; //id uloge korisnika koji je
ulogovan
                if($role == 1):
                    <?>
                    <li><a href="index.php?page=autor">Author</a></li>
                    <li><a href="index.php?page=adminPanel">AdminPanel</a></li>
                    <?php endif; ?>
                    <li><a href="models/users/logout.php">Logout</a></li>
            <!--
                <li><a href="models/users/exportExcel.php"><i class="fa fa-
user m-r-5"></i>Export Excel</a></li>-->
            </ul><!-- /navbar-nav -->
        </div><!-- /navbar-collapse -->
    </nav><!-- /nav -->

```

### 3.3.2. Pages

#### Admin.php

```

<?php
zabeleziPristupStranici();
$role = $_SESSION['user']->id_role;

```

```

if($role != 1){
    header("Location: index.php?page=pocetna");
}

?>

<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Usefull links</h4>
        </div>
        <div class="card-body table-responsive">
            <ul id="adminUl" >

                <li><a href="models/users/exportExcel.php"></i>Export
Excel</a></li>
                <li><a href="models/users/exportWord.php"></i>Export
Word</a></li>
                <li><a href="data/documentaton.pdf"></i>Documentation</a></li>
            </ul>

        </div>
    </div>
    <div class="col-md-2 col-lg-2"></div>
</div>

<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Attendance in the last 24 hours in %</h4>
        </div>
        <div class="card-body table-responsive">
            <table class="flat-table flat-table-3">
<!--          <thead>-->
<!--              <th>Page</th>-->
<!--              <th>%</th>-->
<!--          </thead>-->
                <tbody><tr>
                    <td>Login/Reg</td>
                    <td>Index</td>
                    <td>User Profile</td>
                    <td>Edit Profile</td>
                    <td>Edit Post</td>
                    <td>Comment</td>
                    <td>Admin Panel</td>
                    <td>Author</td>
                </tr>
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
            </tbody>
        </table>
    </div>
</div>
<?php

    $podaci = pristumStranamaUprocentima();
    //var_dump($podaci);

```



```

        foreach ($podaci as $podatak):

            ?>

            <td><?= $podatak; ?></td>

        <?php endforeach; ?>
        </tr>
    </tbody>
</table>

</div>
</div>
<div class="col-md-2 col-lg-2"></div>
</div>
<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Active members: <?php echo
loggedCount();?></h4>
        </div>
        <div class="card-body table-responsive">
            <table class="flat-table flat-table-3">
                <!-- <thead>-->
                <!-- <th>Page</th>-->
                <!-- <th>%</th>-->
                <!-- </thead>-->
                <tbody><tr>
                    <td>No.</td>
                    <td>First Name</td>
                    <td>Last Name</td>
                    <td>Role</td>

                <?php
                $idUserovanogUsera = @file(LOGGING_FAJL);
                $br = 1;
                foreach ($idUserovanogUsera as $id):
                    ?>

                    </tr>
                    <td><?= $br++;?></td>
                    <td><?= nameUser($id)->first_name; ?></td>
                    <td><?= nameUser($id)->last_name; ?></td>
                    <td><?= nameUser($id)->role_name; ?></td>

                    </tr>
                <?php endforeach; ?>

            </tbody>
        </table>
    </div>
</div>

```

```
<div class="col-md-2 col-lg-2"></div>
</div>
```

## Autor.php

```
<?php
zabeleziPristupStranici();
?>

<div class="row">
  <div class="col-md-2 col-lg-2"></div>
  <div class="card col-sm-12 col-md-8 col-lg-8">
    <div class="card-header card-header-info">
      <h4 class="card-title">Author's details</h4>
    </div>
    <div class="card-body table-responsive">
      <table class="flat-table flat-table-3">
        <!-- <thead>-->
        <!-- <th>Page</th>-->
        <!-- <th>%</th>-->
        <!-- </thead>-->
        <tbody><tr>
          <td>First Name</td>
          <td>Last Name</td>
          <td>Birthday</td>
          <td>Faculty</td>
          <td>ID</td>
          <td>Born</td>

        </tr>
        <tr>
          <?php
            $podaci = autor();
            ?>

          <td><?= $podaci->ime; ?></td>
          <td><?= $podaci->prezime; ?></td>
          <td><?= $podaci->datum; ?></td>
          <td><?= $podaci->skola; ?></td>
          <td><?= $podaci->index; ?></td>
          <td><?= $podaci->mesto; ?></td>

        </tr>
      </tbody>
    </table>

  </div>
</div>
<div class="col-md-2 col-lg-2"></div>
</div>

<div class="row">
  <div class="col-md-2 col-lg-2"></div>
  <div class="card col-sm-12 col-md-8 col-lg-8">
    <div class="card-header card-header-info">
      <h4 class="card-title">Author's Photo</h4>
```

```

        </div>
        <div class="card-body table-responsive">
            ime; ?>" >

        </div>
    </div>
    <div class="col-md-2 col-lg-2"></div>
</div>

```

## Comments.php

```

<?php zabeleziPristupStranici(); ?>
<div class="col-md-6">
    <div class="box">
        <form action="models/comments/insertComment.php" method="POST"
id="unosPost" style="border: solid 1px #d8e2e7;">
            <input type="hidden" name="idPost" id="idPost" value="<?=
$_GET['id']; ?>">
            <input type="hidden" name="idComm" id="idComm" value="">
            <textarea class="form-control no-border" rows="3" name="textOpis"
id="textOpis" placeholder="Enter Comment"></textarea>
            <input class="btn btn-info pull-right btn-sm" type="button"
name="btnComment" id="btnComment" value="Post" >
            <input class="btn btn-info pull-right btn-sm" type="button"
name="btnReset" id="btnReset" value="Reset" style="margin-right: 30px">
        </form>
    </div>

    <section id="sviComments" style="margin-top: 70px;">

    </section>

</div><!-- /.col-md-6 -->

<div class="col-md-1">

</div>

<!-- /.row -->
</div><!-- /.container -->
</section><!-- /.section -->

<a id="scrollup">Scroll</a>

```

## editPost.php

```

<?php
zabeleziPristupStranici();
$idUserGET = $_GET['user']; //Dohvatam id usera iz url adrese
$userGET = getUser($idUserGET);
@$_SESSION['idUserDelete'] = $idUserGET;
include ABSOLUTE_PATH."/models/zabranaUser.php";
?>

```

```

<div class="container">
  <div class="row">
    <div class="col-xs-2 col-md-2"></div>
    <div class="col-xs-8 col-md-8">
      <div class="card-body table-responsive">

        <!-- FORMA ZA UPDATE USER INFO -->
        Edit
        <form action="models/posts/updatePost.php" method="POST"
enctype="multipart/form-data">
          <input type="hidden" name="hiddenPost" id="hiddenPost">
          <div class="input">
            <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Title</p>
            <input class="main-form animated bounceInDown"
type="text" name="tbTitle" id="tbTitle" required>
          </div>

          <div class="input">
            <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Description</p>
            <!-- <input class="main-form animated bounceInDown"
type="text" name="tbDescription" id="tbDescription" required>-->
            <textarea class="form-control no-border" rows="3"
name="tbDescription" id="tbDescription" required></textarea>
          </div>

          <div class="input">
            <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Choose Category</p>
            <select class="btn btn-sm" name="ddlCategories"
id="ddlCategories" style="background-color: #1ab394;">
              <?php
                foreach ($categories as $cat):
                  ?>
                  <option value="<?= $cat->id_category; ?>"><?=
$cat->naziv; ?></option>
                <?php endforeach; ?>
              </select>
            </div>

          <div class="input">
            <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Photo is required</p>
            <a class="nav-link" href="#"
onclick="document.getElementById('postPhoto').click()"><i class="fa fa-camera
text-muted"></i></a>
            <input type="file" name="postPhoto" id="postPhoto"
style="display:none;"
onchange="document.getElementById('postPhotoValue').innerHTML=this.value;"
required/>

            <span id="postPhotoValue"></span>
          </div>
          <div class="input">

```

```

        <input type="submit" class="btn btn-success"
value="Save" name="btnUpdatePost" id="btnUpdatePost" />
    </div>

</form>
<!--// FORMA ZA UPDATE USER INFO -->

</div>
</div>
</div>
<div class="row">
    <div class="col-xs-9 col-md-9">
        <table class="table">
            <tr>
                <th>Photo</th>
                <th>Title</th>
                <th>Description</th>
                <th>Category</th>

            </tr>
        </table>
        <table class="table" id="postoviUser">

        </table>

    </div>

</div>
</div>
</div>

```

## editProfile.php

```

<?php
zabeleziPristupStranici();
$idUserGET = $_GET['user']; //Dohvatam id usera iz url adrese
@$_SESSION['idUserDelete'] = $idUserGET;
$userGET = getUser($idUserGET);
include ABSOLUTE_PATH."/models/zabranaUser.php";
?>
<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Change Personal Info</h4>
        </div>
        <div class="card-body table-responsive">

            <!-- FORMA ZA UPDATE USER INFO -->
            <form action="models/posts/updatePost.php" method="POST">
                <input type="hidden" name="tbHidden1" id="tbHidden1"

```

```

value="<?= $_GET['user']; ?>">
    <div class="input">
        <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">First Name</p>
        <input class="main-form animated bounceInDown" type="text"
name="tbFirstName" id="tbFirstName" value="<?= $getUserGET->first_name; ?>">
    </div>

    <div class="input">
        <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Last Name</p>
        <input class="main-form animated bounceInDown" type="text"
name="tbLastName" id="tbLastName" value="<?= $getUserGET->last_name; ?>">
    </div>

    <div class="input">
        <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Mail</p>
        <input class="main-form animated bounceInDown" type="text"
name="tbMail" id="tbMail" value="<?= $getUserGET->email; ?>">
    </div>

    <div class="input">
        <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Description</p>
        <textarea class="form-control animated bounceInDown"
style="border: 3px solid #a0b3b0;" name="tbDescription" id="tbDescription"><?=
$getUserGET->description; ?>

        </textarea>

    </div>
    <div class="input">
        <input type="button" class="btn btn-success" value="Save"
name="btnUpdateProfile" id="btnUpdateProfile" />
    </div>

</form>
<!--// FORMA ZA UPDATE USER INFO -->

</div>
</div>
<div class="col-md-2 col-lg-2"></div>
</div>
<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Change Password</h4>
        </div>
        <div class="card-body table-responsive">

            <!-- FORMA ZA UPDATE USER INFO -->

```

```

        <form action="models/users/updatePassword.php" method="POST">
            <input type="hidden" name="tbHidden2" id="tbHidden2"
value="<?=$_GET['user']; ?>">
            <div class="input">
                <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Password</p>
                <input class="main-form animated bounceInDown"
type="password" name="tbPassword" id="tbPassword">
            </div>

            <div class="input">
                <p style="color: #9e9e9e;font-size: 12px;font-
weight:400;">Confirm Password</p>
                <input class="main-form animated bounceInDown"
type="password" name="tbPasswordConfirm" id="tbPasswordConfirm">
            </div>
            <div class="input">
                <input type="button" class="btn btn-success" value="Save"
name="btnUpdatePassword" id="btnUpdatePassword" />
            </div>

        </form>
        <!--// FORMA ZA UPDATE USER INFO -->

    </div>
</div>
<div class="col-md-2 col-lg-2"></div>
</div>
<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Change User Role</h4>
        </div>
        <div class="card-body table-responsive">

            <!-- FORMA ZA UPDATE role PHOTO -->
            <?php
                $role = $_SESSION['user']->id_role; //id uloge korisnika koji je
ulogovan
                if($role ==1):?>
                    <form action="models/users/updateRole.php" method="post">
                        <input type="hidden" name="hiddenIdUser" value="<?=$_
$getUserGET->id_user; ?>">
                        <select name="ddlRole" class="main-form animated
bounceInDown">
                            <option value="0">Choose role</option>
                            <?php $roles = getRoles();
                            foreach ($roles as $role):
                                if($role->id_role == $getUserGET->id_role):
                                    ?>
                                    <option selected value="<?=$_ $role->id_role;
?>"><?=$_ $role->role_name; ?></option>
                                    <?php else: ?>
                                    <option value="<?=$_ $role->id_role; ?>"><?=$_ $role-
>role_name; ?></option>

```

```

        <?php
            endif;
            endforeach;
        ?>
    </select>
    <input type="submit" class="btn btn-success" value="Save"
name="updateRole" style="margin-left: 100px;" />
    </form>
    <?php endif; ?>
    <!--// FORMA ZA UPDATE Role PHOTO -->

    </div>
</div>
    <div class="col-md-2 col-lg-2"></div>
</div>
<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Change Background Photo</h4>
        </div>
        <div class="card-body table-responsive">

            <!-- FORMA ZA UPDATE BACKGROUND PHOTO -->
            <form action="models/users/updateBackgroundPhoto.php"
method="POST" enctype="multipart/form-data">
                <input type="hidden" name="tbHidden3" value="<?=$_GET['user']; ?>">
                <p style="color: #9e9e9e;font-size: 12px;font-weight:400;">Change Background Photo</p>

                <div class="input-field">

                    <button type="button"
onclick="document.getElementById('backgroundPhoto').click()" class="btn btn-info">Choose profile photo</button>
                    <span id="backgroundPhotoValue"></span>

                    <input type="file" name="backgroundPhoto"
id="backgroundPhoto" style="display:none;"
onchange="document.getElementById('backgroundPhotoValue').innerHTML=this.value;" />

                    <input type="submit" class="btn btn-success" value="Save"
name="endUpdateBackground" style="margin-left: 30px;" />
                </div>

            </form>
            <!--// FORMA ZA UPDATE BACKGORUND PHOTO -->

        </div>
    </div>
    <div class="col-md-2 col-lg-2"></div>

```



```

</div>

<div class="row">
    <div class="col-md-2 col-lg-2"></div>
    <div class="card col-sm-12 col-md-8 col-lg-8">
        <div class="card-header card-header-info">
            <h4 class="card-title">Change Profile Photo</h4>
        </div>
        <div class="card-body table-responsive">

            <!-- FORMA ZA UPDATE Profiles PHOTO -->
            <form action="models/users/updateProfilePhoto.php" method="POST"
enctype="multipart/form-data">
                <input type="hidden" name="tbHidden4" value="<?=$_GET['user']; ?>">
                <p style="color: #9e9e9e;font-size: 12px;font-weight:400;">Change Profile Photo</p>

                <div class="input-field">

                    <button type="button"
onclick="document.getElementById('profilePhoto').click()" class="btn btn-info">Choose profile photo</button>
                    <span id="profilePhotoValue"></span>

                    <input type="file" name="profilePhoto" id="profilePhoto"
style="display:none;"
onchange="document.getElementById('profilePhotoValue').innerHTML=this.value;"/>

                    <input type="submit" class="btn btn-success" value="Save"
name="endUpdateProfile" style="margin-left: 30px;" />
                </div>

            </form>
            <!--// FORMA ZA UPDATE Profile PHOTO -->

        </div>
    </div>

    <div class="col-md-2 col-lg-2">
        <?php
        $role = $_SESSION['user']->id_role; //id uloge korisnika koji je
ulogovan
        if($role ==1):?>
            <form action="models/users/deleteUser.php" method="post">
                <input type="submit" class="btn btn-danger" value="DELETE USER"
name="deleteUser" style="margin-left: 100px;" />
            </form>
        <?php endif; ?>
    </div>

```

```
</div>
</div>
```

## loginRegister.php

```
<?php zabeleziPristupStranici(); ?>
<body>
<div class="container1">
    <div class="row">
        <div class="col-sm-12">
            <ul class="tabs">
                <li><a href="#" id="1">LOGIN</a></li>
                <li><a href="#" id="2">SIGN UP</a></li>
            </ul>
        </div>
        <div class="col-sm-12">
            <form action="models/users/login.php" method="POST">
                <div class="form row" id="login">

                    <div class="col-sm-12 container">
                        <div class="input">
                            <input class="main-form animated bounceInDown"
type="text" name="loginMail" id="loginMail" placeholder=" EMAIL ADDRESS"/>
                        </div>
                        <div class="input">
                            <input class="main-form animated bounceInDown"
type="password" name="loginPassword" id="loginPassword" placeholder="
PASSWORD"/>
                        </div>
                        <div class="input">
                            <input class="main-form btn animated bounceInDown"
type="submit" name="btnLogin" id="btnLogin" value = "LOGIN"/>
                        </div>
                    </div>
                </div>
            </form>

            <form action="models/users/registration.php" method="POST">
                <div class="form row" id="register" style="display: none;">

                    <div class="col-sm-12 container">
                        <div class="input">
                            <input class="main-form animated bounceInDown"
type="text" name="first_name" id="first_name" placeholder=" FIRST NAME"/>
                        </div>
                        <div class="input">
                            <input class="main-form animated bounceInDown"
type="text" name="last_name" id="last_name" placeholder=" LAST NAME"/>
                        </div>

                        <div class="input">
                            <input class="main-form animated bounceInDown"
type="text" name="tbMail" id="tbMail" placeholder=" EMAIL ADDRESS"/>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>
```

```

        </div>
        <div class="input">
            <input class="main-form animated bounceInDown"
type="password" name="tbPassword" id="tbPassword" placeholder=" PASSWORD"/>
        </div>

        <div class="input">
            <input class="main-form btn animated bounceInDown"
type="button" name="btnRegister" id="btnRegister" value = "SIGN UP"/>
        </div>

    </div>
</div>
</form>
</div>
</div>
</div>
</body>

```

## Postovi.php

```

<?php zabeleziPristupStranici(); ?>
<div class="col-md-6">

    <div class="box">
        <form action="models/posts/insertPost.php" method="POST"
enctype="multipart/form-data" id="unosPost" style="border: solid 1px
#d8e2e7;">
            <input class="form-control no-border" type="text" name="tbTitle"
placeholder="Type title" required>
            <textarea class="form-control no-border" rows="3" name="textOpis"
placeholder="Type something..." required></textarea>

            <div class="box-footer clearfix">
                <p style="color: #7e7e7e;font-size: 12px;font-
weight:400;">Choose Category</p>
                <select class="btn btn-info btn-sm" name="ddlCategories">

                    <?php
                    foreach ($categories as $cat):
                        ?>
                        <option value="<?= $cat->id_category; ?>"><?= $cat-
>naziv; ?></option>
                    <?php endforeach; ?>
                </select>
                <ul class="nav nav-pills nav-sm">
                    <li><p style="color: #7e7e7e;font-size: 12px;font-
weight:400;">Photo is Required</p></li>
                    <li class="nav-item"><a class="nav-link" href="#"
onclick="document.getElementById('postPhoto').click()"><i class="fa fa-camera
text-muted"></i></a></li>
                </ul>
            </div>
        </form>
    </div>

```

```

        <input class="btn btn-info pull-right btn-sm" type="submit"
name="btnPost" id="btnPost" value="Post" >
        <input type="file" name="postPhoto" id="postPhoto"
style="display:none;"
onchange="document.getElementById('postPhotoValue').innerHTML=this.value;"
required/>

        <span id="postPhotoValue"></span>

    </div>

</form>
</div>
<section id="sviPostovi">

</section>
</div><!-- /.col-md-6 -->
<div class="col-md-1">
</div>

<!-- /.row -->
</div><!-- /.container -->
</section><!-- /.section -->
<a id="scrollup">Scroll</a>

```

## Postovi2.php

```

<div class="col-lg-7">
    <section id="sviPostoviUser">

    </section>
</div>

</div>
</div>
</section>
<div class="row">
    <div class="col-md-6">
    </div>

```

```

        <div class="col-md-1">
            <h3 style="color: #1ab394">END</h3>

            <!--          <i class="fa fa-refresh fa-3x fa-fw"></i>-->

        </div>
        <div class="col-md-5">

        </div>
    </div>
<a id="scrollup">Scroll</a>

```

## Profile\_details.php

```

<section class="posts-2">
    <div class="container">
        <div class="row">
            <div class="col-lg-1">
            </div>
            <div class="col-lg-4">

                <div class="tbl">
                    <div class="tbl-row">
                        <div class="tbl-cell">
                            <?php $brojPostova =
getAllPostsProfile($idUserSession); ?>

                            <span class="number"><? count($brojPostova);
?></span>Total Posts
                        </div>
<!--          <div class="tbl-cell">-->
<!--          <span class="number">544</span>Total Likes-->
<!--          </div>-->

                    </div>
                </div>

                <div class="card about-user">
                    <div class="card-body">
                        <div class="card-block">
                            <h4 class="card-title info">About</h4>
                            <p class="card-text"><? $getUserSession-
>description; ?></p>

                            <div class="text-left">
                                <p class="card-text"><strong>Full Name
: </strong> <span class="m-l-15"><? $getUserSession->first_name; ?> <? $
getUserSession->last_name; ?></span></p>
<!--          <p class="card-text"><strong>Mobile
: </strong> <span class="m-l-15">( +12) 334 5555 723</span></p>-->
<!--          <p class="card-text"><strong>Website
: </strong> <span class="m-l-15">www.themashabrand.com</span></p>-->
<!--          <p class="card-text"><strong>Location

```

```

: </strong> <span class="m-l-15">USA</span></p>-->
        </div>
    </div>
</div>
</div>

</div>

```

## Profile\_info.php

```

<section class="posts-2">
    <div class="container-fluid">
        <div class="row">
            <div class="col-md-1">

            </div>
            <div class="col-md-4">

                <div class="card widget-info-one">

                    <div class="card-block">
                        <div class="inner">
                            <div class="widget-avatar pull-left"><img alt="<?= $getUser-
>first_name; ?>" src="<?= $getUserPhoto->comment_path; ?>"></div>
                            <h5><?= $getUser->first_name; ?> <?= $getUser->last_name; ?></h5>
                            <span class="subtitle"><?= $getUser->description; ?></span>
                        </div><!-- /.inner -->
                    </div>
                    <div class="card-footer">
                        <ul class="post-view">
                            <li></li>
                            <?php
                                $brojPostova = getAllPostsProfile($_SESSION['id_user']);
                                ?>
                            <li><a href="#"><i class="fa fa-edit"></i></a><?=
count($brojPostova); ?></li>

                        </ul><!-- /.post-view -->
                    </div><!-- /.card-footer -->
                </div><!-- /.panel -->

                <div class="card widget-info-two">
                    <ul class="list-group list-group-flush">
<!--
                        <li class="list-group-item"><i class="fa fa-map-
marker"></i>&nbsp; New York</li>-->
                    <?php
                        $datumJoin = $getUser->created_at;
                        $datumObrada = date('d-m-Y', strtotime($datumJoin));

                    <?>

```

```

        <li class="list-group-item"><i class="fa fa-calendar"></i>&nbsp;
Joined <?= $datumObrada; ?></li>

    </ul>
</div>

    <section class="box-typical">
        <header class="box-typical-header-sm">People you may
know</header>
        <div class="people-rel-list">
            <ul class="people-rel-list-photos" id="usersPretraga">

                </ul>
                <form class="site-header-search">
                    <input placeholder="Search for people" type="text"
id="textUser" style="border: 1px solid #1ab394;">
                    <button type="submit">
                        <span class="fa fa-search"></span>
                    </button>
                    <div class="overlay"></div>
                </form>
            </div>
        </section>
        <h3>Categories</h3>
        <ul style="border:1px solid #d8e2e7;list-style-type: circle;">
            <?php

                foreach ($categories as $getCategory):
                    ?>

                        <li><a class="getCategory"
href="index.php?page=pocetna&category=<?= $getCategory->id_category; ?>"
style="color:black;"><?= $getCategory->naziv; ?></a></li>

                        <?php endforeach; ?>
                        <?php
                            if(isset($_GET['category'])){
                                $_SESSION['id_category'] = $_GET['category'];
                            }else{
                                unset($_SESSION['id_category']);
                            }
                        <?>

                    </ul>
                </div>

```

## Profile\_user\_info.php

```
<!-- =====
Header
===== -->
<?php
zabeleziPristupStranici();
$_SESSION['userStrana'] = $_GET['user']; //kreiranje sesije sa id-em korisnika
na cijem smo profilu
$idUserSession = $_SESSION['userStrana']; // promenljiva koja sadrzi id tog
korisnika na kojoj smo strani
$getUserPhotoSession = getUserPhoto($idUserSession); //dohvatanje slike
korisnika sa zadatim id
$getUserSession = getUser($idUserSession); //dohvatiti korisika sa zadatim id
?>
<header class="profile-bg-picture">

    
$getUserSession->first_name; ?>" class="thumb-lg img-circle">
</header><!-- /header -->

<!-- =====
User Section
===== -->
<section class="user-section">
    <div class="container">
        <div class="profile-user-box">
            <div class="row">
                <div class="col-sm-6">
<!--          Mala profilna slika korisnika -->
                 $getUserSession->first_name; ?>" class="thumb-lg img-circle">

                <div class="media-body">
                    <h4 class="m-t-5 m-b-5 ellipsis"><?> $getUserSession-
>first_name; ?> <?> $getUserSession->last_name; ?></h4>
<!--          -->
                    <?php
                        $role = $_SESSION['user']->id_role; //id uloge
korisnika koji je ulogovan
                        if($role ==1):?>
                            <p class="font-13">User Role: <?> strtoupper
(userRole($_GET['user'])->role_name); ?></p>
                        <?php endif; ?>
<!--          <p class="text-muted m-b-0"><small>California,
United States</small></p>-->
                    </div><!-- /media-body -->
                </div><!-- /col-sm-6 -->
                <div class="col-sm-6">
                    <div class="text-right">
                        <?php
                            $role = $_SESSION['user']->id_role; //id uloge
korisnika koji je ulogovan
                            if($idUser == $idUserSession || $role == 1):
```



```

        ?>
        <button type="button" class="btn btn-success waves-
effect waves-light">
            <a href="index.php?page=editProfile&user=<?=$idUserSession;?>"><i class="fa fa-user m-r-5"></i> Edit Profile</a>
        </button><br>
        <button type="button" class="btn btn-success waves-
effect waves-light" style="margin-top: 5px;">
            <a href="index.php?page=editPost&user=<?=$idUserSession;?>"><i class="fa fa-user m-r-5"></i> Edit Posts</a>
        </button>
        <?php endif; ?>
    </div><!-- /text-right -->
</div><!-- /col-sm-6 -->
</div><!-- /row -->
</div><!-- /profile-user-box -->
</div><!-- /container -->
</section><!-- /section -->

```

## 3.4 JavaScript

```
$(document).ready(function() {
    $('ul.nav li.dropdown').hover(function() {
        $(this).find('.dropdown-menu').stop(true, true).delay(200).fadeIn(500);
    }, function() {
        $(this).find('.dropdown-menu').stop(true, true).delay(200).fadeOut(500);
    });

    /*=====
    Scroll To Top
    =====*/

    //When distance from top = 250px fade button in/out
    $(window).scroll(function(){
        if ($(this).scrollTop() > 250) {
            $('#scrollup').fadeIn(300);
        } else {
            $('#scrollup').fadeOut(300);
        }
    });

    //On click scroll to top of page t = 1000ms
    $('#scrollup').click(scrollNaTop);

    //$('#html, body').animate({scrollTop:$(document).height()}, 'slow');
    /*Moj kod krece ovde*/

    //Korisnici
    $("#btnRegister").click(userRegister);
    $("#btnLogin").click(userLogin);
    $("#btnUpdateProfile").click(updateUprofile);
    $("#btnUpdatePassword").click(updatePassword);

    // Postovi
    if(url.indexOf("pocetna") != -1) {
        getPosts();
        $(document).on("click", ".getCategory", getPosts);
    }
    if(url.indexOf("profilUser") != -1){
        getUserPosts();
    }
    ///Comments
    if(url.indexOf("comment") != -1) {
        $("#btnReset").click(clearFormComment);
        $("#btnComment").click(insertComment);
        getComments();
        //// Brisanje komentara
        $(document).on("click", ".deleteComm", function (e) {
            e.preventDefault();
            let idComm = $(this).data("comm");
            $.ajax({
                url: "models/comments/deleteComm.php",
```

```

        method: "post",
        dataType: "json",
        data:{
            idComm: idComm
        },
        success: function () {
            getComments();
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
});

$(document).on("click", ".updateComm", function (e) {
    e.preventDefault();

    var idComm = $(this).data("comm");
    $("#idComm").val(idComm);

    $.ajax({
        url: "models/comments/getOneComment.php",
        method: "post",
        dataType: "json",
        data:{
            idComm: idComm
        },
        success: function (data) {
            fillFormComm(data);
            scrollNaTop();
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
});

}

//post na user strani za editovanje
getPostsProfile();

$(document).on("click", ".updatePost", function (e) {
    e.preventDefault();
    let idPost = $(this).data("idpost");
    $.ajax({
        url: "models/posts/getOnePost.php",
        method: "post",
        dataType: "json",
        data:{
            idPost: idPost
        },
        success: function (data) {
            fillForm(data);
        },
    });
});

```

```

        error: function (xhr, status, error) {
            console.log(error);
        }
    });
}); //dohvatam jedan post i punim formu podacima o njemu
$(document).on("click", ".deletePost", function (e) {
    e.preventDefault();
    let idPost = $(this).data("idpost");
    console.log(idPost);
    $.ajax({
        url: "models/posts/deletePost.php",
        method: "post",
        dataType: "json",
        data:{
            idPost: idPost
        },
        success: function (data) {
            getPostsProfile(data);
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
});

// Search po title and description
$("#searchPost").keyup(searchPosts);
$("#textUser").keyup(searchUser);
});

function scrollNaTop(){
    $("html, body").animate({ scrollTop: 0 }, 1000);
    return false;
}

window.onscroll = function() {
    //console.log(window.innerHeight + window.scrollY);
    //console.log(document.body.scrollHeight);
    if ((Math.ceil(window.innerHeight + window.scrollY)) >=
Math.ceil(document.body.scrollHeight)) {
        // you're at the bottom of the page
        console.log("Bottom of page");
        setTimeout(getPosts, 300);
        setTimeout(getUserPosts, 300);
    }
};

/*Login and Register form*/

$(".tabs a").on("click", function(){
    var id = $(this).attr("id");
    if(id == 2){
        $("#register").css("display","block");
    }
});

```

```

        $("#login").css("display","none");
    }
    else{
        $("#register").css("display","none");
        $("#forgetP").css("display","none");
        $("#login").css("display","block");
    }
});
$(".reset").on("click", function(){
    $("#login").css("display","block");
    $("#forgetP").css("display","none");
});
$(".forget-password").on("click", function(){
    $("#register").css("display","none");
    $("#login").css("display","none");
    $("#forgetP").css("display","block");
})
function animationHover(element, animation){
    element = $(element);
    element.hover(
        function() {
            element.addClass('animated ' + animation);
            //wait for animation to finish before removing classes
            window.setTimeout( function(){
                element.removeClass('animated ' + animation);
            }, 2000);
        }
    );
};
animationHover("input[type=button]", "shake");

//////////Url definisan
var url = window.location.href;

function userRegister() {
    var ime = $("#first_name").val();
    var prezime = $("#last_name").val();
    var mail = $("#tbMail").val();
    var sifra = $("#tbPassword").val();

    var reImePrezime = /^[A-Z][a-z]{2,15}$/;
    var reMail = /^[A-Za-z\d\!\#\$\%\&'\*\+\-\
\\=\?\^\_\`{\|\}\~\;\\"(\)\,:\;\<\>\@[\\\]\.]{5,}[^\.\]*\@((([a-z\d\-\
]+)\.([\w\d]+)+)$/;

    var greske =[];
    /*Klijentska validacija*/
    if(!reImePrezime.test(ime.trim())){
        greske.push("First Name is not in good format!");
        toastr.error("First Name is not in good format!");
    }
    if(!reImePrezime.test(prezime.trim())){

```

```

        greske.push("Last Name is not in good format!");
        toastr.error("Last Name is not in good format!");
    }
    if(!reMail.test(mail.trim())){
        greske.push("Mail is not in good format!");
        toastr.error("Mail is not in good format!");
    }
    if(sifra.length < 6){
        greske.push("Password length must be over 6");
        toastr.error("Password length must be over 6");
    }

    if(greske.length == 0){

        $.ajax({
            url : "models/users/registration.php",
            method : "POST",
            dataType : "json",
            data : {
                first_name : ime,
                last_name : prezime,
                tbMail : mail,
                tbPassword : sifra,
                btnRegister : true
            },
            success: function (data) {
                console.log(data);
                toastr.success("Registration = Done, You can login now <3!");
            },
            error: function (xhr, status, error) {
                console.log(error);
                var statusniKod = xhr.status;

                switch (statusniKod) {
                    case 403:
                        toastr.error("Access Denied!");
                        break;
                    case 409:
                        toastr.error("Email already exists!");
                        break;
                    case 422:
                        toastr.error("Not valid format!");
                        break;
                    case 500:
                        toastr.error("Error with DataBase!");
                        break;
                    default:
                        toastr.error("ERROR!!! Contact administrator!");
                        break;
                }
            }
        });
    }
}

```

```

}

function userLogin() {
    var mail = $("#loginMail").val();
    var sifra = $("#loginPassword").val();

    var reMail = /^[A-Za-z\d\!\#\$\%\&\'\*\+\-\
\\=\?\^\_\`\\{\|\}\~\;\\"(\)\,:\; \<\>\@\[\]\.\]{6,}[\^\.]*\@((([a-z\d\
-])\.\.[w\d]+)+)$/;

    var greske = [];
    if(!reMail.test(mail.trim())){
        greske.push("Mail is not in good format!");
    }
    if(sifra.length < 6){
        greske.push("Password length must be over 6");
    }
}

function updateUprofile() {
    var ime = $("#tbFirstName").val();
    var prezime = $("#tbLastName").val();
    var mail = $("#tbMail").val();
    var opis = $("#tbDescription").val();
    var id = $("#tbHidden1").val();

    var reImePrezime = /^[A-Z][a-z]{2,15}$/;
    var reMail = /^[A-Za-z\d\!\#\$\%\&\'\*\+\-\
\\=\?\^\_\`\\{\|\}\~\;\\"(\)\,:\; \<\>\@\[\]\.\]{6,}[\^\.]*\@((([a-z\d\
-])\.\.[w\d]+)+)$/;

    var greske =[];
    /*Klijentska validacija*/
    if(!reImePrezime.test(ime.trim())){
        greske.push("First Name is not in good format!");
        toastr.error("First Name is not in good format!");
    }
    if(!reImePrezime.test(prezime.trim())){
        greske.push("Last Name is not in good format!");
        toastr.error("Last Name is not in good format!");
    }
    if(!reMail.test(mail.trim())){
        greske.push("Mail is not in good format!");
        toastr.error("Mail is not in good format!");
    }
}

if(greske.length == 0){
    $.ajax({
        url: 'models/users/updateProfile.php',
        method: 'POST',
        dataType: 'json',
        data: {

```

```

        tbFirstName: ime,
        tbLastName : prezime,
        tbMail : mail,
        tbDescription : opis,
        tbHidden1: id,
        btnUpdateProfile: true
    },
    success: function (data) {
        toastr.success("Update = Done");
    },
    error: function (xhr, status, error) {
        var statusniKod = xhr.status;

        switch (statusniKod) {
            case 422:
                toastr.error("Not valid format");
                break;
            case 500:
                toastr.error("Email already exists!");
                break;
            case 400:
                toastr.error("Not possible at the moment");
                break;
            default:
                toastr.error("ERROR!!! Contact administrator!");
                break;
        }
    }
})
}
}

function updatePassword() {
    var sifra1 = $("#tbPassword").val();
    var sifra2 = $("#tbPasswordConfirm").val();
    var id = $("#tbHidden2").val();

    var greske = [];

    if(sifra1.length < 6){
        greske.push("Password length must be over 6");
        toastr.error("Password length must be over 6");
    }
    if(sifra1 != sifra2){
        greske.push("Passwords must match");
        toastr.error("Passwords must match!");
    }

    if(greske.length == 0){
        $.ajax({
            url: "models/users/updatePassword.php",
            method: "POST",
            dataType: "json",
            data:{
                tbPassword: sifra1,

```



```

        tbHidden2: id,
        btnUpdatePassword: true
    },
    success: function (data) {
        toastr.success("Password has been changed successfully!")
    },
    error: function (xhr, status, error) {
        statusniKod = xhr.status;
        switch (statusniKod) {
            case 422:
                toastr.error("Not valid format");
                break;
            case 500:
                toastr.error("DataBase problem!");
                break;
            case 400:
                toastr.error("Not possible at the moment");
                break;
            default:
                toastr.error("ERROR!!! Contact administrator!");
                break;
        }
    }
})
}
}

if(url.indexOf("pocetna") != -1) {
    var br = 2;

    function getPosts() {
        if ((Math.ceil(window.innerHeight + window.scrollY)) >=
Math.ceil(document.body.scrollHeight)) {
            br += 3;
        }
        // let category = $(this).data('category');

        $.ajax({
            url: "models/posts/getAllPosts.php",
            method: "post",
            dataType: "json",
            data: {
                br: br
            },
            success: function (data) {
                printPosts(data);
                console.log(br);
            },
            error: function (xhr, status, error) {
                console.log(error);
            }
        });
    }

    function printPosts(data) {

```

```

        let html = "";
        if (data != 0) {
            for (let d of data) {
                html += printPost(d);
            }
        } else {
            html += noPosts();
        }
        $("#sviPostovi").html(html);
    }
}

function printPost(post) {
    return `
        <div class="tr-section feed" style="border: 1px solid #1ab394;">
    \t\t <div>
    \t\t <div>
    \t\t <div class="mojCssSlika" >
    \t\t 
    \t\t </div><!-- /entry-thumbnail -->
    \t </div><!-- /entry-header -->
    \t\t <div class="post-content">
    \t\t <div class="author-post">
    \t\t <a href="index.php?page=profilUser&user=${post.id_user}"></a>
    \t\t </div><!-- /author -->
    \t\t <div class="entry-meta">
    \t\t <ul>
    \t\t\t <li><a href="#">${post.first_name} ${post.last_name}</a></li>
    \t\t\t <li>${obradaDatum(post.post_created_at)} ago</li>
    \t\t\t <li>${post.naziv}</li>
    \t\t </ul>
    \t\t\t</div><!-- /.entry-meta -->
    \t\t\t<h2><a href="#" class="entry-title">${post.title}</a></h2>
    \t\t\t<p>${post.post_description}</p>
    \t\t\t<div class="read-more">
    \t\t <div class="feed pull-left">
    \t\t\t <ul>
    \t\t\t\t <li><a href="index.php?page=comment&id=${post.id_post}"><i class="fa
    fa-comments"></i>See comments</a></li>&nbsp;
    \t\t\t\t </li>
    \t\t\t </ul>
    \t\t </div><!-- /feed -->
    \t\t\t</div><!-- /read-more -->
    \t </div><!-- /.post-content -->
    \t </div><!-- /.tr-post -->
    \t </div><!-- /.tr-post -->
    `
}

function obradaDatum(time) {
    timeNow = Date.now();
    vremeIzBaze = new Date(time).getTime();
    razlika = timeNow - vremeIzBaze;

```

```

var years = Math.floor(razlika / (1000 * 60 * 60 * 24) / 365);
var days = Math.floor(razlika / (24 * 60 * 60 * 1000) % 365);
var hours = Math.floor((razlika % (1000 * 60 * 60 * 24)) / (1000 * 60 *
60));
var minutes = Math.floor((razlika % (1000 * 60 * 60)) / (1000 * 60));
var seconds = Math.floor((razlika % (1000 * 60)) / 1000);

if(minutes < 60 && hours == 0 && days == 0 && years == 0){
    return minutes + ' minutes ';
} else if(hours < 24 && days == 0 && years ==0){
    return hours + ' hours ';
} else if(days < 30){
    return days + ' days ';
} else{
    return "months"
}
}

function noPosts() {
    return `
        <div class="row">

            <div class="col-md-12">
                <h3 style="color: #1ab394">No posts at the moment..</h3>

            </div>

        </div>
    `;
}

if(url.indexOf("profilUser") != -1){
    var brojac = 3;
    function getUserPosts() {
        if((Math.ceil(window.innerHeight + window.scrollY)) >=
Math.ceil(document.body.scrollHeight)){
            brojac += 3;
        }
        $.ajax({
            url: "models/posts/getAllPostsUser.php",
            method: "POST",
            dataType: "json",
            data: {
                brojac: brojac
            },
            success: function (data) {
                printPostsUser(data);
            },
            error: function (xhr, status, error) {
                console.log(error);
            }
        });
    }

    function printPostsUser(data) {

```

```

        let html = "";
        if(data !=0){
            for(let d of data){
                html+= printPost(d);
            }
        }else{
            html += noPosts();
        }
        $("#sviPostoviUser").html(html);
    }
}

function searchPosts() {
    let polje = $(this).val();

    $.ajax({
        url: "models/posts/searchPost.php",
        method: "post",
        dataType: "json",
        data: {
            post: polje
        },
        success: function (data) {
            printPosts(data);
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
}

function searchUser() {
    let user = $(this).val();
    $.ajax({
        url: "models/users/searchUser.php",
        method: 'post',
        dataType: "json",
        data:{
            user: user
        },
        success: function (data) {
            printUsers(data);
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
}

function printUsers(data) {
    let html = "";
    for(let d of data){
        html+= printUser(d);
    }
    $("#usersPretraga").html(html);
}

```

```

function printUser(data) {
return `
    <li style="border: 1px solid #1ab394;">${data.first_name}
    ${data.last_name}<a href="index.php?page=profilUser&user=${data.id_user}"
    ></a></li>
`
}
//ispis postova na strani za editovanje
function getPostsProfile() {
$.ajax({
    url: "models/posts/getAllPostsProfile.php",
    method: "post",
    dataType: 'json',
    success: function (data) {
        printPostsProfile(data);
    },
    error: function (xhr, status, error) {
        console.log(error);
    }
});
}

function printPostsProfile(data) {
    let html = "";
    for(let d of data){
        html += printOnePost(d);
    }
    $("#postoviUser").html(html);
}

function printOnePost(d) {
    return `
        <tr>
            <th></th>
            <th>${d.title}</th>
            <th>${d.post_description}</th>
            <th>${d.naziv}</th>
            <th><a class="updatePost" data-idpost="${d.id_post}">Update</a></th>
            <th><a class="deletePost" data-idpost="${d.id_post}">Delete</a></th>
        </tr>
    `
}

function fillForm(data) {
    $("#hiddenPost").val(data.id_post);
    $("#tbTitle").val(data.title);
    $("#tbDescription").val(data.post_description);
    $("#ddlCategories").val(data.id_category);
}

if(url.indexOf("comment") != -1) {
    function getComments() {
        var array = url.split('=');
    }
}

```

```

    idPost = array[array.length - 1];

    $.ajax({
        url: "models/comments/getComments.php",
        method: 'post',
        dataType: 'json',
        data: {
            idPost: idPost
        },
        success: function (data) {
            printComments(data);
        },
        error: function (xhr, status, error) {
            console.log(error);
        }
    });
}

function printComments(data) {
    let html = "";
    for (let d of data) {
        html += printComment(d);
    }
    $('#sviComments').html(html);
}

function printComment(d) {
    return `
    <div class="tr-section feed" style="border: 1px solid #1ab394; overflow:
visible;">
    \t\t <div>
    \t\t <div>
    \t         </div><!-- /entry-header -->
    \t\t <div class="post-content">
    \t\t <div class="author-post">
    \t\t <a href="index.php?page=profilUser&user=${d.id_user}"></a>
    \t\t </div><!-- /author -->
    \t\t <div class="entry-meta">
    \t\t <ul>
    \t\t\t <li><a
href="index.php?page=profilUser&user=${d.id_user}">${d.first_name}
${d.last_name}</a></li>
    \t\t\t <li>${obradaDatum(d.comment_created_at)} ago</li>
    \t\t\t ${obradaDozvole(d)}
    \t\t </ul>
    \t\t\t</div><!-- /.entry-meta -->
    \t\t\t<p>${d.text}</p>
    \t\t </div><!-- /.post-content -->
    \t </div><!-- /.tr-post -->
    `
}

```

```

function obradaDozvole(d) {
    let html = "";
    if(idUser == d.id_user || idRole == 1){
        html += `<li><a class="updateComm" data-
comm="${d.id_comment}">Edit</a></li>
        <li><a class="deleteComm" data-
comm="${d.id_comment}">Delete</a></li>`;
    }
    return html;
}

function insertComment() {
    var idComm = $("#idComm").val();

    if(idComm === ''){

        var idPost = $("#idPost").val();
        var commnet = $("#textOpis").val();

        var reIdPost = /^\\d*$/;
        var reComment = /^(.\\s)*[a-zA-Z]+(\\.\\s)*$/;

        var greske = [];

        if (!reIdPost.test(idPost)) {
            greske.push("Greska sa id");
            //toastr.error("bb");
        }
        if (!reComment.test(commnet)) {
            greske.push("Greska komentar");
            toastr.warning("You should enter comment");
        }
        if (greske.length == 0) {
            $.ajax({
                url: "models/comments/insertComment.php",
                method: 'post',
                dataType: 'json',
                data: {
                    textOpis: commnet,
                    idPost: idPost,
                    btnComment: true
                },
                success: function () {
                    toastr.success("Comment was added successfully!");
                    getComments();
                    clearFormComment();
                    $("#idPost").val(idPost);
                },
                error: function (xhr, status, error) {
                    console.log(error);

                    let statusniKod = xhr.status;
                    switch (statusniKod) {
                        case 422:

```





```
    $("#textOpis").val("");
    $("#idComm").val("");
    $("#btnComment").val("Post");
}

function fillFormComm(data) {
    $("#idPost").val(data.id_post);
    $("#textOpis").val(data.text);
    $("#idComm").val(data.id_comment);
    $("#btnComment").val("Edit");
}
}
```