

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE I KOMUNIKACIONE TEHNOLOGIJE

DOKUMENTACIJA

<https://test19129.000webhostapp.com/pphp/sajt/index.php>

Predmet: Praktikum iz PHP-a

Student: Katarina Kalanj 89/18

DOKUMENTACIJA	1
1. Uvod	4
1.1 Korišćeni programski jezici	4
1.2 Template.....	4
1.3 Opis funkcionalnosti	4
2. Organizacija	4
2.1. Organizaciona šema	4
2.2. Baza podataka	5
2.3. Sitemap.....	5
2.4. robots.txt.....	5
2.5. Stranice.....	6
2.5.1. index.php.....	6
2.6. Struktura fajlova	13
3. Kodovi.....	17
3.1. PHP	17
3.1.1. index.php.....	17
3.1.2. config.....	18
3.1.2.1. config.php.....	18
3.1.2.2. connection.php.....	19
3.1.3. models	20
3.1.3.1. comment.php	20
3.1.3.2. deleteComment.php	21
3.1.3.3. deletePost.php	21
3.1.3.4. deleteUser.php.....	22
3.1.3.5. editComment.php	22
3.1.3.6. editPost.php	23
3.1.3.7. editUser.php.....	25
3.1.3.8. excelDoc.php	28
3.1.3.9. filter.php.....	29
3.1.3.10. functions.php.....	30
3.1.3.11. login.php.....	33
3.1.3.12. logout.php	34
3.1.3.13. newPost.php.....	34
3.1.3.14. photoChange.php.....	36
3.1.3.15. register.php	38
3.1.3.16. wordDoc.php.....	40

3.1.4. views.....	40
3.1.4.1. fixed	40
3.1.4.1.1. footer.php	40
3.1.4.1.2. head.php.....	41
3.1.4.1.3. nav.php	41
3.1.4.1.4. scripts.php	43
3.1.4.2. pages	43
3.1.4.2.1. 403.php.....	43
3.1.4.2.2. account.php.....	43
3.1.4.2.3. admin.php	46
3.1.4.2.4. author.php.....	49
3.1.4.2.5. editComment.php	49
3.1.4.2.6. editPost.php	50
3.1.4.2.7. editUser.php.....	51
3.1.4.2.8. login.php.....	53
3.1.4.2.9. newPost.php.....	53
3.1.4.2.10. photoChange.php.....	55
3.1.4.2.11. post.php	55
3.1.4.2.12. posts.php	57
3.1.4.2.13. register.php	59
3.2. CSS	61
3.2.1. style.scss	61
3.3. JS.....	64
3.3.1. main.js	64
3.4. DB	69

1. Uvod

1.1 Korišćeni programski jezici

Sajt je pravljen korišćenjem jezika PHP, HTML, SASS, JavaScript, jQuery, u programu Visual Studio Code.

1.2 Template

Nije korišćen.

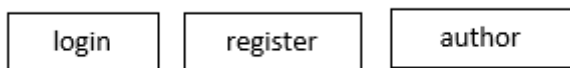
1.3 Opis funkcionalnosti

- Na svakoj stranici se dohvataju konekcija, head, navigacioni meni i footer. U zavisnosti od toga da li je korisnik logovan ili ne, da li je admin, ima pristup različitim stranicama.
- Default stranica je login, postoji link ka stranici za registraciju.
- Nakon logovanja, korisnik ima pristup ostalim stranicama sajta. Ako je obican korisnik preusmeren je na svoj profil, a ako se radi o administratoru ide na stranicu admin.
- Osnovna zamisao je blog o akvaristici, tako da korisnici mogu da prave, menjaju i brišu svoje objave, menjaju svoju profilnu sliku, komentarišu na svoje ili tuđe objave i menjaju ili brišu komentare.
- Sve objave se filteruju, sortiraju, omogućena je pretraga po naslovu.
- Admin panel sadrži podatke iz tekstualnog fajla o broju osoba koje su trenutno logovane na sajt, o najnovijim posetama stranicama kao i informacijama o tome ko ih je posetio.
- Postoji excel fajl sa podacima korisnika koje admin može da preuzme.
- Takođe ima uvid u sve korisnike i objave, koje može da ažurira ili briše.

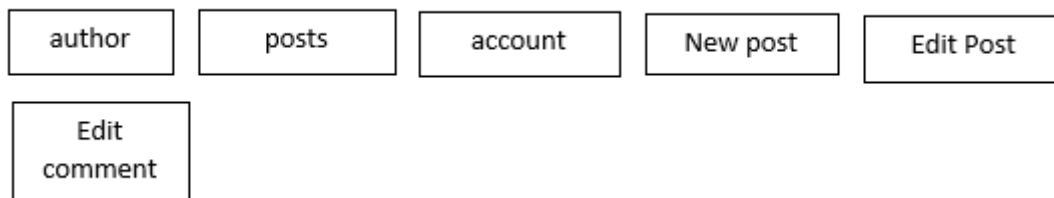
2. Organizacija

2.1. Organizaciona šema

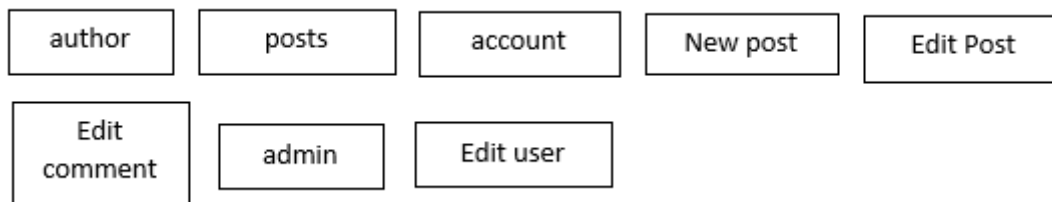
Neautorizovani korisnik



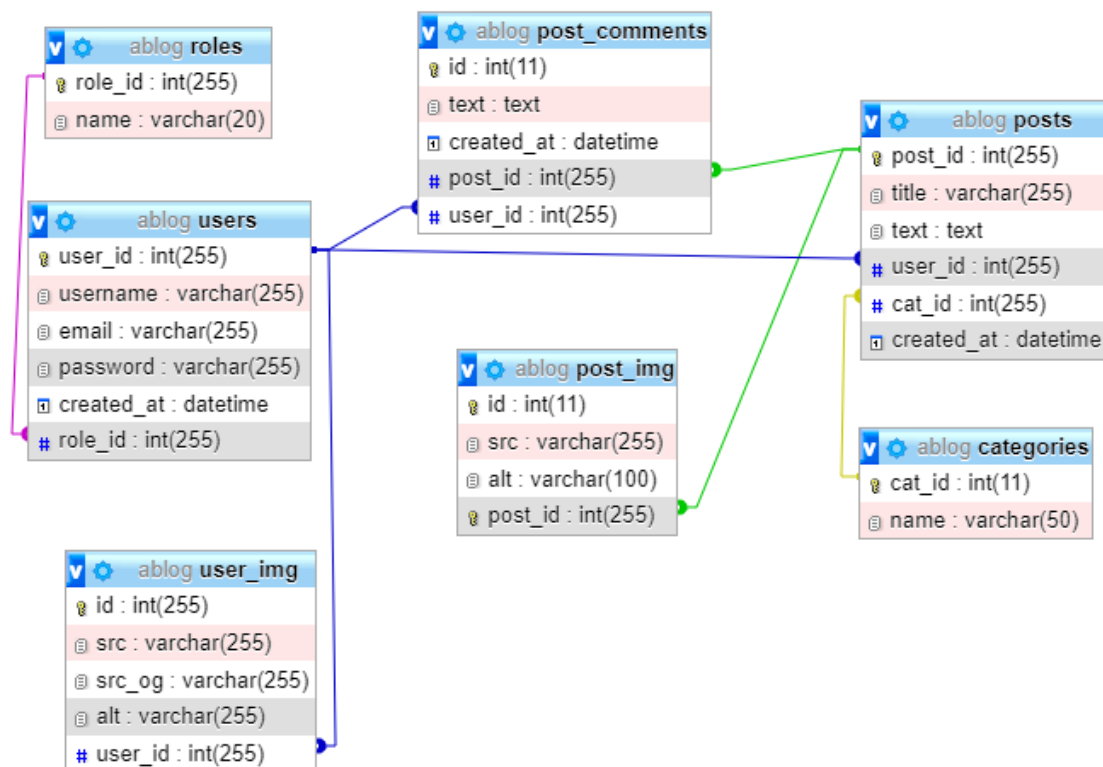
Autorizovani korisnik



Administrator



2.2. Baza podataka



2.3. Sitemap

```

<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9 http
://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">
  <url>
    <loc>https://test19129.000webhostapp.com/pphp/sajt/index.php</loc>
    <changefreq>weekly</changefreq>
    <lastmod>10.3.2021</lastmod>
    <priority>1</priority>
  </url>
</urlset>
  
```

2.4. robots.txt

User-agent: *

Disallow:/admin/

2.5. Stranice

2.5.1. index.php

AkvaBlog Dokumentacija Author Register

Log In

Username

Password

Log In

Don't have an account?
Click [here](#) to register.

Something went wrong, try again.

Inicijalni prikaz. U slučaju greške ispisuje se poruka ispod diva za logovanje.

AkvaBlog Dokumentacija Author Register

Register

Username:

Email :

Password:

Repeat Password:

Register

Registracija. Provera podataka putem JS-a pre slanja serveru.

Register

Username:

small letters or numbers, 4-20

Try again!

Email :

ex. topshrimp@gmail.com

Try again!

Password:

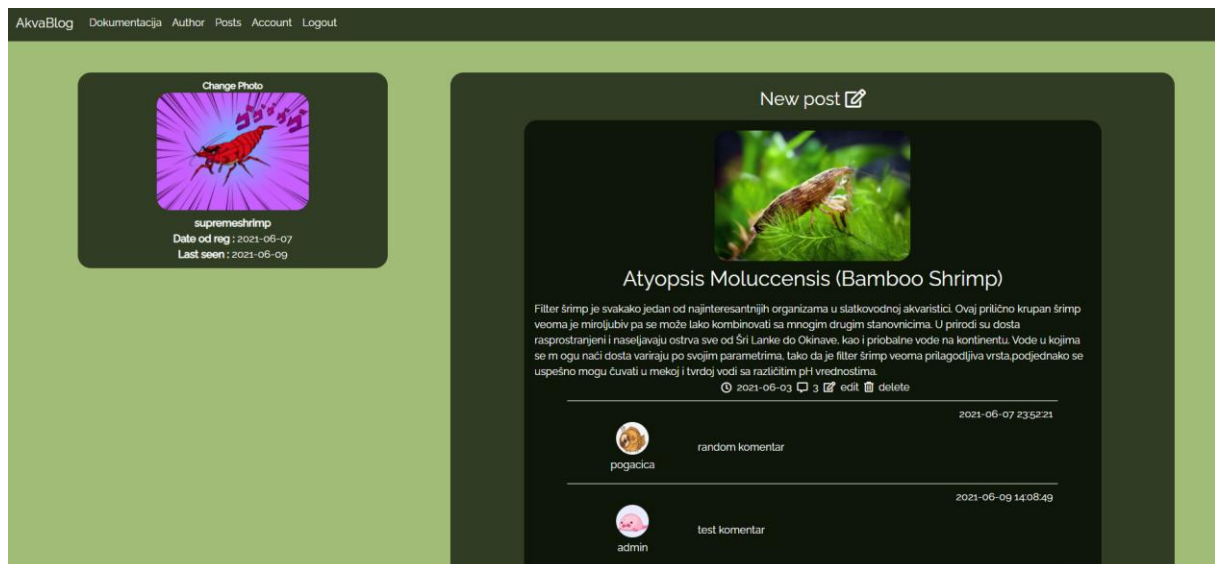
no less than 8 characters!

Try again!

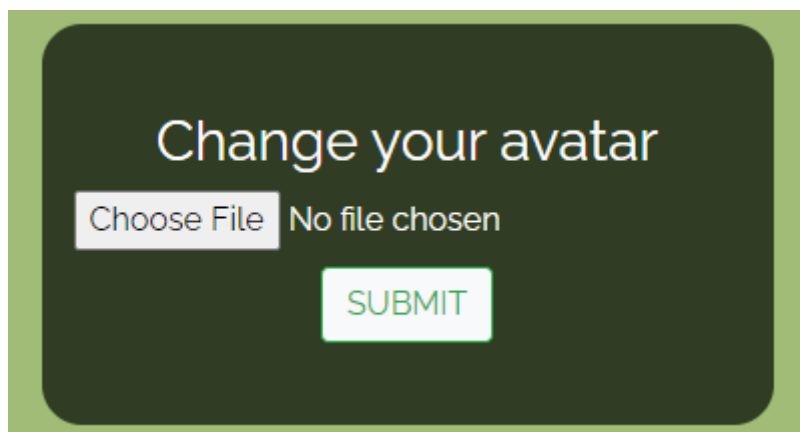
Repeat Password:

Register

Prikaz netačnih polja.



Stranica na kojoj završi korisnik koji nije admin posle logovanja. Sa leve strane je njegova slika, username, podatak o registraciji. Sa desne strane je div sa objavama korisnika, kao i sa komentarima.



Klikom na Change Photo, korisnik ide na stranicu sa formom koja će novu sliku proslediti bazi, i biće vraćen na profil. Za profil se koristi originalna slika, dok se za sliku kod komentara koristi njena smanjena varijanta.

New post

Input title *

10-100 characters

Input text *

10-1000 characters

Choose category *

Category...

Add image *

Choose File No file chosen

SUBMIT

You've made a few mistakes:

- You must upload a picture.
- Wrong title length.
- Wrong text length.
- You must choose a category.

Nova objava otvara novu stranicu, sva polja su obavezna, vrši se serverska provera. Sve greške se ispisuju ispod forme.



Procambarus Alleni (Plavi rak)

Ovi prelepi rakovi dolaze sa Floride, najviše iz močvare Everglejds. Greškom ih nazivaju plavim jastozima, što je pogrešno, jer su jastozi morski rakovi. Mogu narasti i do 15cm, i razlike između polova postoje. Mužjak ima dva dodatna pipka iza zadnjih nogu koje ženka nema, takođe imaju i izduženija klešta nego ženke. U akvarijumu umeju biti agresivni prema drugim stanovnicima, i preporučuje se da u akvarijumu budu sami. Jedu sve čega mogu da se dočepaju, ali masnu hranu izbegavaju. Jedu biljke, hranu koja padne na dno, male ribe i sporije kozice. Rado će jesti i komade šargarepe ili tikvice. Ne treba ih hraniti više od jednom dnevno. Temperatura koja im odgovara se kreće od 18 do 24 stepena, i ne treba ih držati u akvarijumu manjem od 60l. On bi trebao biti ubiljen, sa dosta mesta za skrivanje i pokriven, jer umeju izaći iz njega.

🕒 2021-06-06 💬 4 ✎ edit 🗑 delete

Klikom na edit, na profilu, korisnik pristupa formi za menjanje objave.

Edit post

Change title

Procambarus Alleni (Plavi rak)

Change text

Ovi prelepi rakovi dolaze sa Floride, najviše iz močvare Everglejds. Greškom ih nazivaju plavim jastozima, što je pogrešno, jer su jastozi morski rakovi. Mogu narasti i do 15cm, i razlike između polova postoje. Mužjak

Change image

Choose File

No file chosen

UPDATE

Podaci o specifičnoj objavi se dohvataju, ponovo se vrši provera, nije obavezno menjati sliku.

AkvaBlog
Dokumentacija
Author
Posts
Account
Admin
Logout

admin

Procambarus Alleni (Plavi rak)
2021-06-06

waterman

Paracheiroidon Axelrodi (Kraljevska neonka)
2021-06-01

Sort By:
All
Search:
SUBMIT
RESET

Strana posts sadrži objave, dok se sa desne strane nalazi filter. Na dnu strane je paginacija, objavljuje se 10 postova po stranici. Rađeno ajax-om. Klikom na naslov se otvara post, gde se mogu videti komentari, ostaviti novi, itd, izbrisati.

AkvaBlog
Dokumentacija
Author
Posts
Account
Admin
Logout



admin


Procambarus Alleni (Plavi rak)

Ovi prelepi rakovi dolaze sa Floride, najviše iz močvare Everglejds. Greškom ih nazivaju plavim jastozima, što je pogrešno, jer su jastozi morski rakovi. Mogu narasti i do 15cm, i razlike između polova postoje. Mužjak ima dva dodatna pipka iza zadnjih nogu koje ženka nema, takođe imaju i izduženja kilesta nego ženke. U akvarijumu umeju biti agresivni prema drugim stanovnicima, i preporučuje se da u akvarijumu budu sami. Jedu sve čega mogu da se dodopaju, ali mesnu hranu izbegavaju. Jedu biljke, hranu koja padne na dno, male ribe i sponje kozoce. Rado će jesti i komade šargarepe ili tikvice. Ne treba ih hraniti više od jednom dnevno. Temperatura koja im odgovara se kreće od 18 do 24 stepena, i ne treba ih držati u akvarijumu manjem od 60l. On bi trebao biti ubijen, sa dosta mesta za skrivanje i pokriven, jer umeju izaci iz njega.

2021-06-06 21:15:33
edit
delete



Pretty sweet
admin


2021-06-07 16:20:48  edit  delete



Promena komentara test

supremeshrimp

2021-06-08 00:24:43  edit  delete



At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident

jagoda

5-250 characters

Leave a comment



Admin za svaki komentar vidi edit i delete, dok ostali korisnici vide to samo za svoj komentar.


Edit comment

Change text

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident

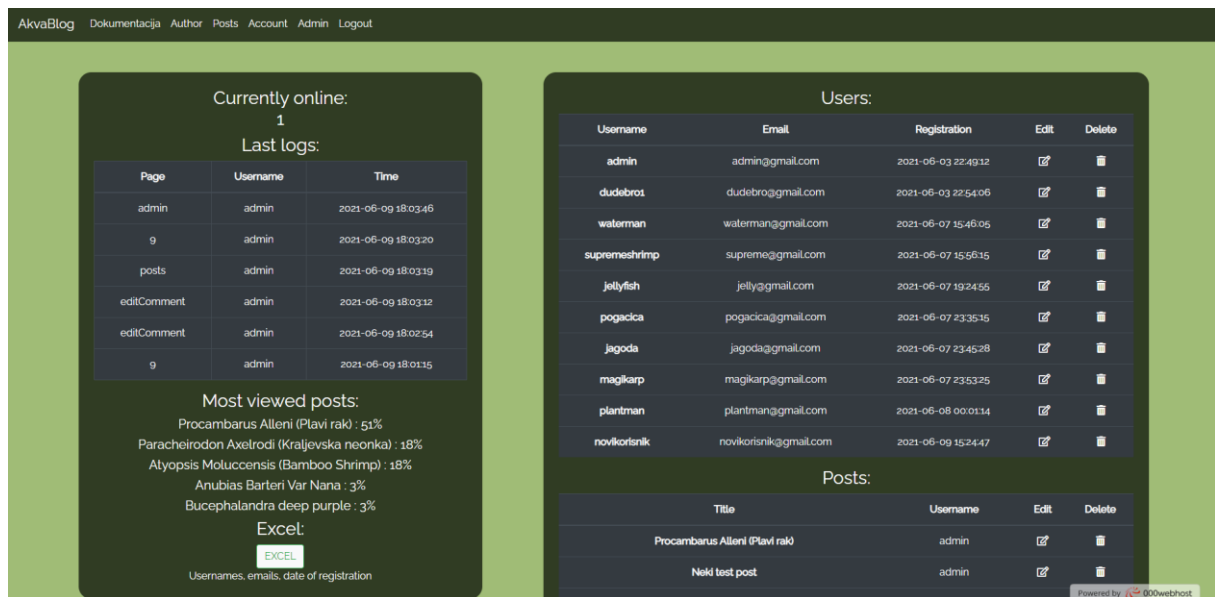
UPDATE

2021-06-08 00:24:43  edit  delete



mala promena teksta

jagoda



Admin stranica. Sa leve strane se broje svi korisnici koji su online iz tekst fajla, prikazuje se poslednjih 6 pristupa stranicama. Ispod toga se nalazi statistika o najposećenijim objavama, a zatim excel fajl sa podacima o korisnicima. Sa desne strane je lista korisnika i objava, koje admin može da ažurira ili briše.

The screenshot shows the 'Edit user' form. It includes fields for changing the username, password, email, role, and image. The current user being edited is 'novikorisnik'. The role is currently set to 'user'. There is a 'Choose File' button for the image and an 'UPDATE' button at the bottom.

Edit user

Change username - small letters or numbers, 4-20

novikorisnik

Change password - must contain uppercase, lowercase, number, spec. char, 8+

Change email - ex. topshrimp@gmail.com

novikorisnik@gmail.com

Change role

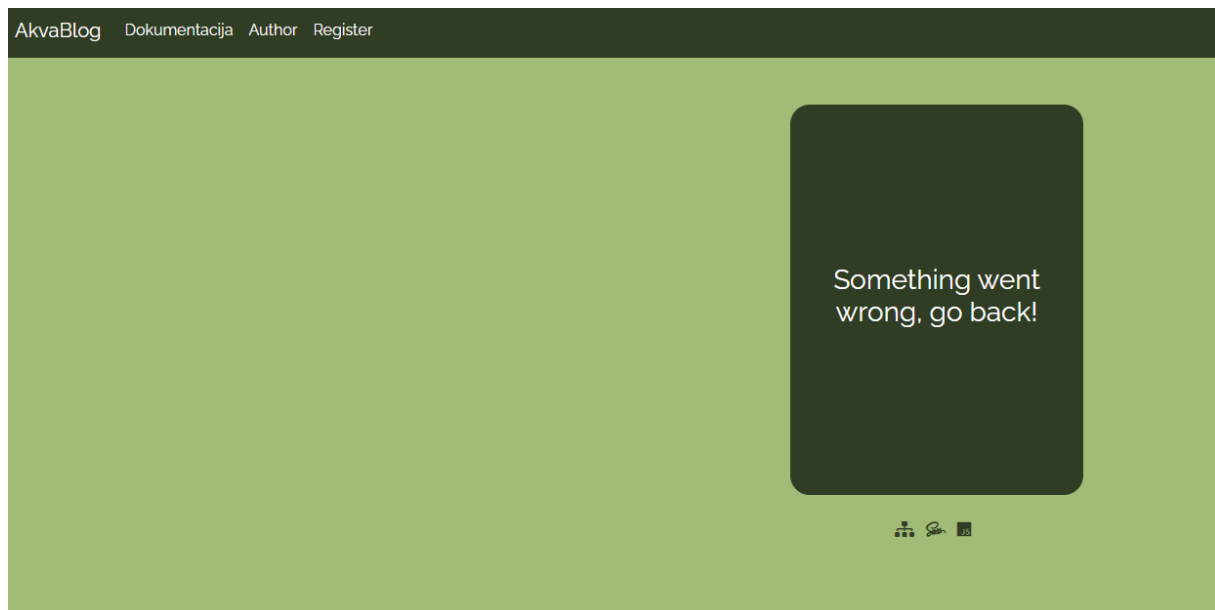
user

Change image

Choose File No file chosen

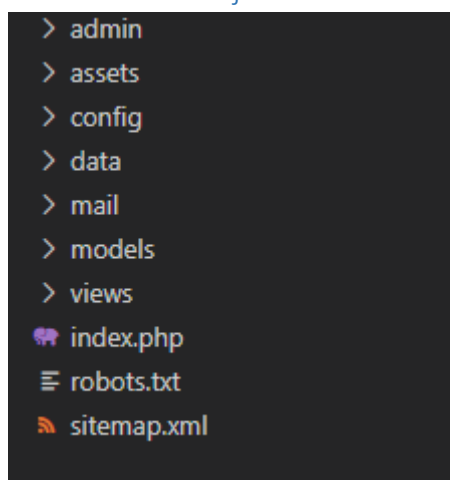
UPDATE

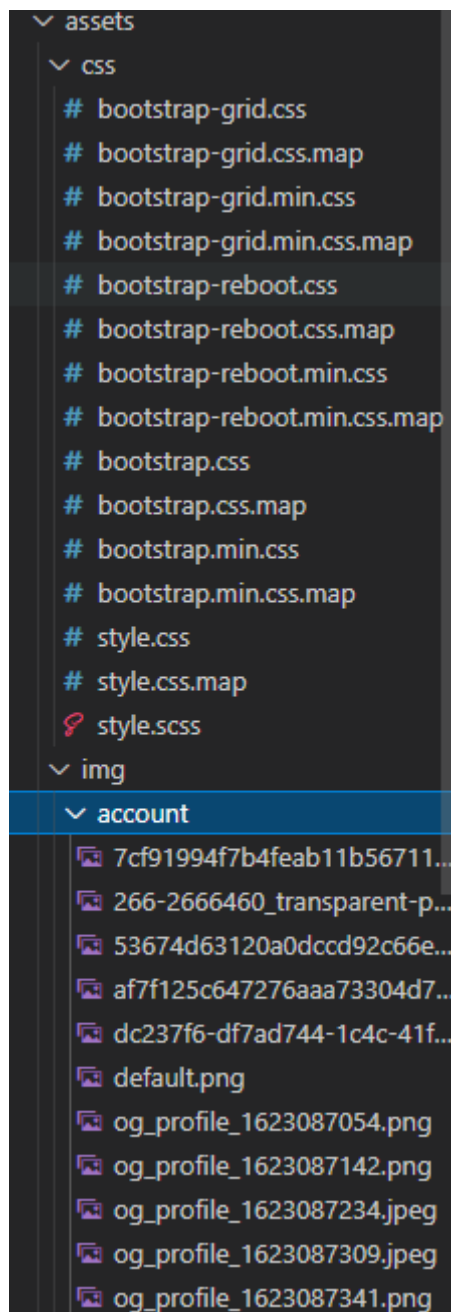
Menja sve podatke o korisnicima, kao i njihovu ulogu.

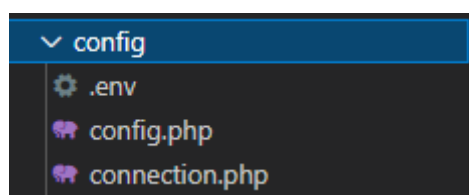
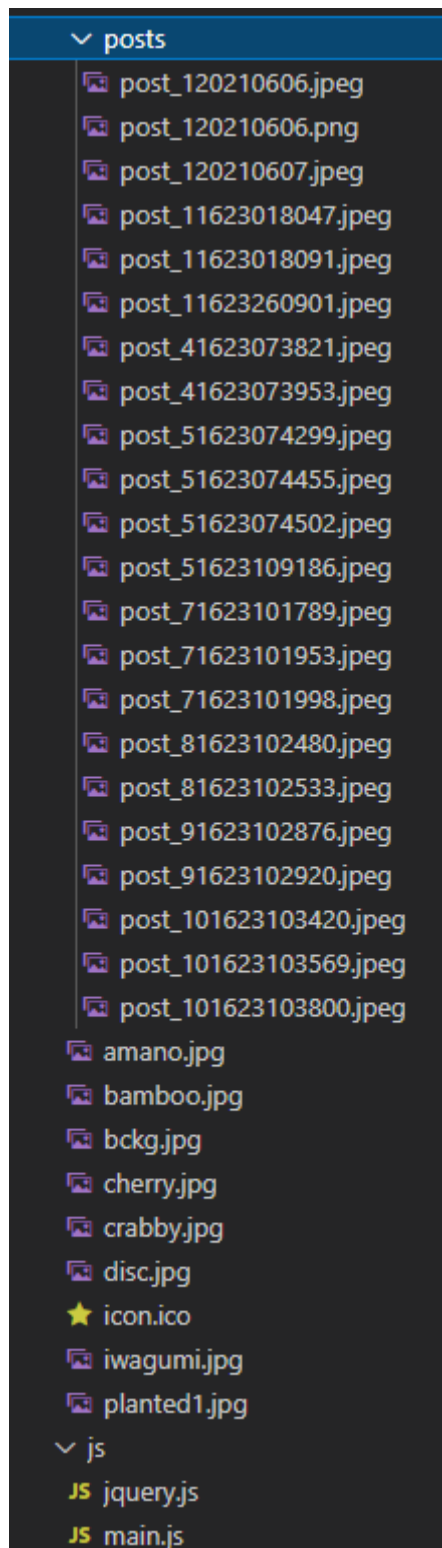


U slučaju pristupa admin stranici, ako korisnik nije admin.

2.6. Struktura fajlova







▼ data

≡ currentlyLogged.txt
≡ logs.txt

▼ models

🐘 comment.php
🐘 deleteComment.php
🐘 deletePost.php
🐘 deleteUser.php
🐘 editComment.php
🐘 editPost.php
🐘 editUser.php
🐘 excelDoc.php
🐘 filter.php
🐘 functions.php
🐘 holder.php
🐘 login.php
🐘 logout.php
🐘 mail.php
🐘 newPost.php
🐘 photoChange.php
🐘 register.php
🐘 wordDoc.php

▼ views

▼ fixed
🐘 footer.php
🐘 head.php
🐘 nav.php
🐘 scripts.php
▼ pages
🐘 403.php
🐘 account.php
🐘 admin.php
🐘 author.php
🐘 contact.php
🐘 editComment.php
🐘 editPost.php
🐘 editUser.php
🐘 login.php
🐘 newPost.php
🐘 photoChange.php
🐘 post.php
🐘 posts.php
🐘 register.php

3. Kodovi

3.1. PHP

3.1.1. index.php

```
<?php

include "views/fixed/head.php";
include "views/fixed/nav.php";

if(isset($_GET['page'])){
    switch($_GET['page'])
    {
        case 'login':
            if(isset($_SESSION['user']))
                include "views/pages/posts.php";
            else
                include "views/pages/login.php";
            break;
        case 'register':
            if(isset($_SESSION['user']))
                include "views/pages/posts.php";
            else
                include "views/pages/register.php";
            break;
        case 'posts':
            if(!isset($_SESSION['user']))
                include "views/pages/login.php";
            else
                include "views/pages/posts.php";
            break;
        case 'post':
            if(!isset($_SESSION['user']))
                include "views/pages/login.php";
            else
                include "views/pages/post.php";
            break;
        case 'account':
            if(!isset($_SESSION['user']))
                include "views/pages/login.php";
            else
                include "views/pages/account.php";
            break;
        case 'admin':
            if(isset($_SESSION['user']) && $_SESSION['user']->role_id==2)
                include "views/pages/admin.php";
            else
                include "views/pages/403.php";
            break;
        case 'editPost':
```

```

        if(!isset($_SESSION['user']))
            include "views/pages/login.php";
        else
            include "views/pages/editPost.php";
        break;
    case 'newPost':
        if(!isset($_SESSION['user']))
            include "views/pages/login.php";
        else
            include "views/pages/newPost.php";
        break;
    case 'phChange':
        if(!isset($_SESSION['user']))
            include "views/pages/login.php";
        else
            include "views/pages/photoChange.php";
        break;
    case 'editComment':
        if(!isset($_SESSION['user']))
            include "views/pages/login.php";
        else
            include "views/pages/editComment.php";
        break;
    case 'author':
        include "views/pages/author.php";
        break;
    case 'editUser':
        if(!isset($_SESSION['user']))
            include "views/pages/login.php";
        else
            include "views/pages/editUser.php";
        break;
    }
} else {
    include "views/pages/login.php";
}

include "views/fixe/footer.php";
include "views/fixe/scripts.php";
?>

```

3.1.2. config

3.1.2.1. config.php

```

<?php
    $location = $_SERVER['PHP_SELF'];
    if($location == "/pphp/sajt/config/connection.php"){
        header("Location: ../index.php");
    }
// abs

```

```

define("absolutePath", $_SERVER['DOCUMENT_ROOT'].'/pphp/sajt');
//
define("envFile", absolutePath."/config/.env");
define("logsFile", absolutePath."/data/logs.txt");
define("currentlyLogged", absolutePath."/data/currentlyLogged.txt");
// db
define("SERVER", env("SERVER"));
define("DATABASE", env("DBNAME"));
define("USERNAME", env("USERNAME"));
define("PASSWORD", env("PASSWORD"));
//baza
function env($naziv){
    $prep = fopen(envFile, "r");
    $data = file(envFile);
    $value= "";
    foreach($data as $key=>$d){
        $config = explode("=", $d);
        if($config[0]==$naziv){
            $value = trim($config[1]);
        }
    }
    return $value;
}

```

3.1.2.2. connection.php

```

<?php
    session_start();

    $location = $_SERVER['PHP_SELF'];
    if($location == "/pphp/sajt/config/connection.php"){
        header("Location: ../index.php");
    }
    require_once "config.php";

    pageAccess();
    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 pageAccess(){
        $username = "Anon";
        $page = "Login";
        $date = date("Y-m-d H:i:s");
    }

```

```

        $location = $_SERVER['PHP_SELF'];
        $ip = $_SERVER['REMOTE_ADDR'];
        $postId = 0;
        if(isset($_SESSION['user'])){
            $username = $_SESSION['user']->username;
        }
        if(isset($_GET['page'])){
            $page = $_GET['page'];
        }
        if(isset($_GET['postId'])){
            $postId = $_GET['postId'];
        }
        define("forbiddenFolder","models");
        $file = fopen(logsFile, "a");

        if($file){
            if(!strpos($_SERVER['PHP_SELF'],forbiddenFolder)){
                if($postId!=0){
                    fwrite($file, "{$location}\t\t{$postId}\t\t{$username}\t\t{$ip}\t\t{$date}\n");
                    fclose($file);
                }
                else{
                    fwrite($file, "{$location}\t\t{$page}\t\t{$username}\t\t{$ip}\t\t{$date}\n");
                    fclose($file);
                }
            }
        }
    }
    function logUser($user){
        $file = fopen(currentlyLogged, "a");
        fwrite($file, "{$user->username}\n");
        fclose($file);
    }
}
?>

```

3.1.3. models

3.1.3.1. comment.php

```

<?php
    if(isset($_POST['komentar'])){
        require_once "../config/connection.php";
        $postId = $_POST['postId'];
        $txt = $_POST['tkomentar'];
        $userId = $_SESSION['user']->user_id;
        //reg
        $textReg = '/^[A-zšđžćŠĐŽĆ 1-9]{5,250}/';
        $errors= [];
    }
}

```

```

        if(!preg_match($textReg, $txt)){
            array_push($errors, "Wrong text length.");
        }
        if(count($errors) > 0){
            $_SESSION['errors'] = $errors;
            header("Location: ../index.php?page=posts");
        }
        else{
            try{
                $query = "INSERT INTO post_comments(text, post_id, user_id) VA
LUES(:txt, :post_id, :user_id)";
                $prep = $conn->prepare($query);
                $prep->bindParam(':txt', $txt);
                $prep->bindParam(':post_id', $postId);
                $prep->bindParam(':user_id', $userId);
                $data = $prep->execute();
                if($data){
                    header("Location: ../index.php?page=post&postId=".$postId)
;
                }
            }
            catch(PDOException $ex){
                echo $ex->getMessage();
            }
        }
    }
}
?>

```

3.1.3.2. deleteComment.php

```

<?php
    if(isset($_GET['id'])){
        $commId = $_GET['id'];
        try{
            require_once "../config/connection.php";
            $query = "DELETE FROM post_comments WHERE id=$commId";
            $result = $conn->exec($query);
            header("Location: ../index.php?page=posts");
        }
        catch(PDOException $ex){
            echo $ex->getMessage();
        }
    }
}
?>

```

3.1.3.3. deletePost.php

```

<?php
    if(isset($_GET['id'])){

```

```

$postId = $_GET['id'];
try{
    require_once "../config/connection.php";
    $query = "DELETE FROM posts WHERE post_id=$postId";
    $result = $conn->exec($query);
    header("Location: ../index.php?page=account");
}
catch(PDOException $ex){
    echo $ex->getMessage();
}
}

?>

```

3.1.3.4. deleteUser.php

```

<?php
    if(isset($_GET['id'])){
        $userId = $_GET['id'];
        try{
            require_once "../config/connection.php";
            $query = "DELETE FROM users WHERE user_id=$userId";
            $result = $conn->exec($query);
            header("Location: ../index.php?page=admin");
        }
        catch(PDOException $ex){
            echo $ex->getMessage();
        }
    }

?>

```

3.1.3.5. editComment.php

```

<?php
    if(isset($_POST['update'])){
        require_once "../config/connection.php";
        $id = $_POST['postId'];
        $text = $_POST['text'];
        //reg
        $textReg = '/^[A-zšđžčćšĐŽČĆ 1-9]{5,250}/';
        $errors= [];
        //check
        if(!preg_match($textReg, $text)){
            array_push($errors, "Wrong text length.");
        }
        if(count($errors) > 0){
            $_SESSION['errors'] = $errors;
            header("Location: ../index.php?page=editComment&id=".$id);
        }
    }
}

```

```

    }
    else{
        try{
            $query = "UPDATE post_comments SET `text`=:text WHERE id = :id";

            $prep = $conn->prepare($query);
            $prep->bindParam(':text', $text);
            $prep->bindParam(':id', $id);
            $done = $prep->execute();
            if($done){
                header("Location: ../index.php?page=editComment&id=".$id."&outcome=done");
            }
            else{
                header("Location: ../index.php?page=editComment&id=".$id."&outcome=failed");
            }
        }
        catch(PDOException $ex){
            echo $ex->getMessage();
        }
    }
}
?>

```

3.1.3.6. editPost.php

```

<?php
    if(isset($_POST['update'])){
        require_once "../config/connection.php";
        $title = $_POST['title'];
        $text = $_POST['text'];
        $postId = $_POST['postId'];
        $userId = $_SESSION['user']->user_id;
        //reg
        $titleReg = '/^[A-zšđžćŠĐŽČĆ 1-9]{10,100}/';
        $textReg = '/^[A-zšđžćŠĐŽČĆ 1-9]{10,1000}/';
        $errors= [];
        //checks
        if(!preg_match($titleReg, $title)){
            array_push($errors, "Wrong title length.");
        }
        if(!preg_match($textReg, $text)){
            array_push($errors, "Wrong text length.");
        }
        if(count($errors) > 0){
            $_SESSION['errors'] = $errors;
            header("Location: ../index.php?page=editPost&id=".$postId);
        }
    }
}

```

```

        else{
            try{
                $query = "UPDATE posts SET `title`=:title,`text`=:text WHERE p
ost_id = :postId";
                $prep = $conn->prepare($query);
                $prep->bindParam(':title', $title);
                $prep->bindParam(':text', $text);
                $prep->bindParam(':postId', $postId);
                $prep->execute();
                if(isset($_FILES['slika'])){
                    if($_FILES['slika']['size'] != 0){
                        $fileName = $_FILES['slika']['name'];
                        $fileSize = $_FILES['slika']['size'];
                        $fileType = $_FILES['slika']['type'];
                        $fileTmp = $_FILES['slika']['tmp_name'];
                        $typePrep = explode('/', $fileType);
                        $type = $typePrep[1];
                        //reg
                        $types = ['image/jpeg', 'image/jpg', 'image/png'];
                        define('allowedSize', 2000000);
                        if(!in_array($fileType, $types)){
                            array_push($errors, "Wrong file type.");
                        }
                        if($fileSize > allowedSize){
                            array_push($error, "The file is too big.");
                        }
                    }
                    if(count($errors) > 0){
                        $_SESSION['errors'] = $errors;
                        header("Location: ../index.php?page=editPost&id=".$pos
tId);
                    }
                }
            }
            else{
                $path = "assets/img/posts/post_".$userId.time().".".$t
ype;

                $alt = $_SESSION['user']->username.time();
                move_uploaded_file($fileTmp, "../".$path);
                $queryImg = "UPDATE post_img SET `src`=:src,`alt`=:alt
WHERE post_id = :postId";
                $prepImg = $conn->prepare($queryImg);
                $prepImg->bindParam(':src', $path);
                $prepImg->bindParam(':alt', $alt);
                $prepImg->bindParam(':postId', $postId);
                $done = $prepImg->execute();
                if($done){
                    header("Location: ../index.php?page=editPost&id=".$
$postId."&outcome=done");
                }
            }
            else{

```



```

        header("Location: ../index.php?page=editPost&id=".$postId."&outcome=failed");
    }
}
}
else{
    header("Location: ../index.php?page=editPost&id=".$postId."&outcome=done");
}
}
}
}
}
catch(PDOException $ex){
    echo $ex->getMessage();
}
}
}
}
?>

```

3.1.3.7. editUser.php

```

<?php
    require_once "../config/connection.php";
    if($_SESSION['user']->role_id != 2){
        header("Location: ../index.php");
    }
    if(isset($_POST['update'])){
        $userId = $_POST['userId'];
        $username = $_POST['username'];
        $email = $_POST['email'];
        $inpass = $_POST['password'];
        $roleId = $_POST['role'];
        //reg
        $reUser = "/^[a-z0-9]{4,20}$/";
        $reEmail = "/^[a-z][a-z\d\_\.\-]+\@[a-z\d](\.[a-z]{2,4})+$/";
        $rePass = "/(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}/";
        $err=false;
        if(!preg_match($reUser,$username)){
            $err = true;
        }
        if(!preg_match($reEmail,$email)){
            $err = true;
        }
        if(!preg_match($rePass,$inpass)){
            $err = true;
        }
        $password = md5($inpass);
        if($err){

```

```

        echo "Something isn't quite right!";
    }
    else{
        try{
            $mailquery = "SELECT * FROM users WHERE (email = :email OR username = :username) AND NOT user_id=:userId";
            $preparemail = $conn->prepare($mailquery);
            $preparemail->bindParam(":email",$email);
            $preparemail->bindParam(":username",$username);
            $preparemail->bindParam(":userId",$userId);
            $preparemail->execute();
            $doesExist= $preparemail->fetch();
            if($doesExist){
                header("Location: ../index.php?page=editUser&outcome=failed&id=".$userId);
            }
        }
        else{
            $query = "UPDATE users SET `username`=:username,`email`=:email,`password`=:password,`role_id`=:roleId WHERE user_id = :userId";
            $prep = $conn->prepare($query);
            $prep->bindParam(":username", $username);
            $prep->bindParam(":email", $email);
            $prep->bindParam(":password", $password);
            $prep->bindParam(":roleId", $roleId);
            $prep->bindParam(":userId", $userId);
            $userInsert = $prep->execute();
            if(isset($_FILES['slika'])){
                if($_FILES['slika']['size'] != 0){
                    $fileName = $_FILES['slika']['name'];
                    $fileSize = $_FILES['slika']['size'];
                    $fileType = $_FILES['slika']['type'];
                    $fileTmp = $_FILES['slika']['tmp_name'];
                    $typePrep = explode('/', $fileType);
                    $type = $typePrep[1];
                    //reg
                    $types = ['image/jpeg', 'image/jpg', 'image/png'];
                    define('allowedSize', 2000000);
                    if(!in_array($fileType, $types)){
                        array_push($errors, "Wrong file type.");
                    }
                    if($fileSize > allowedSize){
                        array_push($error, "The file is too big.");
                    }
                    if(count($errors) > 0){
                        $_SESSION['errors'] = $errors;
                        header("Location: ../index.php?page=editUser&id=".$userId);
                    }
                }
            }
        }
    }
}
else{

```

```

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

$newHeight = 180;
$newWidth = ($newHeight/$height) * $width;

$uploadedPic = null;
switch($fileType){
    case 'image/jpeg':
        $uploadedPic = imagecreatefromjpeg($fileTmp);

        break;
    case 'image/png':
        $uploadedPic = imagecreatefrompng($fileTmp);

        break;
}
$newPic = imagecreatetruecolor($newWidth, $newHeight);

imagecopyresampled($newPic, $uploadedPic, 0, 0, 0, 0, 0, $newWidth, $newHeight, $width, $height);
$namePic = 'profile_'.time();
$path = 'assets/img/account/small_'. $namePic.'.'. $type;

$pathOg = 'assets/img/account/og_'. $namePic.'.'. $type;

switch($fileType){
    case 'image/jpeg':
        imagejpeg($newPic, '../'. $path);
        break;
    case 'image/png' :
        imagepng($newPic, '../'. $path);
        break;
}
$moved = move_uploaded_file($fileTmp, '../'. $pathOg);

if($moved){
    try{
        $alt = $username.date('Ymd');
        echo $alt;
        $query = "UPDATE user_img SET `src`=:path, `src_og`=:srcOg, `alt`=:alt WHERE user_id = :idUser ";
        $prep = $conn->prepare($query);
        $prep->bindParam(':path', $path);
        $prep->bindParam(':srcOg', $pathOg);
        $prep->bindParam(':alt', $alt);
        $prep->bindParam(':idUser', $userId);
        $result = $prep->execute();
        if($result){

```

```

                                header("Location: ../index.php?page=editUser&id=".$userId."&outcome=done");
                                }
                                else{
                                    header("Location: ../index.php?page=editUser&id=".$userId."&outcome=done");
                                }
                            }
                            catch(PDOException $ex){
                                echo $ex->getMessage();
                            }
                        }
                    }
                }
            }
        }
        else{
            header("Location: ../index.php?page=editUser&id=".$userId."&outcome=done");
        }
    }
}
catch(PDOException $ex){
    echo $ex->getMessage();
}
}
}
?>

```

3.1.3.8. excelDoc.php

```

<?php
    require_once '../config/connection.php';
    $users = $conn->query("SELECT * FROM users WHERE role_id=1")->fetchAll();
    $fileName = "User info.xls";
    header("Content-Type: application/xls");
    header("Content-Disposition: attachment; filename=$fileName");
    header("Pragma: no-cache");
    header("Expires: 0");
    if($users){
        echo "Username"."\\t\\t"."Email"."\\t\\t\\t"."Registered"."\\n";
        foreach($users as $u){
            $date = explode(" ", $u->created_at);
            echo "$u->username"."\\t\\t"."$u->email"."\\t\\t\\t"."$date[0]"."\\n";
        }
    }
}
?>

```

3.1.3.9. filter.php

```
<?php
    require_once "../config/connection.php";
    header("Content-Type: application/json");
    $data=null;
    try{
        $limit = isset($_POST['limit'])?$_POST['limit']:0;
        $broj = 10;
        $query="SELECT p.post_id as id, um.src as src, um.alt as alt, title, p
.created_at as created, username, pm.src as pSrc, pm.alt as pAlt
        FROM posts p INNER JOIN users u ON p.user_id=u.user_id INNER JOIN user
_img um ON u.user_id=um.user_id
        INNER JOIN post_img pm ON pm.post_id=p.post_id";

        if(!isset($_POST['resetuj'])){
            if(isset($_POST['txt'])){
                $txt = "'%".$_POST['txt']."'";
                $query.= " WHERE title LIKE ".$txt;
            }
            if(isset($_POST['cat'])){
                $category = $_POST['cat'];
                switch($category){
                    case 0: $query.= ""; break;
                    case 1: $query.= " AND p.cat_id=".$category; break;
                    case 2: $query.= " AND p.cat_id=".$category; break;
                    case 3: $query.= " AND p.cat_id=".$category; break;
                    case 4: $query.= " AND p.cat_id=".$category; break;
                }
            }
            if(isset($_POST['sort'])){
                $sort = $_POST['sort'];
                switch($sort){
                    case "ASC-id": $query.= " ORDER BY p.post_id ASC"; break;
                    case "DESC-
date": $query.= " ORDER BY p.created_at DESC"; break;
                    case "ASC-
date": $query.= " ORDER BY p.created_at ASC"; break;
                    case "ASC-title": $query.= " ORDER BY p.title ASC"; break;
                    case "DESC-
title": $query.= " ORDER BY p.title DESC"; break;
                }
            }
            else{
                $query.= " ORDER BY p.post_id ASC";
            }
        }
        $query.=" LIMIT $limit, $broj";
        $posts = $conn->query($query)->fetchAll();
        $num = count($posts);
    }
```

```

        $numOfPages = ceil($num / 9);
        echo json_encode([
            "posts" => $posts,
            "num" => $num,
            "pg" => $numOfPages
        ]);
        http_response_code(200);
    }
    catch(PDOException $ex){
        echo $ex->getMessage();
        http_response_code(500);
    }
?>

```

3.1.3.10. functions.php

```

<?php
require_once "config/connection.php";

function getCategories(){
    global $conn;
    $data = $conn->query("SELECT * FROM categories")->fetchAll();
    return $data;
}

function getAdminPosts(){
    global $conn;
    $data = $conn->query("SELECT * FROM posts")->fetchAll();
    return $data;
}

function getPostWithUserAndImages($postId){
    global $conn;
    $data = $conn->query("SELECT p.post_id as postId, p.title as title, p.text as txt, p.created_at as created, pm.src as postSrc, pm.alt as postAlt, username, ui.src userSrc, ui.alt as userAlt FROM posts p INNER JOIN post_img pm ON p.post_id=pm.post_id INNER JOIN users u ON u.user_id=p.user_id INNER JOIN user_img ui ON u.user_id=ui.user_id WHERE p.post_id=$postId")->fetchAll();
    return $data;
}

function getCommentsOfPost($postId){
    global $conn;
    $query = "SELECT pc.id as id, pc.text as txt, pc.created_at as created, u.username as username, u.user_id as userId, um.src as src, um.alt as alt FROM post_comments pc INNER JOIN users u ON pc.user_id=u.user_id

```

```

        INNER JOIN user_img um ON u.user_id=um.user_id
        WHERE pc.post_id = :postId
        ORDER BY pc.created_at ASC";
        $prep = $conn->prepare($query);
        $prep->bindParam(':postId', $postId);
        $prep->execute();
        $data = $prep->fetchAll();
        return $data;
    }

function getNumOfComms($postId){
    global $conn;
    $query = "SELECT COUNT(id) as num FROM post_comments WHERE post_id=:postId";
    $prep = $conn->prepare($query);
    $prep->bindParam(':postId', $postId);
    $prep->execute();
    $data = $prep->fetch();
    return $data;
}

function getAcc($userId){
    global $conn;
    $data = $conn->query("SELECT * FROM users u INNER JOIN user_img um ON u.user_id=um.user_id WHERE u.user_id=$userId")->fetch();
    return $data;
}

function getPostsOfAcc($userId){
    global $conn;
    $data = $conn->query("SELECT * FROM posts p INNER JOIN post_img pm ON p.post_id=pm.post_id WHERE p.user_id=$userId")->fetchAll();
    return $data;
}

function getPostEdit($postId){
    global $conn;
    $data = $conn->query("SELECT * FROM posts WHERE post_id=$postId")->fetch();
    return $data;
}

function getComment($commId){
    global $conn;
    $data = $conn->query("SELECT * FROM post_comments WHERE id=$commId")->fetch();
    return $data;
}

```

```

}

function getUserEdit($userId){
    global $conn;
    $data = $conn->query("SELECT * FROM users WHERE user_id=$userId")->fetch();
    return $data;
}

function getRoles(){
    global $conn;
    $data = $conn->query("SELECT * FROM roles");
    return $data;
}

//

function allPosts($limit = 0){
    global $conn;
    $query="SELECT p.post_id as id, um.src as src, um.alt as alt, title, p.created_at as created, username, pm.src as pSrc, pm.alt as pAlt
    FROM posts p INNER JOIN users u ON p.user_id=u.user_id INNER JOIN user_img um ON u.user_id=um.user_id
    INNER JOIN post_img pm ON pm.post_id=p.post_id ORDER BY p.post_id ASC
    LIMIT $limit, 10";
    $data = $conn->query($query)->fetchAll();
    return $data;
}

function numOfInitialPosts(){
    global $conn;
    $data = $conn->query("SELECT COUNT(post_id) as num FROM posts")->fetch();
    return $data;
}

function numOfPosts($posts){
    $num = count($posts);
    return $num;
}

function numOfPages($num){
    $numOfPosts = $num;
    $numOfPages = ceil($numOfPosts / 9);
    return $numOfPages;
}
?>

```


3.1.3.11. login.php

```
<?php
    if(isset($_POST['login'])){
        require_once "../config/connection.php";
        $username = $_POST['username'];
        $inpass = $_POST['password'];
        $err = false;

        $reUser = "/^[a-z0-9]{4,20}$/";
        $rePass = "/(?!.*[a-z])(?!.*[A-Z])(?!.*\d)(?!.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}/";

        if(!preg_match($reUser,$username)){
            $err = true;
        }
        if(!preg_match($rePass,$inpass)){
            $err = true;
        }
        $password = md5($inpass);
        if($err){
            header("Location: ../index.php?page=login&outcome=error");
        }
        else{
            try{
                $logquery="SELECT * FROM users WHERE username = :username AND
password = :password";
                $preparelog = $conn->prepare($logquery);
                $preparelog->bindParam(":username",$username);
                $preparelog->bindParam(":password",$password);
                $result=$preparelog->execute();

                if($result){
                    $info = $preparelog->fetch();

                    $_SESSION['user'] = $info;
                    logUser($_SESSION['user']);
                    if($_SESSION['user']->role_id=="2"){
                        header("Location: ../index.php?page=admin");
                    }

                    else if($_SESSION['user']->role_id=="1"){
                        header("Location: ../index.php?page=account");
                    }
                    else{
                        unset($_SESSION['user']);
                        header("Location: ../index.php?page=login&outcome=erro
r");
                    }
                }
            }
        }
    }
}
```

```

    }
    catch(PDOException $ex){
        echo $ex->getMessage();
    }
}
}
?>

```

3.1.3.12. logout.php

```

<?php
require_once '../config/connection.php';
if(isset($_SESSION['user'])){
    $file = fopen(currentlyLogged, 'w');
    $data = file(currentlyLogged);
    $allData = [];
    foreach($data as $d){
        $allData[] = explode('\n', $d);
        echo $allData;
    }
    for($i=0;$i<count($allData);$i++){
        if($allData[$i][0]==$_SESSION['user']->username){
            unset($allData[$i]);
        }
    }
    $newData = implode('\n', $allData);
    fwrite($file, $newData);
    fclose($file);
    unset($_SESSION['user']);
    session_destroy();
    header("Location: ../index.php");
}
else{
    header("Location: ../index.php");
}
?>

```

3.1.3.13. newPost.php

```

<?php
if(isset($_POST['submit'])){
    require_once "../config/connection.php";
    $title = $_POST['title'];
    $text = $_POST['text'];
    $userId = $_SESSION['user']->user_id;
    $category = $_POST['category'];
    //reg
    $types = ['image/jpeg', 'image/jpg', 'image/png'];
    $titleReg = '/^[A-zšđžčćŠĐŽČĆ 1-9]{10,100}/';
}

```

```

$textReg = '/^[A-zšđžććŠĐŽČĆ 1-9]{10,1000}/';
define('allowedSize', 2000000);
$errors= [];
//image
if(isset($_FILES['slika'])){
    if($_FILES['slika']['size'] == 0){
        array_push($errors, "You must upload a picture.");
    }
    else{
        $fileName = $_FILES['slika']['name'];
        $fileSize = $_FILES['slika']['size'];
        $fileType = $_FILES['slika']['type'];
        $fileTmp = $_FILES['slika']['tmp_name'];
        $typePrep = explode('/', $fileType);
        echo $fileType;
        $type = $typePrep[1];

        if(!in_array($fileType, $types)){
            array_push($errors, "Wrong file type.");
        }
        if($fileSize > allowedSize){
            array_push($error, "The file is too big.");
        }
    }
}

//checks
if(!preg_match($titleReg, $title)){
    array_push($errors, "Wrong title length.");
}
if(!preg_match($textReg, $text)){
    array_push($errors, "Wrong text length.");
}
if($category == 0){
    array_push($errors, "You must choose a category.");
}
if(count($errors) > 0){
    $_SESSION['errors'] = $errors;
    header("Location: ../index.php?page=newPost");
}
else{
    try{
        $path = "assets/img/posts/post_".$userId.time().".$type;
        move_uploaded_file($fileTmp, "../".$path);
        $query = "INSERT INTO posts( title, text, user_id, cat_id) VAL
UES( :title, :text, :userId, :category)";
        $prep = $conn->prepare($query);
    }
}

```

```

        $prep->bindParam(':title', $title);
        $prep->bindParam(':text', $text);
        $prep->bindParam(':userId', $userId);
        $prep->bindParam(':category', $category);
        $result = $prep->execute();
        if($result){
            $lastId = $conn->lastInsertId();
            $alt = $_SESSION['user']->username.time();
            $queryImg = "INSERT INTO post_img( src, alt, post_id) VALU
ES( :src, :alt, :post_id)";
            $prepImg = $conn->prepare($queryImg);
            $prepImg->bindParam(':src', $path);
            $prepImg->bindParam(':alt', $alt);
            $prepImg->bindParam(':post_id', $lastId);
            $done = $prepImg->execute();
            if($done){
                header("Location: ../index.php?page=newPost&outcome=do
ne");
            }
            else{
                header("Location: ../index.php?page=newPost&outcome=fa
iled");
            }
        }
    }
    catch(PDOException $ex){
        echo $ex->getMessage();
    }
}
}
?>

```

3.1.3.14. photoChange.php

```

<?php

if(isset($_POST['submit'])){
    require_once "../config/connection.php";

    $fileName = $_FILES['slika']['name'];
    $fileSize = $_FILES['slika']['size'];
    $fileType = $_FILES['slika']['type'];
    $fileTmp = $_FILES['slika']['tmp_name'];
    $typePrep = explode('/', $fileType);
    $type = $typePrep[1];
    $types = ['image/jpeg', 'image/jpg', 'image/png'];
    define('allowedSize', 2000000);
}

```

```

$errors= [];

if(!in_array($fileType, $types)){
    array_push($errors, "Wrong file type.");
}
if($fileSize > allowedSize){
    array_push($error, "The file is too big.");
}
if(count($errors)==0){
    list($width, $height) = getimagesize($fileTmp);
    $newHeight = 180;
    $newWidth = ($newHeight/$height) * $width;

    $uploadedPic = null;
    switch($fileType){
        case 'image/jpeg':
            $uploadedPic = imagecreatefromjpeg($fileTmp);
            break;
        case 'image/png':
            $uploadedPic = imagecreatefrompng($fileTmp);
            break;
    }
    $newPic = imagecreatetruecolor($newWidth, $newHeight);
    imagecopyresampled($newPic, $uploadedPic, 0, 0, 0, 0, $newWidth, $
newHeight, $width, $height);
    $namePic = 'profile_'.time();
    $path = 'assets/img/account/small_'. $namePic.'.'.$type;
    $pathOg = 'assets/img/account/og_'. $namePic.'.'.$type;
    switch($fileType){
        case 'image/jpeg':
            imagejpeg($newPic, '../'.$path);
            break;
        case 'image/png' :
            imagepng($newPic, '../'.$path);
            break;
    }
    $moved = move_uploaded_file($fileTmp, '../'.$pathOg);
    if($moved){
        try{
            $idUser = $_SESSION['user']->user_id;
            $alt = $_SESSION['user']->username.date('Ymd');
            $query = "UPDATE user_img SET `src`=:path,`src_og`=:srcOg,`alt`=:alt WHERE user_id = :idUser ";
            $prep = $conn->prepare($query);
            $prep->bindParam(':path', $path);
            $prep->bindParam(':srcOg', $pathOg);
            $prep->bindParam(':alt', $alt);
            $prep->bindParam(':idUser', $idUser);
            $result = $prep->execute();

```

```

        if($result){
            header("Location: ../index.php?page=account");
        }
        else{
            header("Location: ../index.php?page=account&outcome=error"
);
        }
    }
    catch(PDOException $ex){
        echo $ex->getMessage();
    }
}

}

}

?>

```

3.1.3.15. register.php

```

<?php

require_once "../config/connection.php";
if(isset($_POST["register"])){
    $username = $_POST['username'];
    $email = $_POST['email'];
    $inpass = $_POST['password'];
    $err = false;

    //reg

    $reUser = "/^[a-z0-9]{4,20}$/";
    $reEmail = "/^[a-z][a-z\d\_\.\-]+\@[a-z\d](\.[a-z]{2,4})+$/";
    $rePass = "/(?(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-
z\d@$!%*?&]{8,})/";

    if(!preg_match($reUser,$username)){
        $err = true;
    }
    if(!preg_match($reEmail,$email)){
        $err = true;
    }
    if(!preg_match($rePass,$inpass)){
        $err = true;
    }
    $password = md5($inpass);
    if($err){
        echo "Something isn't quite right!";
    }
}

```

```

    }
    else{
        try{
            $mailquery = "SELECT * FROM users WHERE email = :email OR username
= :username";
            $preparemail = $conn->prepare($mailquery);
            $preparemail->bindParam(":email",$email);
            $preparemail->bindParam(":username",$username);
            $preparemail->execute();
            $doesExist= $preparemail->fetch();
            if($doesExist){
                header("Location: ../index.php?page=register&outcome=failed");
            }
            else{
                $query = "INSERT INTO users (username,email,password,role_
id) VALUES (:username, :email, :password, 1)";
                $prep = $conn->prepare($query);
                $prep->bindParam(":username", $username);
                $prep->bindParam(":email", $email);
                $prep->bindParam(":password", $password);
                $userInsert = $prep->execute();
                if($userInsert){
                    $lastId = $conn->lastInsertId();
                    $alt = $username.date('Ymd');
                    $srcDefault = 'assets/img/account/default.png';
                    $queryImg = "INSERT INTO user_img (src,src_og,alt,user
_id) VALUES(:src,:srcog,:alt,:user_id)";
                    $prep = $conn->prepare($queryImg);
                    $prep->bindParam(':src', $srcDefault);
                    $prep->bindParam(':srcog', $srcDefault);
                    $prep->bindParam(':alt', $alt);
                    $prep->bindParam(':user_id', $lastId);
                    $done = $prep->execute();
                    if($done){
                        header("Location: ../index.php?page=register&outco
me=done");
                    }
                }
            }
        }
        catch(PDOException $ex){
            echo $ex->getMessage();
        }
    }
}

```

```
?>
```

3.1.3.16. wordDoc.php

```
<?php
header("Content-type: application/vnd.ms-word");
header("Content-Disposition: attachment; Filename=KatarinaKalanj.doc");
?>
<!DOCTYPE html PUBLIC "-//
//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1
-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="Content-Type" content="text/html; charset=Windows-
1252">
  </head>
  <body>
    <h1 class="text-center">Katarina Kalanj</h1>
    <p class="text-
justify">I'm currently a student, and you probably got this from one of my pro
jects.
        I've always loved being creative and challenging myself.
        I'm passionate about making websites; a team player who knows how
to make your ideas a reality.<br>
        Click the following link to visit my portfolio: <br>
        <br>
        <a href="https://katarinakalanj.netlify.com/">Portfolio</a>
  </body>
</html>
```

3.1.4. views

3.1.4.1. fixed

3.1.4.1.1. footer.php

```
<div id="footer">
  <div id="linkovi" class="d-flex flex-row justify-content-center">
    <div class="mt-4">
      <a href="sitemap.xml"><i class="fa fa-sitemap mr-
2 fttxt" aria-hidden="true"></i></a>
      <a href="assets/css/style.scss"><i class="fab fa-sass mr-
2 fttxt"></i></a>
      <a href="assets/js/main.js"><i class="fab fa-js mr-
2 fttxt"></i></a>
    </div>
  </div>
</div>
```


3.1.4.1.2. head.php

```
<?php
    include "config/connection.php";
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <title>AkvaBlog</title>
        <meta charset="UTF-8">
        <link rel="icon" href="assets/img/icon.ico" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" />
        <link rel="stylesheet" type="text/css" href="assets/css/style.css" />
        <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css" integrity="sha384-oS3vJWv+0UjzBfQzYUhtDYW+Pj2yciDJxpsK10YPAYjqT085Qq/1cq5FLXAZQ7Ay" crossorigin="anonymous">
        <meta name="keywords" content="akvaristika, kozice, ribice, puzevi, blog, akvarijum" />
        <meta name="description" content="Akvaristika blog" />
    </head>
```

3.1.4.1.3. nav.php

```
<nav class="navbar navbar-expand-lg navbar-dark">
    <a class="navbar-brand" href="index.php">AkvaBlog</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav mr-auto">
            <li class="nav-item active">
                <a class="nav-link" href="dokumentacija.pdf" target="_blank">Dokumentacija <span class="sr-only">(current)</span></a>
            </li>
            <li class="nav-item active">
                <a class="nav-link" href="index.php?page=author">Author <span class="sr-only">(current)</span></a>
            </li>
        </ul>
    </div>
</nav>
<?php
    if(!isset($_SESSION['user'])):
```

```
<li class="nav-item active">
  <a class="nav-
link" href="index.php?page=register">Register <span class="sr-
only">(current)</span></a>
</li>
<?php
  endif;
?>
<?php
  if(isset($_SESSION['user'])):
  ?>
  <li class="nav-item active">
    <a class="nav-link" href="index.php?page=posts">Posts <span class="sr-
only">(current)</span></a>
  </li>
  <li class="nav-item active">
    <a class="nav-
link" href="index.php?page=account">Account <span class="sr-
only">(current)</span></a>
  </li>
  <?php
    if(isset($_SESSION['user'])):
    if($_SESSION['user']->role_id=="2"):
    ?>
    <li class="nav-item active">
      <a class="nav-link" href="index.php?page=admin">Admin <span class="sr-
only">(current)</span></a>
    </li>
    <?php
      endif;
    endif;
  ?>
  <li class="nav-item active">
    <a class="nav-link" href="models/logout.php">Logout <span class="sr-
only">(current)</span></a>
  </li>
  <?php
    endif;
  ?>

</ul>
</div>
</nav>
```

3.1.4.1.4. scripts.php

```
<script src="assets/js/jquery.js" type="text/javascript"></script>
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
    <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
    <script src="assets/js/main.js" type="text/javascript"></script>
</body>

</html>
```

3.1.4.2. pages

3.1.4.2.1. 403.php

```
<div class="d-flex flex-column align-items-center justify-content-center mt-5 text-center">
    <div id="logovanje" class="p-3">
        <h3>Something went wrong, go back!</h3>
    </div>
</div>
```

3.1.4.2.2. account.php

```
<?php
    require_once "models/functions.php";
    $userId = $_SESSION['user']->user_id;
    $userData = getAcc($userId);
?>

<div class="d-flex flex-column flex-md-row align-items-center align-items-md-start justify-content-center">
    <div class="col-6 col-md-3 m-5 p-4 text-center d-flex flex-column align-items-center justify-content-center profilePersonal profilLeft">
        <div>
            <p id="hoverEdit"><a href="index.php?page=phChange">Change Photo</a></p>
            alt ?>" class="img-fluid mb-2 profilSlika">
            <p><span class="infoAcc"><?= $userData->username ?></span><br>
            <span class="infoAcc">Date od reg :</span> <?php
                $prep = $userData->created_at;
                $ceo = explode(' ', $prep);
                $date = $ceo[0];
                echo $date;
            ?><br>
            <span class="infoAcc">Last seen :</span> <?php
                echo date('Y-m-d');
            ?>
        </div>
    </div>
```

```

        </div>
    </div>
    <div class="col-11 col-md-7 m-5 p-4 profilePersonal d-flex flex-
column align-items-center justify-content-center">
        <div class="text-right">
            <h3><a href="index.php?page=newPost">New post <i class="far fa-
edit"></i></a></h3>
        </div>
        <div class="d-flex flex-column align-items-center justify-content-
center">
            <?php
                $posts = getPostsOfAcc($userId);
                if($posts==null){
                    echo "<h3>You haven't made any posts yet.</h3>";
                }
                foreach($posts as $p):
            ?>

            <div class=" d-flex flex-column align-items-center justify-
content-center p-3 mt-2 col-10 profilePersonal">
                <div class="a">
                    alt ?>" class="img-
fluid postSlika">
                </div>
                <div class="postTitle mt-2">
                    <a href="index.php?page=post&postId=<?= $p-
>post_id ?>"><h2><?= $p->title ?></h2></a>
                </div>
                <div class="mt-2">
                    <p><?= $p->text ?></p>
                </div>
                <div class="d-flex justify-content-end align-items-center">
                    <i class="far fa-clock pr-2 pl-2"></i>
                    <?php
                        $prep = $p->created_at;
                        $ceo = explode(' ', $prep);
                        $date = $ceo[0];
                        echo $date;
                    ?>
                    <i class="far fa-comment-alt pr-2 pl-2 komClick"></i>
                    <?php
                        $postId = $p->post_id;
                        $number = getNumOfComms($postId);
                        echo $number->num;
                    ?>
                    <a href="index.php?page=editPost&id=<?= $postId ?>"><i cla
ss="far fa-edit pr-2 pl-2"></i>edit</a>
                    <a href="models/deletePost.php?id=<?= $postId ?>"><i class
="far fa-trash-alt pr-2 pl-2"></i>delete</a>

```

```

        </div>
        <div class="komDiv">
        <?php
            $comms = getCommentsOfPost($postId);
            foreach($comms as $c):
        ?>
        <div class="d-flex flex-column komentari mt-2 p-2">
        <div class="d-flex justify-content-end align-items-center">
        <p><?= $c->created ?></p>
        <?php
            if($_SESSION['user']->user_id == $c-
>userId || $_SESSION['user']->role_id == 2):

                ?>
                <a href="index.php?page=editComment&id=<?= $c-
>id ?>"><i class="far fa-edit pr-2 pl-2"></i>edit</a>
                <a href="models/deleteComment.php?id=<?= $c-
>id ?>"><i class="far fa-trash-alt pr-2 pl-2"></i>delete</a>
                <?php
                    endif;
                ?>
            </div>
            <div class="d-flex">
            <div class="d-flex flex-column justify-content-center align-items-
center col-3">
                alt ?>" class="img-
fluid postIco mb-1">
                <p><?= $c->username ?></p>
            </div>
            <div class="d-flex align-items-center p-2 col-9">
                <p><?= $c->txt ?></p>
            </div>

        </div>
    </div>
    <?php
        endforeach;
    ?>
</div>
<?php
    endforeach;
?>
</div>
</div>
</div>

```

3.1.4.2.3. admin.php

```
<div class="d-flex flex-column flex-md-row align-items-center align-items-md-
start justify-content-center">
    <div class="col-11 col-md-4 m-5 p-4 text-center d-flex flex-column align-
items-center justify-content-center profilePersonal profilLeftAdmin">
        <h3>Currently online:<br>
        <?php
            $file = file(currentlyLogged);
            $number = count($file);
            echo $number;
        ?>
        </h3>
        <h3>Last logs: </h3>
        <table class="table table-bordered table-dark">
<thead>
    <tr>
        <th scope="col">Page</th>
        <th scope="col">Username</th>
        <th scope="col">Time</th>
    </tr>
</thead>
<tbody>
    <?php
        $file = file(logsFile);
        $number = count($file);
        $allData = [];
        foreach($file as $d){
            $allData[] = explode('\n', $d);
        }
        $startPos = count($allData);
        for($i=$startPos; $i>=count($allData)-5; $i--):
            $oneLine = explode("\t\t",$allData[$i-1][0]);

        ?>
    <tr>
        <td><?= $oneLine[1] ?></td>
        <td><?= $oneLine[2] ?></td>
        <td><?= $oneLine[4] ?></td>
    </tr>
    <?php
        endfor;
    ?>
</tbody>
</table>
<h3>Most viewed posts: </h3>
<?php
    $fileP = file(logsFile);
    $fileAll = [];
    foreach($fileP as $f){
```

```

        $fileAll[] = explode("\t",$f);
    }
    $filePName=[];
    foreach ($fileAll as $fa){
        $filePName[]=$fa[2];
    }

    $allPosts = $conn->query("SELECT * FROM posts")->fetchAll();
    $arrPosts = [];
    foreach($allPosts as $a){
        $arrPosts[$a->post_id]= 0;
    }
    $arrId=[];
    foreach($allPosts as $p){
        $arrId[]=$p->post_id;
    }
    $totalVisit = 0;

    foreach($filePName as $fp){
        if(in_array($fp, $arrId)){
            $arrPosts[$fp]++;
            $totalVisit++;
        }
    }
    arsort($arrPosts); //najveci prvo
    function visits($val,$totalVisit=1){
        return ((float)$val*100)/$totalVisit;
    }
    $numOfPosts = 0;
    foreach($arrPosts as $key => $val):
        $title = $conn-
>query("SELECT post_id, title FROM posts WHERE post_id=$key")->fetch();
?>
<h5><a href="index.php?page=post&postId=<?= $title->post_id ?>"><?= $title-
>title ?></a> : <?= visits($val,$totalVisit) ?>%</h5>
<?php
    var_dump(visits($val,$totalVisit));
    $numOfPosts++;
    if($numOfPosts==5){
        break;
    }
endforeach;
?>
<h3>Excel: </h3>
<a href="models/excelDoc.php"><button type="button" class="btn btn-light btn-
outline-success">EXCEL</button></a>
<p>Usernames, emails, date of registration</p>
</div>

```

```
<div class="col-11 col-md-6 m-5 p-4 profilePersonal d-flex flex-column align-items-center justify-content-center">
```

```
<h3>Users: </h3>
```

```
<table class="table table-dark text-center">
```

```
<thead>
```

```
<tr>
```

```
<th scope="col">Username</th>
```

```
<th scope="col">Email</th>
```

```
<th scope="col">Registration</th>
```

```
<th scope="col">Edit</th>
```

```
<th scope="col">Delete</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
$users = $conn->query("SELECT * FROM users")->fetchAll();
```

```
foreach($users as $u):
```

```
?>
```

```
<tr>
```

```
<th scope="row"><?= $u->username ?></th>
```

```
<td><?= $u->email ?></td>
```

```
<td><?= $u->created_at ?></td>
```

```
<td><a href="index.php?page=editUser&id=<?= $u->user_id ?>"><i class="far fa-edit"></i></a></td>
```

```
<td><a href="models/deleteUser.php?id=<?= $u->user_id ?>"><i class="fas fa-trash-alt"></i></a></td>
```

```
</tr>
```

```
<?php
```

```
endforeach;
```

```
?>
```

```
</tbody>
```

```
</table>
```

```
<h3>Posts: </h3>
```

```
<table class="table table-dark text-center">
```

```
<thead>
```

```
<tr>
```

```
<th scope="col">Title</th>
```

```
<th scope="col">Username</th>
```

```
<th scope="col">Edit</th>
```

```
<th scope="col">Delete</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
$posts = $conn-
```

```
>query("SELECT * FROM posts p INNER JOIN users u ON p.user_id=u.user_id ORDER BY p.post_id ASC")->fetchAll();
```



```

        foreach($posts as $p):
            ?>
            <tr>
                <th scope="row"><?= $p->title ?></th>
                <td><?= $p->username ?></td>
                <td><a href="index.php?page=editPost&id=<?= $p-
>post_id ?>"><i class="far fa-edit"></i></a></td>
                <td><a href="models/deletePost.php?id=<?= $p-
>post_id ?>"><i class="fas fa-trash-alt"></i></a></td>
            </tr>
            <?php
                endforeach;
            ?>
        </tbody>
    </table>
</div>
</div>

```

3.1.4.2.4. author.php

```

<div class="d-flex flex-column flex-md-row align-items-center align-items-md-
start justify-content-center">
    <div class="col-6 col-md-3 m-5 p-4 text-center d-flex flex-column align-
items-center justify-content-center profilePersonal profilLeft">
        
    </div>
    <div class="col-11 col-md-6 m-5 p-4 profileAuthor text-justify">
        <p>
            <h4>Hey, you've stumbled upon my website!</h4><br>
            My name is Katarina Kalanj.<br>
            I'm currently a student, and this is one of my projects.
            I've always loved being creative and challenging myself.
            I'm passionate about making websites; a team player who knows how
to make your ideas a reality.<br>
            Click <span class="authUnder"><a href="https://katarinakalanj.netlify.
com/" target="_blank">here</a></span> to see my portfolio.<br>
            Click <span class="authUnder"><a href="models/wordDoc.php">here</a></s
pan> to get a Word file with my info.
        </p>
    </div>
</div>

```

3.1.4.2.5. editComment.php

```

<?php
    require_once "models/functions.php";
    $commId = $_GET['id'];

```

```

        $data = getComment($commId);
    ?>
<div class="d-flex flex-column align-items-center justify-content-center text-
left">
    <div class="col-6 m-5 p-4 profilePersonal">
        <h3>Edit comment</h3>
        <div class="form-group mt-2">
            <form method="POST" action="models/editComment.php" enctype="multipart
/form-data">
                <input type="hidden" name="postId" value="<?= $commId ?>">
                <label for="text">Change text </label><br>
                <textarea class="p-2" id="text" name="text"><?= $data-
>text ?></textarea>
                <button type="submit" class="btn btn-light btn-outline-success mt-
3 mb-2" id="btnSrc" name="update">UPDATE</button>
            </form>
        <?php
            if(isset($_SESSION['errors']))){
                echo "You've made a few mistakes: <br>";
                foreach($_SESSION['errors'] as $err){
                    echo "- ".$err."<br>";
                }
                unset($_SESSION['errors']);
            }
            if(isset($_GET['outcome']))){
                if($_GET['outcome']=='done'){
                    echo "You've successfully edited your comment.";
                }
                else{
                    echo "Something went wrong, try again.";
                }
            }
        ?>
    </div>
</div>

```

3.1.4.2.6. editPost.php

```

<?php
    require_once "models/functions.php";
    $postId = $_GET['id'];
    $data = getPostEdit($postId);
    ?>
<div class="d-flex flex-column align-items-center justify-content-center text-
left">
    <div class="col-6 m-5 p-4 profilePersonal">
        <h3>Edit post</h3>
        <div class="form-group mt-2">

```

```

        <form method="POST" action="models/editPost.php" enctype="multipart/form-data">
            <input type="hidden" name="postId" value="<?= $postId ?>">
            <label for="title">Change title </label><br>
            <input type="text" class="form-control mb-1" id="newTitle" name="title" value="<?= $data->title ?>">
            <label for="text">Change text </label><br>
            <textarea class="p-2" id="text" name="text"><?= $data->text ?></textarea>
            <label for="slika">Change image </label>
            <input type="file" class="form-control-file mt-1" id="slika" name="slika">
            <button type="submit" class="btn btn-light btn-outline-success mt-3 mb-2" id="btnSrc" name="update">UPDATE</button>
        </form>
    <?php
        if(isset($_SESSION['errors'])){
            echo "You've made a few mistakes: <br>";
            foreach($_SESSION['errors'] as $err){
                echo "- ".$err."<br>";
            }
            unset($_SESSION['errors']);
        }
        if(isset($_GET['outcome'])){
            if($_GET['outcome']=='done'){
                echo "You've successfully edited a post.";
            }
            else{
                echo "Something went wrong, try again.";
            }
        }
    ?>
</div>
</div>

```

3.1.4.2.7. editUser.php

```

<?php
    require_once "models/functions.php";
    $userId = $_GET['id'];
    $data = getUserEdit($userId);
    $roles = getRoles();
?>
<div class="d-flex flex-column align-items-center justify-content-center text-left">
    <div class="col-12 col-md-5 m-5 p-4 profilePersonal">
        <h3>Edit user</h3>
        <div class="form-group mt-2">

```

```

        <form method="POST" action="models/editUser.php" enctype="multipart/form-data" onsubmit="return editCheck()">
            <input type="hidden" name="userId" value="<?= $userId ?>">
            <label for="username">Change username - small letters or numbers, 4-20</label><br>
            <input type="text" class="form-control mb-1" id="username" name="username" value="<?= $data->username ?>">
            <label for="password">Change password - must contain uppercase, lowercase, number, spec. char, 8+</label><br>
            <input type="password" class="form-control mb-1" id="password" name="password">
            <label for="email">Change email - ex. topshrimp@gmail.com</label><br>
            <input type="text" class="form-control mb-1" id="email" name="email" value="<?= $data->email ?>">
            <label for="role">Change role </label><br>
            <select class="custom-select mb-2" id="role" name="role">
                <?php
                    foreach($roles as $r):
                ?>
                <option value="<?= $r->role_id ?>"><?= $r->name ?></option>
                <?php
                    endforeach;
                ?>
            </select>
            <label for="slika">Change image </label>
            <input type="file" class="form-control-file mt-1" id="slika" name="slika">
            <button type="submit" class="btn btn-light btn-outline-success mt-3 mb-2" id="btnSrc" name="update">UPDATE</button>
        </form>
        <?php
            if(isset($_SESSION['errors'])) {
                echo "You've made a few mistakes: <br>";
                foreach($_SESSION['errors'] as $err) {
                    echo "- ".$err."<br>";
                }
                unset($_SESSION['errors']);
            }
            if(isset($_GET['outcome'])) {
                if($_GET['outcome']=='done') {
                    echo "You've successfully edited a user.";
                }
                else {
                    echo "Something went wrong, try again.";
                }
            }
        ?>
    </div>

```

```
</div>
```

3.1.4.2.8. login.php

```
<div class="d-flex flex-column align-items-center justify-content-center mt-5 text-center">
  <div id="logovanje" class="p-3">
    <h3>Log In</h3>

    <form method="POST" action="models/login.php">
      <div class="form-group mt-4">
        <label for="username">Username</label>
        <input type="text" class="form-control" id="username" name="username">
      </div>
      <div class="form-group mt-2">
        <label for="password">Password</label>
        <input type="password" class="form-control" id="password" name="password">
      </div>
      <div class="form-group mt-2 pb-3" id="loginSbmt">
        <button type="submit" class="btn btn-light btn-outline-success" id="login" name="login">Log In</button>
      </div>
    <p>
      Don't have an account?<br>
      Click <span id="regClick"><a href="index.php?page=register">here</a></span> to register.
    </p>
  </form>
</div>
<?php
  if(isset($_GET['outcome'])):
  ?>
    <div class="fttxt title text-center mb-2"><h5>Something went wrong, try again.</h5>
    </div>
  <?php
    endif;
  ?>
</div>
```

3.1.4.2.9. newPost.php

```
<div class="d-flex justify-content-center text-left">

  <div class="col-6 m-5 p-4 profilePersonal">
    <h3>New post</h3>
    <div class="form-group mt-2">
```

```

        <form method="POST" action="models/newPost.php" enctype="multipart/form-data">
            <label for="title">Input title *</label><br>
            <input type="text" class="form-control mb-1" id="newTitle" name="title" placeholder="10-100 characters">
            <label for="text">Input text *</label><br>
            <textarea class="p-2" id="text" name="text" placeholder="10-1000 characters"></textarea>
            <label for="category">Choose category *</label><br>
            <select class="custom-select mb-2" id="newCategory" name="category">
                <option selected value="0">Category...</option>
                <?php
                    require_once "models/functions.php";
                    $categories = getCategories();
                    foreach($categories as $cat):
                ?>
                <option value="<?= $cat->cat_id ?>"><?= $cat->name ?></option>
                <?php
                    endforeach;
                ?>
            </select>
            <label for="slika">Add image *</label>
            <input type="file" class="form-control-file mt-1" id="slika" name="slika">
            <button type="submit" class="btn btn-light btn-outline-success mt-3 mb-2" id="btnSrc" name="submit">SUBMIT</button>
        </form>
    </div>
    <?php
        if(isset($_SESSION['errors'])) {
            echo "You've made a few mistakes: <br>";
            foreach($_SESSION['errors'] as $err) {
                echo "- ".$err."<br>";
            }
            unset($_SESSION['errors']);
        }
        if(isset($_GET['outcome'])) {
            echo "You've successfully made a post.";
        }
    ?>

</div>
</div>

```

3.1.4.2.10. photoChange.php

```
<div class="d-flex flex-column align-items-center justify-content-center mt-5 text-center">
    <div id="phChange" class="d-flex flex-column align-items-center justify-content-center p-3">
        <h3>Change your avatar</h3>

        <form method="POST" action="models/photoChange.php" enctype="multipart/form-data">

<input type="file" class="form-control-file" id="slika" name="slika">
<button type="submit" class="btn btn-light btn-outline-success mt-2" id="submit" name="submit">SUBMIT</button>

</form>
    </div>
    <?php
        if(isset($_GET['outcome'])):
            ?>
            <div class="fttxt title text-center mb-2"><h5>Something went wrong, try again.</h5>
            </div>
            <?php
                endif;
            ?>
        </div>
```

3.1.4.2.11. post.php

```
<?php
    require_once "models/functions.php";
?>
<div class="d-flex justify-content-center">
    <div id="posts col-9">
<?php
    $postId = $_GET['postId'];
    $data = getPostWithUserAndImages($postId);
    foreach($data as $d):
?>
<div class="d-flex justify-content-center m-5 text-center col-5 col-sm-11">
    <div class="d-flex flex-column align-items-center justify-content-center postAu col-2">
        userAlt ?>" class="img-fluid postIco">
        <p><?= $d->username ?></p>
    </div>
    <div class="postBckg d-flex flex-column p-3 col-9">
        <div class="a">
```

```

        postAlt ?>" class="img-
fluid postSlika">
    </div>
    <div class="postTitle mt-2">
        <h2><?= $d->title ?></h2>
    </div>
    <div class="text-justify">
        <p><?= $d->txt ?></p>
    </div>
    <div class="d-flex justify-content-end align-items-center">
        <i class="far fa-clock pr-2 pl-2"></i>
        <?php
            $prep = $d->created;
            $ceo = explode(' ', $prep);
            $date = $ceo[0];
            echo $date;
        ?>
        <i class="far fa-comment-alt pr-2 pl-2 komClick"></i>
        <?php
            $postId = $d->postId;
            $number = getNumOfComms($postId);
            echo $number->num;
        ?>
    </div>
<div class="komDiv">

    <?php
        $comms = getCommentsOfPost($postId);
        foreach($comms as $c):
    ?>
    <div class="d-flex flex-column komentari mt-2 p-2">
        <div class="d-flex justify-content-end align-items-center">
            <p><?= $c->created ?></p>
            <?php
                if($_SESSION['user']->user_id == $c-
>userId || $_SESSION['user']->role_id == 2):
                ?>
                <a href="index.php?page=editComment&id=<?= $c-
>id ?>"><i class="far fa-edit pr-2 pl-2"></i>edit</a>
                <a href="models/deleteComment.php?id=<?= $c-
>id ?>"><i class="far fa-trash-alt pr-2 pl-2"></i>delete</a>
                <?php
                    endif;
                ?>
            </div>
            <div class="d-flex">
                <div class="d-flex flex-column justify-content-center align-items-
center col-3">

```



```

        alt ?>" class="img-fluid postIco mb-1">
        <p><?= $c->username ?></p>
    </div>
    <div class="d-flex align-items-center p-2 col-9">
        <p><?= $c->txt ?></p>
    </div>

</div>
</div>
<?php
    endforeach;
?>
    <div class="komentar d-flex flex-column align-items-center justify-content-center p-2 ">
        <form method="POST" action="models/comment.php">
            <input type="hidden" name="postId" value="<?= $d->postId ?>">
            <textarea class="p-2 mt-2" name="tkomentar" placeholder="5-250 characters"></textarea>
            <button type="submit" class="btn btn-light btn-outline-success mt-2" id="komentar" name="komentar">Leave a comment</button>
        </form>
    </div>
</div>
</div>
</div>
<?php
    endforeach;
?>
</div>
</div>

```

3.1.4.2.12. posts.php

```

<?php
    require_once "models/functions.php";
    $posts = allPosts();
    $numOfPosts = numOfInitialPosts();
    $numOfPages = numOfPages($numOfPosts->num);
?>
<div class="d-flex flex-column flex-md-row justify-content-center">
    <div class="col-12 col-md-9 order-2 order-md-1">
        <div id="sviPostovi">
            <?php
                foreach($posts as $p):
                    $dat = explode(' ', $p->created);
                ?>
            </div>
        </div>
    </div>
</div>

```

```

        <div class="d-flex flex-column flex-md-row justify-content-center m-
5 text-center col-11">
            <div class="d-flex flex-column align-items-center justify-content-
center postAu col-2">
                alt ?>" class="img-
fluid postIco">
                <p><?= $p->username ?></p>
            </div>
            <div class="postBckgSmall d-flex flex-column p-3 col-9">
                <div class="a">
                    pAlt ?>" class="img-fluid postSlika">
                </div>
                <div class="postTitle mt-2">
                    <a href="index.php?page=post&postId=<?= $p->id ?>"><h2><?= $p-
>title ?></h2></a>
                </div>
                <div class="d-flex justify-content-end align-items-center">
                    <i class="far fa-clock pr-2 pl-2"></i> <?= $dat[0] ?>
                </div>
            </div>
        </div>
        <?php
        endforeach;
    ?>
</div>
<div class="d-flex justify-content-center col-12">
    <ul id="pagination" class="d-flex flex-row justify-content-center text-
center col-12">
        <?php
        for($i=1;$i<=$numOfPages;$i++):
        ?>
            <li class="paginacija m-2"><a href="#" class="paginationNum" data-
limit="<?= $i ?>"><?= $i ?></a></li>
        <?php
        endfor;
        ?>
    </ul>
</div>
</div>
<div id="filter" class="col-12 col-md-3 order-1 order-nd-2">
    <div class="m-5 d-flex justify-content-center">
        <div class="form-group d-flex flex-column justify-content-
center">
            <select class="custom-select mt-2 mb-
2" id="ddSort" name="ddSort">
                <option value="ASC-id" selected>Sort By...</option>
                <option value="DESC-date">Newest</option>
                <option value="ASC-date">Oldest</option>
            </select>
        </div>
    </div>
</div>

```

```

        <option value="ASC-title">A-Z</option>
        <option value="DESC-title">Z-A</option>
    </select>
    <select class="custom-select mt-2 mb-
2" id="ddCategory" name="ddCategory">
        <option value="0" selected>All</option>
        <?php
            $categories = getCategories();
            foreach($categories as $cat):
                <?php
                    <option value="<?= $cat->cat_id ?>"><?= $cat-
>name ?></option>
                <?php
                    endforeach;
                <?php
            </select>
            <input type="text" class="form-control mt-2 mb-
2" id="srchTxt" name="srchTxt" placeholder="Search...">

            <button type="button" class="btn btn-light btn-outline-
success mt-1 mb-2" id="filSub">SUBMIT</button>
            <button type="button" class="btn btn-light btn-outline-
success mt-1 mb-2" id="filRes">RESET</button>
        </div>
    </div>
</div>

```

3.1.4.2.13. register.php

```

<div class="d-flex flex-column align-items-center justify-content-center mt-
5 text-center">
    <div id="logovanje" class="p-3">
        <h3>Register</h3>
        <form id="forma" action="models/register.php" method="POST" onsubmit="
return formCheck()">
            <div class="form-group mt-4">
                <label for="username">Username:</label><br>
                <input type="text" class="form-
control" id="username" name="username" placeholder="small letters or numbers,
4-20">
                <span class="text-center feedbackBad">Try again!</span>
            </div>
            <div class="form-group mt-2">
                <label for="email">Email :</label><br>
                <input type="text" class="form-
control" id="email" name="email" placeholder="ex. topshrimp@gmail.com">
                <span class="text-center feedbackBad">Try again!</span>
            </div>
        </form>
    </div>
</div>

```

```

        </div>
        <div class="form-group mt-2">
            <label for="password">Password:</label><br>
            <input type="password" class="form-
control" id="password" name="password" placeholder="no less than 8 characters!
">

            <span class="text-center feedbackBad">Try again!</span>
        </div>
        <div class="form-group mt-2">
            <label for="rePassword">Repeat Password:</label><br>
            <input type="password" class="form-
control" id="rePassword" name="rePassword">

            <span class="text-center feedbackBad">Try again!</span>
        </div>
        <div class="form-group mt-2">
            <button type="submit" class="btn btn-light btn-outline-
success" id="register" name="register">Register</button>
        </div>
    </form>
</div>

<?php
    if(isset($_GET['outcome'])):
        if($_GET['outcome']=='done'):
            ?>

            <div class="flex-column d-flex justify-content-center align-items-
center mt-2 short p-2">
                <div class="ftttx title text-center mb-
2"><h5>You have been registered.<br>
                    <span class="ftttx"><a href="index.php?page=login">Log in!</a>
</span></h5>
                </div>
            </div>
        </div>
    <?php
        endif;
    <?php
        endif;
    ?>
    <?php
        if(isset($_GET['outcome'])) {
            if($_GET['outcome']=='failed') {
                echo "Something went wrong, try again.";
            }
        }
    ?>
</div>

```

3.2. CSS

3.2.1. style.scss

```
4. @import url('https://fonts.googleapis.com/css?family=Raleway&display=swa
   p');
5. *{
6.     margin: 0px;
7.     padding: 0px;
8.     box-sizing: border-box;
9. }
10.html{
11.     scroll-behavior: smooth;
12.}
13.
14.$bckg: url('../img/planted1.jpg');
15.$divTransparent :rgba(0, 5, 0, 0.7);
16.
17.body{
18.     font-family: 'Raleway', sans-serif;
19.     background: #a0bc76;
20.     background-repeat: no-repeat;
21.     background-attachment: fixed;
22.     background-position: center;
23.     color: white;
24.     height: 100%;
25.}
26.a{
27.     color: white;
28.     &:hover{
29.         color: white;
30.     }
31.}
32.p{
33.     margin-bottom: 0px;
34.}
35.#footer{
36.     width: 100%;
37.     /*position: absolute;
38.     bottom:0px;
39.     right:0px;*/
40.}
41..fttxt{
42.     color: rgba(0, 5, 0, 0.7);
43.     font-weight: bold;
44. }
45..navbar{
46.     background-color: rgba(0, 5, 0, 0.7);
47.     color: white;
48.     .navbar-toggler-icon{
```

```
49.         color:white;
50.     }
51.}
52.#logovanje{
53.    min-height: 400px;
54.    width: 300px;
55.    background-color: $divTransparent;
56.    border-radius: 20px;
57.    #loginSbmt{
58.        border-bottom: 1px solid white;
59.    }
60.    #regClick{
61.        text-decoration: underline;
62.    }
63.}
64..postBckg{
65.    min-height: 400px;
66.    max-width: 600px;
67.    background-color: $divTransparent;
68.    border-radius: 2px 20px 20px 20px;
69.}
70..postBckgSmall{
71.    max-width: 600px;
72.    background-color: $divTransparent;
73.    border-radius: 20px 20px 20px 20px;
74.}
75..postSlika{
76.    height: 200px;
77.    border-radius: 20px;
78.}
79..postAu{
80.    background-color: $divTransparent;
81.    height: 110px;
82.    border-radius: 20px 20px 20px 20px;
83.
84.}
85..postIco{
86.    width: 50px;
87.    height: 50px;
88.    border-radius: 30px;
89.}
90..komentari{
91.    border-top: 1px solid white;
92.}
93..profilePersonal{
94.    background-color: $divTransparent;
95.    border-radius: 20px;
96.}
97..profileAuthor{
```

```
98.     height: 300px;
99.     background-color: $divTransparent;
100.         border-radius: 20px;
101.     }
102.     .slikaAuthor{
103.         border-radius: 20px;
104.         height: 250px;
105.     }
106.     .authUnder{
107.         text-decoration: underline;
108.     }
109.     .profilLeft{
110.         height: 300px;
111.     }
112.     .profilLeftAdmin{
113.         min-height: 500px;
114.     }
115.     ul{
116.         list-style: none;
117.     }
118.     .paginacija{
119.         height: 30px;
120.         width: 30px;
121.         background-color: $divTransparent;
122.         border-radius: 20px;
123.     }
124.     #hoverEdit{
125.         font-size: 14px;
126.         font-weight: bold;
127.     }
128.     #phChange{
129.         background-color: $divTransparent;
130.         border-radius: 20px;
131.         height: 200px;
132.     }
133.     .profilSlika{
134.         border-radius: 20px;
135.         max-height: 180px;
136.     }
137.     textarea{
138.         width: 90% !important;
139.     }
140.     .infoAcc{
141.         font-weight: 600;
142.     }
143.     .feedbackBad{
144.         display: none;
145.         background-color: red;
146.         margin-top: 5px;
```

```
147.         padding: 3px;
148.         border-radius: 4px;
149.         width: 100px;
150.     }
151.     #newTitle, #newCategory{
152.         width: 90% !important;
153.     }
```

3.3. JS

3.3.1. main.js

```
$(document).ready(function () {

    $('.paginationNum').click(posts);
    $('#filSub').click(filterPosts);
    $('#filRes').click(resetPosts);
})

function posts(e){
    e.preventDefault();
    let limit = ($(this).data('limit')-1)*10;
    $.ajax({
        url: "models/filter.php",
        method: "post",
        data:{
            limit: limit
        },
        dataType: "json",
        success: function(data){
            ispisiPost(data.posts);
            pagination(data.pg);
        },
        error:function(er){
            console.log(er);
        }
    })
}

function resetPosts(){
    let resetuj = 1;
    $.ajax({
        url: "models/filter.php",
        method: "post",
        data: {
            resetuj: resetuj
        },
        success: function(data){
            ispisiPost(data.posts);
            pagination(data.pg);
        },
    },
```



```

        error:function(er){
            console.log(er);
        }
    });
}
function filterPosts(){
    let sortiranje = $('#ddSort').val();
    let kategorija = $('#ddCategory').val();
    let tekst = $('#srchTxt').val();

    let limit = 0;
    console.log("sort: "+sortiranje+", kategorija: "+kategorija+", tekst: "+tekst)
    $.ajax({
        url: "models/filter.php",
        method: "post",
        data: {
            limit : limit,
            sort : sortiranje,
            cat : kategorija,
            txt : tekst
        },
        dataType: "json",
        success: function(data){
            ispisiPost(data.posts);
            pagination(data.pg);
        },
        error:function(er){
            console.log(er);
        }
    });
}
function pagination(num){
    let html="";
    for(let i=1;i<=num;i++){
        html+=`<li class="paginacija m-2"><a href="#" class="paginationNum" data-limit="${i}">${i}</a></li>`
    }
    $('#pagination').html(html);
    $('.paginationNum').click(posts);
}
function ispisiPost(data){
    console.log("ispisalo")
    let html="";
    data.forEach(e =>{
        let dat = e.created.split(' ');
        html += `<div class="d-flex flex-column flex-md-row justify-content-center m-5 text-center col-11">

```

```

        <div class="d-flex flex-column align-items-center justify-content-
center postAu col-2">
            
            <p>{e.username}</p>
        </div>
        <div class="postBckgSmall d-flex flex-column p-3 col-9">
            <div class="a">
                
            </div>
            <div class="postTitle mt-2">
                <a href="index.php?page=post&id={e.id}"><h2>{e.title}</h2></
a>

            </div>
            <div class="d-flex justify-content-end align-items-center">
                <i class="far fa-clock pr-2 pl-2"></i> {dat[0]}
            </div>
        </div>
    </div>`;
    })
    $('#sviPostovi').html(html);
}

function formCheck(){
    let user = document.getElementById("username").value;
    let email = document.getElementById("email").value;
    let pass = document.getElementById("password").value;
    let repPass = document.getElementById("rePassword").value;
    let reUser = /^[a-z0-9]{4,20}$/;
    let reEmail = /^[a-z][a-z\d\_\.\-]+\@[a-z\d]+(\.[a-z]{2,4})+$/;
    let rePass = /^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-
z\d@$!%*?&]{8,}$/;

    let greske= 0;

    if (!reUser.test(user)) {
        $("#username").next().css("display", "block");
        greske++;
    }
    else {
        $("#username").next().css("display", "none");
    }

    if (!reEmail.test(email)) {
        $("#email").next().css("display", "block");
        greske++;
    }
    else {
        $("#email").next().css("display", "none");
    }
}

```

```

    if (!rePass.test(pass)) {
        $("#password").next().css("display", "block");
        greske++;
    }
    else {
        $("#password").next().css("display", "none");
    }
    if (pass != repPass) {
        $("#rePassword").next().css("display", "block");
        greske++;
    }
    else {
        $("#rePassword").next().css("display", "none");
    }
    console.log(greske)
    if(greske>0){
        return false;
    }
    else{
        return true;
    }
}
function editCheck(){
    let user = document.getElementById("username").value;
    let email = document.getElementById("email").value;
    let pass = document.getElementById("password").value;
    //
    let reUser = /^[a-z0-9]{4,20}$/;
    let reEmail = /^[a-z][a-z\d\_\.\-]+\@[a-z\d](\.[a-z]{2,4})+$/;
    let rePass = /^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$/;
    let greske= 0;
    if (!reUser.test(user)) {
        $("#username").css("border", "2px solid red");
        greske++;
    }
    else{
        $("#username").css("border", "none");
    }
    if (!reEmail.test(email)) {
        $("#email").css("border", "2px solid red");
        greske++;
    }
    else{
        $("#email").css("border", "none");
    }
    if (!rePass.test(pass)) {
        $("#password").css("border", "2px solid red");

```

```

        greske++;
    }
    else{
        $("#password").css("border", "none");
    }
    console.log(greske)
    if(greske>0){
        return false;
    }
    else{
        return true;
    }
}
function editContact(){
    let name = document.getElementById("name").value;
    let reName = /^[A-Z][a-z]{2,9}(\s[A-Z][a-z]{2,14})+$/;
    let email = document.getElementById("email").value;
    let reEmail = /^[a-z][a-z\d\_\.\-]+\@[a-z\d]+(\.[a-z]{2,4})+$/;
    let subject = document.getElementById("subject").value;
    let reSubj = /^[A-zšđžčćŠĐŽČĆ 1-9]{10,150}/;
    let msg = document.getElementById("text").value;
    let reMsg = /^[A-zšđžčćŠĐŽČĆ 1-9]{10,300}/;
    //
    let greske= 0;
    if (!reName.test(name)) {
        $("#name").css("border", "2px solid red");
        greske++;
    }
    else{
        $("#name").css("border", "none");
    }
    if (!reEmail.test(email)) {
        $("#email").css("border", "2px solid red");
        greske++;
    }
    else{
        $("#email").css("border", "none");
    }
    if (!reSubj.test(subject)) {
        $("#subject").css("border", "2px solid red");
        greske++;
    }
    else{
        $("#subject").css("border", "none");
    }
    if (!reMsg.test(msg)) {
        $("#text").css("border", "2px solid red");
        greske++;
    }
}

```

```

    else{
        $("#text").css("border", "none");
    }
    console.log(greske)
    if(greske>0){
        return false;
    }
    else{
        return true;
    }
}

```

3.4. DB

-- phpMyAdmin SQL Dump

-- version 5.0.1

-- <https://www.phpmyadmin.net/>

--

-- Host: 127.0.0.1

-- Generation Time: Jun 09, 2021 at 02:51 PM

-- Server version: 10.4.11-MariaDB

-- PHP Version: 7.4.2

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

SET AUTOCOMMIT = 0;

START TRANSACTION;

SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8mb4 */;

--

-- Database: `ablog`

--

-- -----

--

-- Table structure for table `categories`

--

```
CREATE TABLE `categories` (  
  `cat_id` int(11) NOT NULL,  
  `name` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `categories`

--

```
INSERT INTO `categories` (`cat_id`, `name`) VALUES  
(1, 'Shrimp'),  
(2, 'Snails'),  
(3, 'Fish'),  
(4, 'Plants');
```

-- -----

--

-- Table structure for table `posts`

--

```
CREATE TABLE `posts` (  
  `post_id` int(255) NOT NULL,
```

```
`title` varchar(255) NOT NULL,  
`text` text NOT NULL,  
`user_id` int(255) NOT NULL,  
`cat_id` int(255) NOT NULL,  
`created_at` datetime NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `posts`

--

```
INSERT INTO `posts` (`post_id`, `title`, `text`, `user_id`, `cat_id`, `created_at`) VALUES
```

(9, 'Procambarus Alleni (Plavi rak)', 'Ovi prelepi rakovi dolaze sa Floride, najviše iz močvare Everglejds. Greškom ih nazivaju plavim jastozima, što je pogrešno, jer su jastozi morski rakovi. Mogu narasti i do 15cm, i razlike između polova postoje. Mužjak ima dva dodatna pipka iza zadnjih nogu koje ženka nema, takođe imaju i izduženija klešta nego ženke. U akvarijumu umeju biti agresivni prema drugim stanovnicima, i preporučuje se da u akvarijumu budu sami. Jedu sve čega mogu da se dočepaju, ali masnu hranu izbegavaju. Jedu biljke, hranu koja padne na dno, male ribe i sporije kozice. Rado će jesti i komade šargarepe ili tikvice. Ne treba ih hraniti više od jednom dnevno. Temperatura koja im odgovara se kreće od 18 do 24 stepena, i ne treba ih držati u akvarijumu manjem od 60l. On bi trebao biti ubiljen, sa dosta mesta za skrivanje i pokriven, jer umeju izaći iz njega.', 1, 1, '2021-06-06 20:42:56'),

(13, 'Paracheirodu Axelrodi (Kraljevska neonka)', 'Kraljevska neonka je verovatno najzastupljenija ribica u akvarijumima kako domaćih, tako i svetskih akvarista. Popularnost ove male jatne ribice ne jenjava skoro čitav vek, dok se interesovanje za većinu drugih vrsta vremenom menja. Ovu vrstu je 1956 godine detaljno opisao poznati ihtiolog Leonard Peter Šulc i nazvao je u čast drugog poznatog ihtiologa i izdavača Herberta R. Axelroda (TFH-tropical fish hobbyist, poznati Axelrodov Atlas). Često se naziva i Kardinal neonka ili Crvena neonka. Ove male i jatne ribe naseljavaju slivove reka Rio Negro i Orinoko, manje populacije nalaze se i u drugim rekama. Kraljevsku neonku nalazimo u malim pritokama, u plitkim potocima i plavljenim područjima sa sporim tokom vode. Takve vode su praktično bez karbonata, sa malo mineralnih materija i visokom koncentracijom huminskih kiselina koje boje vodu. Baš zbog života u ovim tamnim vodama kraljevska neonka ima reflektivnu (iridescentnu) plavu liniju.', 4, 3, '2021-06-01 15:50:21'),

(14, 'Tylomelania (Sulawesi Rabbit Snail)', 'Tylomelania je rod slatkovodnih puževa poreklom iz Sulavezija (nekadšnji Celebes). Na ovom ostrvu neobičnog oblika postoji nekoliko jezera koja su karakteristična po svojoj mekoj i baznoj vodi. Veliki broj endemskih vrsta kozica, puževa i riba se poslednjih godina uključuju u akvaristiku. Mnogi od ovih organizama nisu čak ni klasifikovani od strane naučnika, već u prodaju dolaze sa različitim nazivima, imenom lokaliteta i sl. Puževi roda Tylomelania su veoma interesantni akvaristima, za razliku od drugih egzotičnih puževa (Brotia

sp.npr.) veoma se dobro prilagođavaju životu u akvarijumu, nisu zahtevni i redovno se razmnožavaju.', 4, 2, '2021-06-02 15:52:33'),

(16, 'Atyopsis Moluccensis (Bamboo Shrimp)', 'Filter šrimp je svakako jedan od najinteresantnijih organizama u slatkovodnoj akvaristici. Ovaj prilično krupan šrimp veoma je miroljubiv pa se može lako kombinovati sa mnogim drugim stanovnicima. U prirodi su dosta rasprostranjeni i naseljavaju ostrva sve od Šri Lanke do Okinave, kao i priobalne vode na kontinentu. Vode u kojima se mogu naći dosta variraju po svojim parametrima, tako da je filter šrimp veoma prilagodljiva vrsta, podjednako se uspešno mogu čuvati u mekoj i tvrdoj vodi sa različitim pH vrednostima. ', 5, 1, '2021-06-03 16:00:55'),

(17, 'Neocardina Davidi Var. Blue Diamond', 'Jedne od najpopularnijih vrsta kozica u svetu koje ne samo da su lepe, nego su i odlični čistači akvarijuma. Dolaze nam sa Tajvana. Veoma su aktivne i brzo se razmnožavaju. Spadaju u istu porodicu kao i red cherry kozice, tako da im odgovaraju isti uslovi. lako se mogu držati i u malim akvarijumima (već od 10l), mnogi ljudi ih koriste i u velikim, jer skupljaju otpad i utiču na rasprostranjenost algi. Imaju prelepu plavu boju, potpuno su bezopasni za sve stanovnike akvarijuma, dobri su čistači i brinu o biljkama. ', 5, 1, '2021-06-07 16:01:42'),

(18, 'Betta Splendens (Borac)', 'Poreklom iz porodice Anabantidae, ove ribice svoje prirodno stanište imaju na Tajlandu. Dužine su oko 7 santimetara, a minimalna dužina akvarijuma u kojima mogu da obitavaju ne sme da bude ispod 50cm. Dakle, nije preporučljivo da budu smešteni u minimalne akvarijume, što se neretko dešava. Ipak, atraktivnost njihovog izgleda ne znači da im ne treba dati što veći životni prostor. Mužjaci su veoma agresivni jedni prema drugima i njihove borbe se ponekad završavaju čak i sa smrtnim ishodom.', 7, 3, '2021-06-07 23:36:29'),

(19, 'Epalzeorhynchus Frenatus (Frenatus)', 'Frenatus je još jedna vrsta koja svoju popularnost duguje izuzetno lepom izgledu i neobaveštenosti akvarista. Kao i druge vrste iz roda Epalzeorhynchus, Bikolor i Lisica (Kalopterus), odrasli frenatusi ponekad postaju agresivni i teritorijalni. Ova agresivnost je kod frenatusa nešto manje izražena i oni su pogodniji za zajednički akvarijum od bikolora. Još uvek je potrebno obratiti pažnju na izbor ostalih riba. Dugo godina je naziv roda bio Labeo, tek pre nekoliko godina je promenjen u Epalzeorhynchus.', 7, 3, '2021-06-07 23:39:13'),

(20, 'Clithon Corona (Rogata nerita)', 'Rogata nerita, Clithon Corona raste do 1,5 cm i najmanja je nerita koju držimo u akvarijumima. Poreklom je iz jugoistočne Azije i živi u priobalnom pojasu Pacifika. Najčešće se sreću primerci kod kojih se smenjuju crne i žute pruge. Druge kombinacije boja su moguće i ovi puževi, pored toga što su vrhunski algari, odlikuju se neobičnom lepotom. Rogovi na ovim puževima mogu biti veoma izraženi, veličina im varira od primerka do primerka. Kod hvatanja treba obratiti pažnju da se akvarista ne ubode na puža.', 7, 2, '2021-06-07 23:39:58'),

(21, 'Mikrogeophagus Ramirezi Electric Blue', 'Mikrogeophagus ramirezi (Ramirez), sa razlogom je jedna od najomiljenijih vrsta ribica u akvarijumima domaćih akvarista koji su prihvatili drugi deo latinskog naziva kao domaći naziv za ovu ribu. U stranoj literaturi naziva se German Blue Ram (koloritni soj razvijen je u Nemačkoj), Butterfly cichlid, Dwarf butterfly cichlid ili jednostavno Ram. U prirodi žive u mirnim vodama pritoka reke Orinoko, koje protiču kroz delove Venecuele i Kolumbije. Tipična su „blackwater“ vrsta, žive u sporim toplim vodama sa malom količinom rastvorenih minerala i dosta tanina. Primerci odgajeni u akvarijumima bolje tolerišu tvrdu vodu, ali se i dalje za držanje ovih riba preporučuje meka voda tretirana tresetom, Sera Super Peat, kako bi se dobila blago kisela (pH 5.2-6.7) voda u kojoj su Ramirez najvitalniji i imaju najlepše boje.', 8, 3, '2021-06-07 23:48:00'),

(22, 'Desmopuntius Pentazona', 'Jatna je vrsta koja najviše dolazi do izražaja u biljnom akvarijumu, treba obezbediti i par tamnijih uglova gde će se ove ribe sakrivati, kao i deo slobodnog prostora gde

će formirati jato. Najbolje ih je naseljavati u jatu od minimum 5-6 riba, zbog male veličine pogodne su i za manje akvarijume. Upotrebom dekorativnog drveta i kamena imitiraju se prirodni uslovi, pa se ribice dobro osećaju i pokazuju svu svoju lepotu. Nezahtevne su po pitanju ishrane, i prihvataju svu suhu hranu. Mrest se odvija slično kao i kod ostalih Barbusa, do mresta dolazi u jutarnjim časovima, ikru polažu slobodno među kamenjem i vegetacijom, roditeljske brige nema i ikru često proždiru. Mlađ se u početku hrani infuzorijom.', 8, 3, '2021-06-07 23:48:53'),

(23, 'Cardina Multidendata (Amano Kozica)', 'Vrsta kozice poreklom iz Japana, Tajvana i Koreje. Naziv je dobila po pokojnom japanskom akvaristi Takaši Amanu. Naziva se još i japanska kozica. Odlikuje je providno telo sa crvenkastom linijom i tačkicama sa strane. Treba ih držati u grupama od bar 5 komada, u akvarijumu ne manjem od 30l. Vole biljne akvarijume, što zbog skloništa, to zbog hrane. Amano kozice su odlični algari. Ne treba ih držati sa agresivnijim ribama poput barbusa i ciklida, jer te ribe kozice gledaju kao hranu. Minije vrste tetri, koridorasi, mali algari i druge kozice se lepo slažu sa Amanima, koji su miroljubivi.', 9, 1, '2021-06-07 23:54:36'),

(24, 'Neritina Natalensis Sp. Zebra (Zebrasta nerita)', 'Jedan od najčešćih i najpopularnijih puževa u akvaristici. Ime su dobile po svom specifično obojenom oklopu. Narastu do 2cm, ne više, tako da su pogodni i za manje akvarijume. Opasnost od preteranog razmnožavanja ne postoji, jer im za to treba poluslana voda. Postoje podaci da je razmnožavanje uspelo i u slatkoj vodi, ali oni nisu potvrđeni. Glavna odlika Nerita je to što se hrane svim vrstama algi, i na taj način akvarijum održavaju čistim. Druga odlika je, naravno, njihov izgled. Pored algi, rado prihvataju tablete spiruline, kai i bareno povrće. Kao i kod svih drugih nerita, sklone su ispadanju iz akvarijuma, pogotovo ako im parametri vode ne odgovaraju, pa se preporučuje da akvarijum u kom se drže ima poklopac.', 9, 2, '2021-06-07 23:55:20'),

(25, 'Bucephalandra deep purple', 'Bucephalandra deep purple je jedna od najatraktivnijih biljaka iz ove porodice. Odlikuje je plavkasto ljubicasta boja lista. Veoma sporo raste i koristi se za postizanje jaceg kontrasta u biljnom akvarijumu upravo zbog svoje neobicne boje. Prija joj umereno do jace svetlo sa CO2 dodavanjem. Umereno 6-14 mg/L. U slucaju jaceg osvetljenja 15-25 mg/L. Mesto sađenja: Prednji plan, na kamen ili panj.', 10, 4, '2021-06-08 00:03:40'),

(26, 'Ludwigia Repens "Rubin"', 'Ludwigia repens "Rubin" potiče iz Severne Amerike i najprepoznatljivija je po svojim bordo listovima i stabljikom koja ume da razvije visinu od 20-50 cm i širinu od 4-6cm. Zbog svoje specifične boje idealna je za stvaranje šarenolike slike unutar akvarijuma i obično se preporučuje da u akvarijum bude posađena u vidu grupa. Najbolje je ako se nalazi pod svetlom jačeg intenziteta (čak ispod samog izvora svetlosti) kako bi njen žbunasti oblik zadržao svoju kompaktnost, a jačina njene boje dala što bolji kontrast zelenim tonovima koji mahom prevlađuju akvarijumom.', 10, 4, '2021-06-08 00:06:09'),

(27, 'Anubias Barteri Var Nana', 'Anubias barteri var nana, poreklom iz Kameruna, je mala, atraktivna biljka koja uspeva u svim uslovima. Visina koju dostiže iznosi između 5-10cm, sa rizomom od 10-15cm (a ponekad i više). Prilikom kupovine naročito je važno proveriti rizom ove biljke – ukoliko on nije u idealnom stanju, biljka posle izvesnog vremena može početi da truli. Anubias barteri var. Nana spada u red spororastućih biljaka, često cveta pod vodom (žuti cvet sa belim laticama koji svojim oblikom podseća na klip kukuruza), a listovi joj traju i po nekoliko godina. Najbolje rezultate daje ukoliko je vežete za kamen, panj ili na koren nekog stabla unutar akvarijuma.', 10, 4, '2021-06-08 00:10:00'),

(28, 'Cryptocoryne Parva', 'Cryptocoryne parva vodi poreklo iz Sri Lanke, i jedna je od manjih kripti. Takpodje, jedna je od retkih biljaka kojoj dodatna prihrana i CO2 ne uticu na rast i zgled lista. Zbog svoje visine trazi vise svetla i savetuje se da se sadi jedna pored druge kako bi se postgao gust zibun zelenila u prednjem planu akvarijuma.', 5, 4, '2021-06-08 01:39:46');

--

-- Table structure for table `post_comments`

--

```
CREATE TABLE `post_comments` (  
  `id` int(11) NOT NULL,  
  `text` text NOT NULL,  
  `created_at` datetime NOT NULL DEFAULT current_timestamp(),  
  `post_id` int(255) NOT NULL,  
  `user_id` int(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `post_comments`

--

```
INSERT INTO `post_comments` (`id`, `text`, `created_at`, `post_id`, `user_id`) VALUES  
(1, 'Pretty schweet', '2021-06-06 21:15:33', 9, 1),  
(2, 'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut  
labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris  
nisi ut aliquip ex ea commodo consequat.', '2021-06-07 15:48:37', 9, 4),  
(5, 'Promena komentara test', '2021-06-07 16:20:48', 9, 5),  
(6, 'Yepypypypyp', '2021-06-07 23:50:14', 13, 8),  
(7, 'Super interesting ', '2021-06-07 23:51:09', 14, 8),  
(8, 'yes quite', '2021-06-07 23:51:33', 19, 8),  
(9, 'Absolutely', '2021-06-07 23:51:55', 13, 7),  
(10, 'random komentar', '2021-06-07 23:52:21', 16, 7),  
(11, 'schweet', '2021-06-07 23:52:42', 21, 7),  
(12, 'test komentar test test', '2021-06-08 00:18:29', 18, 10),
```

(13, 'jos jedan komentar', '2021-06-08 00:18:41', 19, 10),
(14, 'super super super', '2021-06-08 00:19:13', 24, 10),
(15, 'most excellent !', '2021-06-08 00:19:33', 23, 10),
(16, 'Kkoommeennttaarr', '2021-06-08 00:20:07', 22, 10),
(17, 'yep yep jos jedan komentar', '2021-06-08 00:20:46', 21, 10),
(18, 'testtesttesttest', '2021-06-08 00:22:40', 27, 8),
(19, 'super spikey ', '2021-06-08 00:23:01', 20, 8),
(20, 'yeeyee', '2021-06-08 00:23:36', 23, 8),
(21, 'At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident', '2021-06-08 00:24:21', 24, 8),
(22, 'At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident', '2021-06-08 00:24:43', 9, 8),
(23, 'At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident', '2021-06-08 00:25:05', 26, 8);

-- -----

--

-- Table structure for table `post_img`

--

```
CREATE TABLE `post_img` (  
  `id` int(11) NOT NULL,  
  `src` varchar(255) NOT NULL,  
  `alt` varchar(100) NOT NULL,  
  `post_id` int(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `post_img`

--

```
INSERT INTO `post_img` (`id`, `src`, `alt`, `post_id`) VALUES
(3, 'assets/img/posts/post_11623018091.jpeg', 'admin1623018091', 9),
(7, 'assets/img/posts/post_41623073821.jpeg', 'waterman1623073821', 13),
(8, 'assets/img/posts/post_41623073953.jpeg', 'waterman1623073953', 14),
(10, 'assets/img/posts/post_51623074455.jpeg', 'supremeshrimp1623074455', 16),
(11, 'assets/img/posts/post_51623074502.jpeg', 'supremeshrimp1623074502', 17),
(12, 'assets/img/posts/post_71623101789.jpeg', 'pogacica1623101789', 18),
(13, 'assets/img/posts/post_71623101953.jpeg', 'pogacica1623101953', 19),
(14, 'assets/img/posts/post_71623101998.jpeg', 'pogacica1623101998', 20),
(15, 'assets/img/posts/post_81623102480.jpeg', 'jagoda1623102480', 21),
(16, 'assets/img/posts/post_81623102533.jpeg', 'jagoda1623102533', 22),
(17, 'assets/img/posts/post_91623102876.jpeg', 'magikarp1623102876', 23),
(18, 'assets/img/posts/post_91623102920.jpeg', 'magikarp1623102920', 24),
(19, 'assets/img/posts/post_101623103420.jpeg', 'plantman1623103420', 25),
(20, 'assets/img/posts/post_101623103569.jpeg', 'plantman1623103570', 26),
(21, 'assets/img/posts/post_101623103800.jpeg', 'plantman1623103800', 27),
(22, 'assets/img/posts/post_51623109186.jpeg', 'supremeshrimp1623109186', 28);
```

--

-- Table structure for table `roles`

--

```
CREATE TABLE `roles` (
  `role_id` int(255) NOT NULL,
  `name` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```

--

-- Dumping data for table `roles`

--

INSERT INTO `roles` (`role_id`, `name`) VALUES
(1, 'user'),
(2, 'admin');

-----

--

-- Table structure for table `users`

--

CREATE TABLE `users` (
  `user_id` int(255) NOT NULL,
  `username` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `created_at` datetime NOT NULL DEFAULT current_timestamp(),
  `role_id` int(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--

-- Dumping data for table `users`

--

INSERT INTO `users` (`user_id`, `username`, `email`, `password`, `created_at`, `role_id`) VALUES
(1, 'admin', 'admin@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-03 22:49:12', 2),
(2, 'dudebro1', 'dudebro@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-03
22:54:06', 1),

```

```
(4, 'waterman', 'waterman@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07
15:46:05', 1),

(5, 'supremeshrimp', 'supreme@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07
15:56:15', 1),

(6, 'jellyfish', 'jelly@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07 19:24:55', 1),

(7, 'pogacica', 'pogacica@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07 23:35:15',
1),

(8, 'jagoda', 'jagoda@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07 23:45:28', 1),

(9, 'magikarp', 'magikarp@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-07
23:53:25', 1),

(10, 'plantman', 'plantman@gmail.com', '2ad043736f2c4a12c7f6480e8449e38a', '2021-06-08
00:01:14', 1);
```

--

-- Table structure for table `user_img`

--

```
CREATE TABLE `user_img` (
  `id` int(255) NOT NULL,
  `src` varchar(255) NOT NULL,
  `src_og` varchar(255) NOT NULL,
  `alt` varchar(255) NOT NULL,
  `user_id` int(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `user_img`

--

```
INSERT INTO `user_img` (`id`, `src`, `src_og`, `alt`, `user_id`) VALUES
```

```
(1, 'assets/img/account/small_profile_1623200612.jpeg',  
'assets/img/account/og_profile_1623200612.jpeg', 'admin20210609', 1),  
  
(4, 'assets/img/account/small_profile_1623087341.png',  
'assets/img/account/og_profile_1623087341.png', 'waterman20210607', 4),  
  
(5, 'assets/img/account/small_profile_1623087309.jpeg',  
'assets/img/account/og_profile_1623087309.jpeg', 'supremeshrimp20210607', 5),  
  
(6, 'assets/img/account/small_profile_1623087142.png',  
'assets/img/account/og_profile_1623087142.png', 'jellyfish20210607', 6),  
  
(7, 'assets/img/account/small_profile_1623101737.png',  
'assets/img/account/og_profile_1623101737.png', 'pogacica20210607', 7),  
  
(8, 'assets/img/account/small_profile_1623104537.jpeg',  
'assets/img/account/og_profile_1623104537.jpeg', 'jagoda20210608', 8),  
  
(9, 'assets/img/account/small_profile_1623102824.png',  
'assets/img/account/og_profile_1623102824.png', 'magikarp20210607', 9),  
  
(10, 'assets/img/account/small_profile_1623103710.jpeg',  
'assets/img/account/og_profile_1623103710.jpeg', 'plantman20210608', 10),  
  
(11, 'assets/img/account/small_profile_1623201156.png',  
'assets/img/account/og_profile_1623201156.png', 'dudebro120210609', 2);
```

--

-- Indexes for dumped tables

--

--

-- Indexes for table `categories`

--

ALTER TABLE `categories`

ADD PRIMARY KEY (`cat_id`);

--

-- Indexes for table `posts`

--

ALTER TABLE `posts`

ADD PRIMARY KEY (`post_id`),

```
ADD KEY `user_id` (`user_id`),
```

```
ADD KEY `cat_id` (`cat_id`);
```

```
--
```

```
-- Indexes for table `post_comments`
```

```
--
```

```
ALTER TABLE `post_comments`
```

```
ADD PRIMARY KEY (`id`),
```

```
ADD KEY `post_id` (`post_id`),
```

```
ADD KEY `user_id` (`user_id`);
```

```
--
```

```
-- Indexes for table `post_img`
```

```
--
```

```
ALTER TABLE `post_img`
```

```
ADD PRIMARY KEY (`id`),
```

```
ADD UNIQUE KEY `post_id` (`post_id`);
```

```
--
```

```
-- Indexes for table `roles`
```

```
--
```

```
ALTER TABLE `roles`
```

```
ADD PRIMARY KEY (`role_id`);
```

```
--
```

```
-- Indexes for table `users`
```

```
--
```

```
ALTER TABLE `users`
```

```
ADD PRIMARY KEY (`user_id`),
```

```
ADD KEY `role_id` (`role_id`);
```



```
--  
  
-- Indexes for table `user_img`  
  
--  
  
ALTER TABLE `user_img`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `user_id` (`user_id`);  
  
--  
  
-- AUTO_INCREMENT for dumped tables  
  
--  
  
--  
  
-- AUTO_INCREMENT for table `categories`  
  
--  
  
ALTER TABLE `categories`  
  MODIFY `cat_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;  
  
--  
  
-- AUTO_INCREMENT for table `posts`  
  
--  
  
ALTER TABLE `posts`  
  MODIFY `post_id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=29;  
  
--  
  
-- AUTO_INCREMENT for table `post_comments`  
  
--  
  
ALTER TABLE `post_comments`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=24;  
  
--  
  
-- AUTO_INCREMENT for table `post_img`
```

```
--  
  
ALTER TABLE `post_img`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=23;  
  
--  
  
-- AUTO_INCREMENT for table `roles`  
  
--  
  
ALTER TABLE `roles`  
  MODIFY `role_id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;  
  
--  
  
-- AUTO_INCREMENT for table `users`  
  
--  
  
ALTER TABLE `users`  
  MODIFY `user_id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;  
  
--  
  
-- AUTO_INCREMENT for table `user_img`  
  
--  
  
ALTER TABLE `user_img`  
  MODIFY `id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=12;  
  
--  
  
-- Constraints for dumped tables  
  
--  
  
--  
  
-- Constraints for table `posts`  
  
--  
  
ALTER TABLE `posts`
```

```

    ADD CONSTRAINT `posts_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`) ON
DELETE CASCADE ON UPDATE CASCADE,

    ADD CONSTRAINT `posts_ibfk_2` FOREIGN KEY (`cat_id`) REFERENCES `categories` (`cat_id`) ON
DELETE CASCADE ON UPDATE CASCADE;

--

-- Constraints for table `post_comments`

--

ALTER TABLE `post_comments`

    ADD CONSTRAINT `post_comments_ibfk_1` FOREIGN KEY (`post_id`) REFERENCES `posts` (`post_id`)
ON DELETE CASCADE ON UPDATE CASCADE,

    ADD CONSTRAINT `post_comments_ibfk_2` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`)
ON DELETE CASCADE ON UPDATE CASCADE;

--

-- Constraints for table `post_img`

--

ALTER TABLE `post_img`

    ADD CONSTRAINT `post_img_ibfk_1` FOREIGN KEY (`post_id`) REFERENCES `posts` (`post_id`) ON
DELETE CASCADE ON UPDATE CASCADE;

--

-- Constraints for table `users`

--

ALTER TABLE `users`

    ADD CONSTRAINT `users_ibfk_1` FOREIGN KEY (`role_id`) REFERENCES `roles` (`role_id`) ON DELETE
NO ACTION ON UPDATE NO ACTION;

--

-- Constraints for table `user_img`

--

ALTER TABLE `user_img`

    ADD CONSTRAINT `user_img_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`) ON
DELETE CASCADE ON UPDATE CASCADE;

```

COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;