Visoka škola strukovnih studija za informacione i komunikacione tehnologije

The Whisper’s Cove

http://thewhisperscove.epizy.com/

Miloš Parezanović 110/17

Smer: Internet tehnologije

Predmet: Praktikum iz web programiranja PHP

Contents

Uvod 4

Za izradu sajta korišćene su sledeće tehnologije 4

Opis funkcionalnosti 4

Template 4

Organizacija po folderima 5

Šema baze 6

Mapa sajta 6

Sitemap 6

Opis stranica sajta 7

Home 7

Forum 8

login 9

register 10

about 11

admin 13

user 14

post 15

contact 16

Kodovi 17

PHP 17

index.php 17

header.php 18

menu.php 19

footer.php 20

home.php 21

forum.php 24

about.php 26

admin.php 27

contact.php 28

post.php 29

user.php 30

login.php 31

logout.php 33

register.php 34
page-stats.php 36
add-categories.php 38
delete-categories.php 39
get-categories-opt.php 39
contact_validation.php 39
functions.php 40
general.php 42
get-links.php 42
register_activation.php 42
users.php 43
add-post.php 47
delete-post.php 48
get-post.php 48
get-posts.php 49
get-user-posts.php 49
post-pagination.php 50
sort-posts.php 50
update-post.php 51
add-user.php 53
delete-user.php 55
get-roles.php 55
get-user.php 56
get-users.php 56
update-user.php 57
config.php 58
connect.php 59
init.php 60
JavaScript 60
main.js 60
CSS 78
style.css 78

Uvod

Za izradu sajta korišćene su sledeće tehnologije

- PHP, MySQL
- HTML 5, CSS 3
- JavaScript, jQuery, Ajax

Opis funkcionalnosti

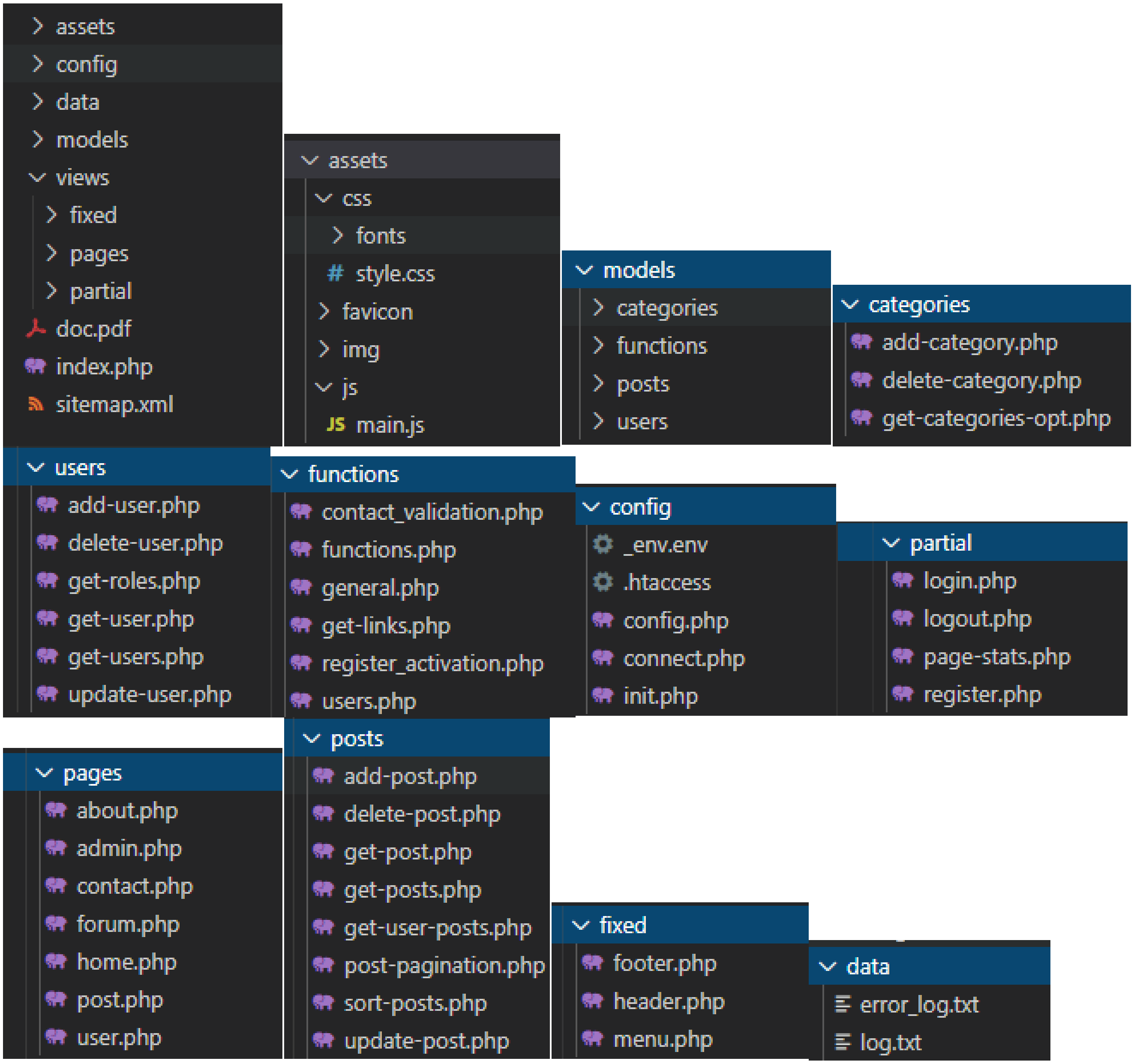
Sajt poseduje sledeće funkcionalnosti:

- Dohvatanje menija iz baze i njegovo ispisivanje na svim stranicam
- Dohvatanje postova i kategorija i njihovo ispisivanje na forum stranici, uz paginaciju i mogućnost filtriranja postova po kategoriji kao i sortitanje
- Provera unosa podataka prilikom prijavljivanja ili registracije na sajt pomocu regularnih izraza, provera sa klijentske i serverske strane
- Upravljanje dinamickog dela sajta sa admin strane
- Mogucnost postovanja sadržaja na sajt kao user korisnik
- Klikom na continue reading user moze da procita ceo sadržaj posta na zasebnoj strane

Template

Pravljen rucno nije korišćen gotov template

Organizacija po folderima



```
> assets
> config
> data
> models
v views
  > fixed
  > pages
  > partial
doc.pdf
index.php
sitemap.xml
```

```
v assets
  v css
    > fonts
    # style.css
  > favicon
  > img
  v js
    JS main.js
```

```
v models
  > categories
  > functions
  > posts
  > users
```

```
v categories
  add-category.php
  delete-category.php
  get-categories-opt.php
```

```
v users
  add-user.php
  delete-user.php
  get-roles.php
  get-user.php
  get-users.php
  update-user.php
```

```
v functions
  contact_validation.php
  functions.php
  general.php
  get-links.php
  register_activation.php
  users.php
```

```
v config
  _env.env
  .htaccess
  config.php
  connect.php
  init.php
```

```
v partial
  login.php
  logout.php
  page-stats.php
  register.php
```

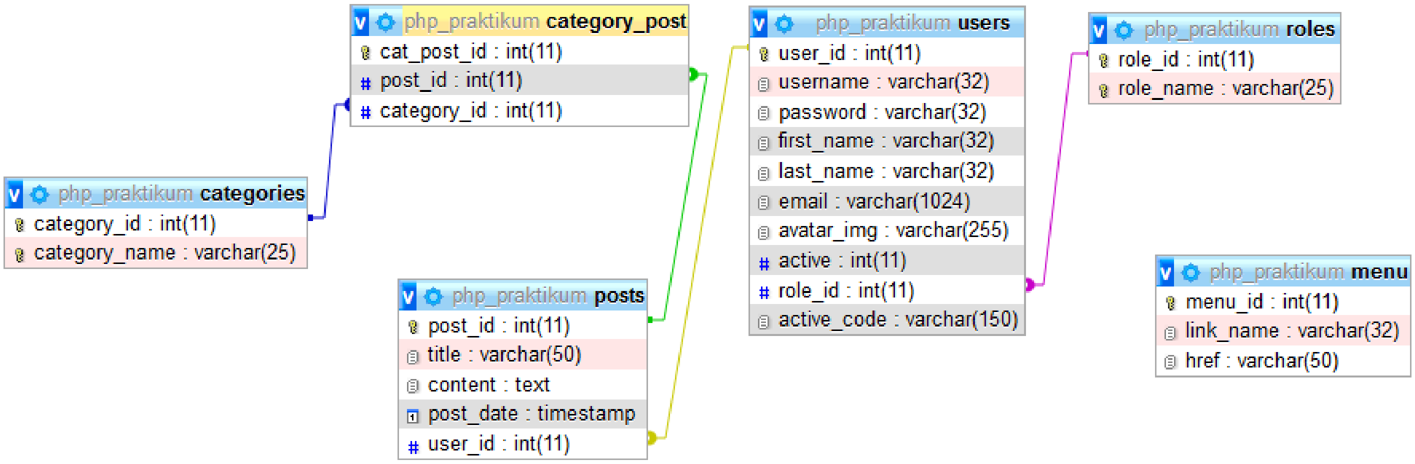
```
v pages
  about.php
  admin.php
  contact.php
  forum.php
  home.php
  post.php
  user.php
```

```
v posts
  add-post.php
  delete-post.php
  get-post.php
  get-posts.php
  get-user-posts.php
  post-pagination.php
  sort-posts.php
  update-post.php
```

```
v fixed
  footer.php
  header.php
  menu.php
```

```
v data
  error_log.txt
  log.txt
```

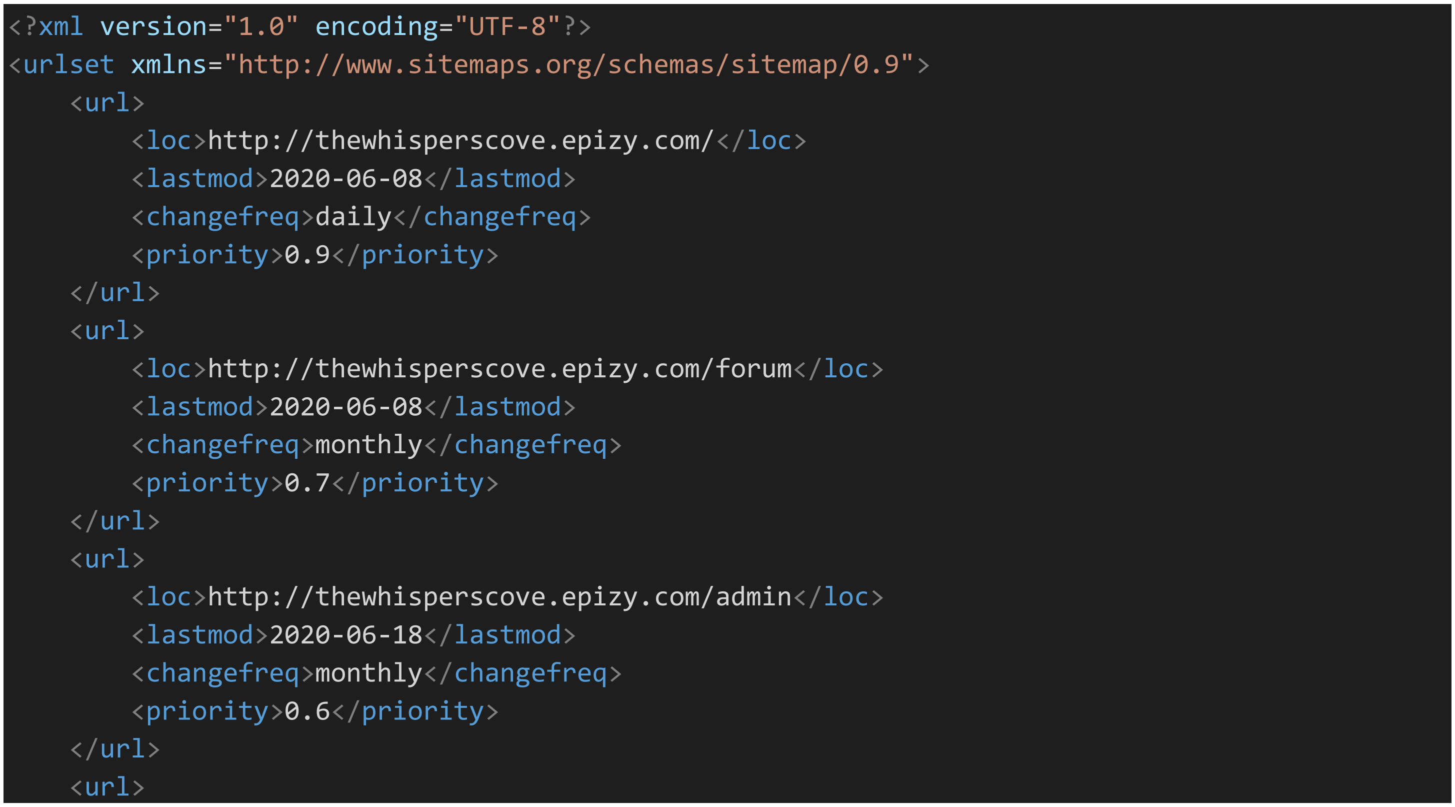
Šema baze



Mapa sajta

Napomena, sve stranice se nalaze na index.php strani a putem url adrese se ucitava strana po prosledjenim parametrom, stranice admin i user su zastićene i vrsi se redirekcija za ne autorizovane korisnike, kao i sve stranice za unos i izmenu postova ili korisnika.

Sitemap



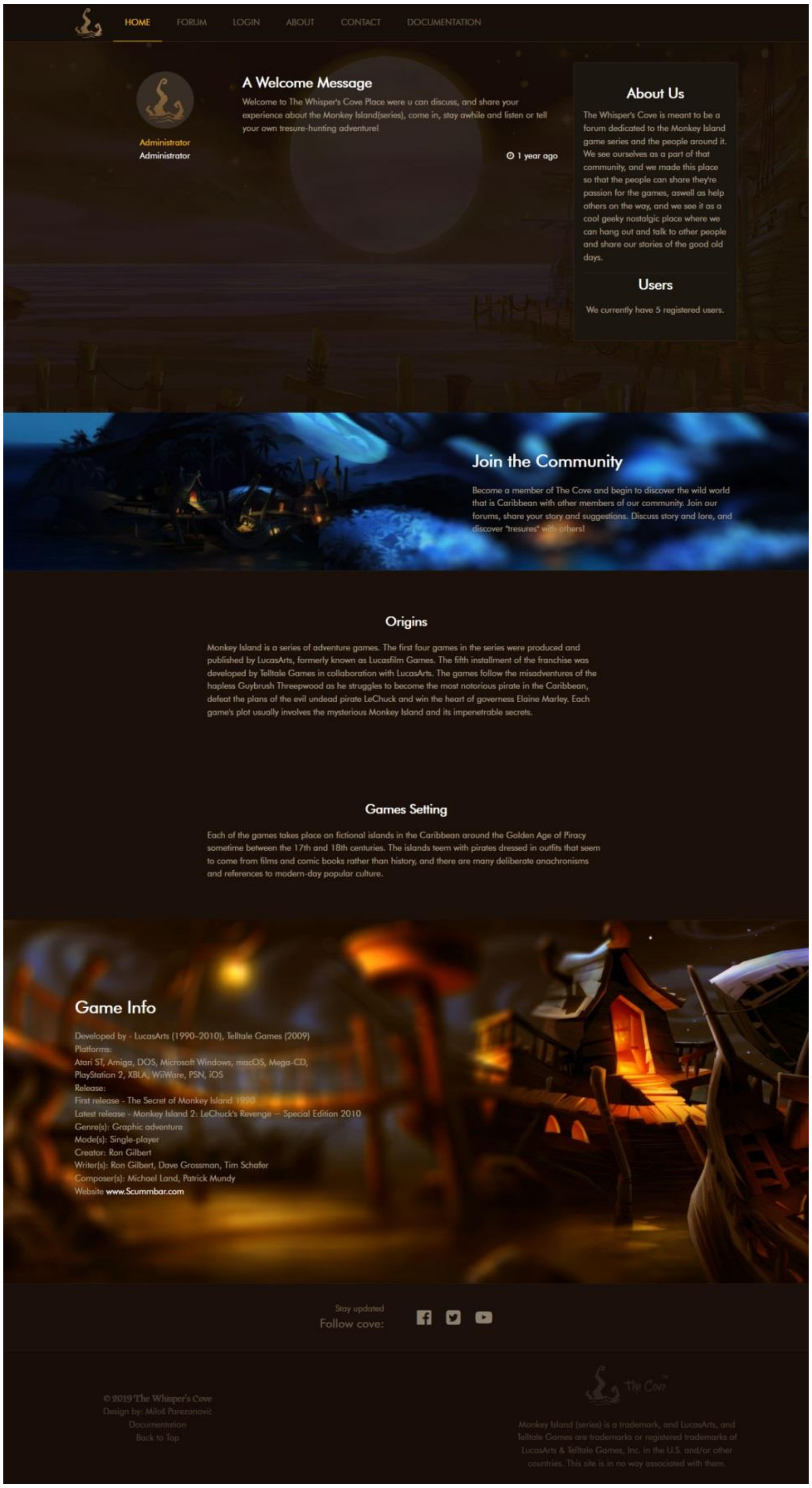
```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
    <url>
        <loc>http://thewhisperscove.epizy.com/</loc>
        <lastmod>2020-06-08</lastmod>
        <changefreq>daily</changefreq>
        <priority>0.9</priority>
    </url>
    <url>
        <loc>http://thewhisperscove.epizy.com/forum</loc>
        <lastmod>2020-06-08</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.7</priority>
    </url>
    <url>
        <loc>http://thewhisperscove.epizy.com/admin</loc>
        <lastmod>2020-06-18</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.6</priority>
    </url>
    <url>
```

```xml
        <loc>http://thewhisperscove.epizy.com/user</loc>
        <lastmod>2020-06-18</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
    <url>
        <loc>http://thewhisperscove.epizy.com/contact</loc>
        <lastmod>2020-06-18</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
</urlset>
```

Opis stranica sajta

Home

Index strana sadrži generalni opis sajta i njegovog sadržaja, zbližava posetioca sa temom i svrhom sajta.



Administrator
Administrator

A Welcome Message

Welcome to The Whisper's Cove Place were u can discuss, and share your experience about the Monkey Island(series), come in, stay awhile and listen or tell your own tresure-hunting adventure!

1 year ago

About Us

The Whisper's Cove is meant to be a forum dedicated to the Monkey Island game series and the people around it. We see ourselves as a part of that community, and we made this place so that the people can share they're passion for the games, aswell as help others on the way, and we see it as a cool geeky nostalgic place where we can hang out and talk to other people and share our stories of the good old days.

Users

We currently have 5 registered users.

Join the Community

Become a member of The Cove and begin to discover the wild world that is Caribbean with other members of our community. Join our forums, share your story and suggestions. Discuss story and lore, and discover "tresures" with others!

Origins

Monkey Island is a series of adventure games. The first four games in the series were produced and published by LucasArts, formerly known as Lucasfilm Games. The fifth installment of the franchise was developed by Telltale Games in collaboration with LucasArts. The games follow the misadventures of the hapless Guybrush Threepwood as he struggles to become the most notorious pirate in the Caribbean, defeat the plans of the evil undead pirate LeChuck and win the heart of governess Elaine Marley. Each game's plot usually involves the mysterious Monkey Island and its impenetrable secrets.

Games Setting

Each of the games takes place on fictional islands in the Caribbean around the Golden Age of Piracy sometime between the 17th and 18th centuries. The islands teem with pirates dressed in outfits that seem to come from films and comic books rather than history, and there are many deliberate anachronisms and references to modern-day popular culture.

Game Info

Developed by - LucasArts (1990–2010), Telltale Games (2009)
Platforms:
Atari ST, Amiga, DOS, Microsoft Windows, macOS, Mega-CD, PlayStation 2, XBLA, WiiWare, PSN, iOS
Release:
First release - The Secret of Monkey Island 1990
Latest release - Monkey Island 2: LeChuck's Revenge – Special Edition 2010
Genre(s): Graphic adventure
Mode(s): Single-player
Creator: Ron Gilbert
Writer(s): Ron Gilbert, Dave Grossman, Tim Schafer
Composer(s): Michael Land, Patrick Mundy
Website www.Scummbar.com

Stay updated
Follow cove:

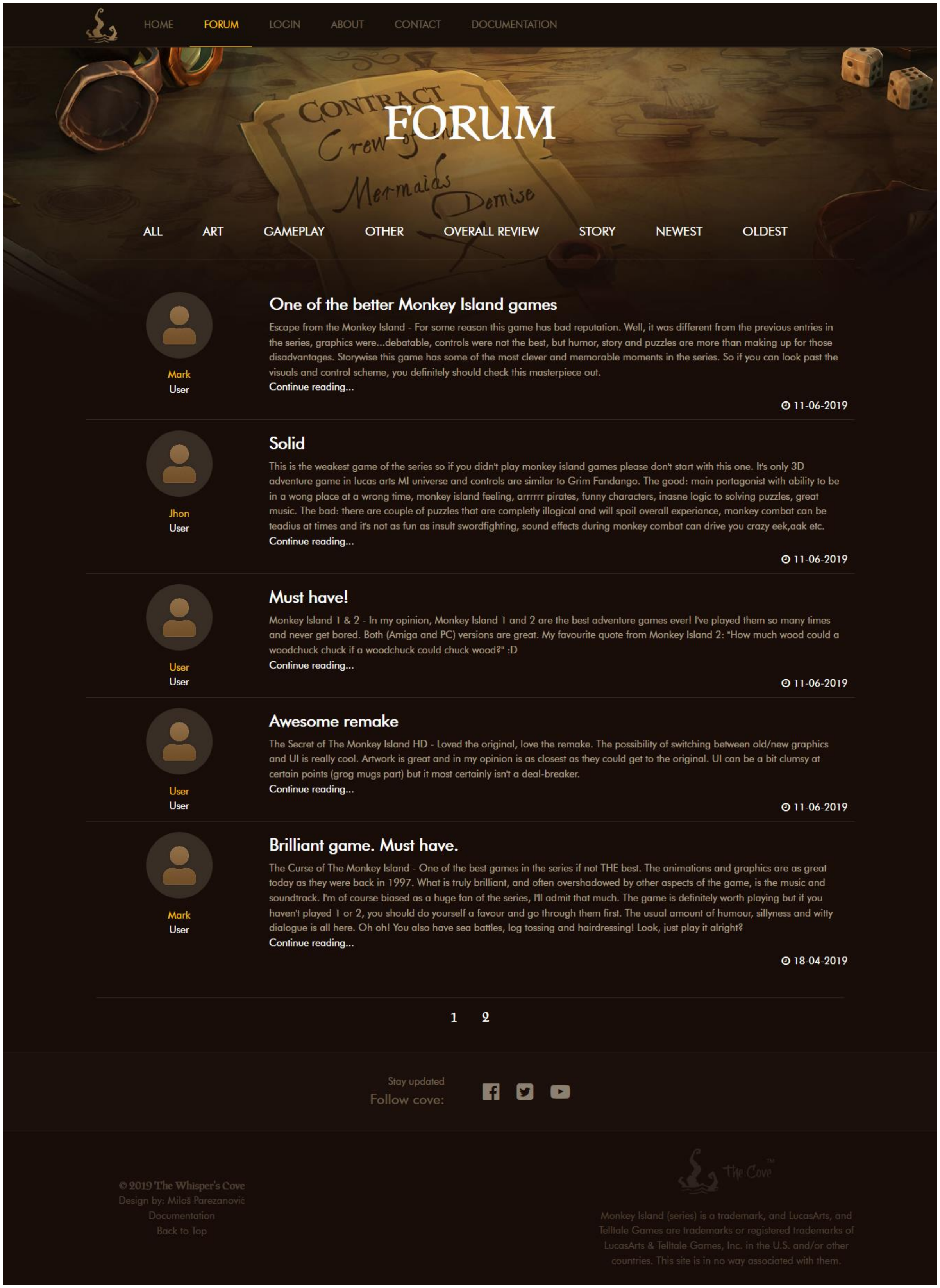
© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

The Cove

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

Forum

Forum strana sadrži postove dinamički ispisane iz baze kao i kategorije – u vidu linkova pomoću kojih se filtrira sadržaj, paginacija po kojoj se stranice postovi.



Mark
User

One of the better Monkey Island games

Escape from the Monkey Island - For some reason this game has bad reputation. Well, it was different from the previous entries in the series, graphics were...debatable, controls were not the best, but humor, story and puzzles are more than making up for those disadvantages. Storywise this game has some of the most clever and memorable moments in the series. So if you can look past the visuals and control scheme, you definitely should check this masterpiece out.

Continue reading...

11-06-2019

Jhon
User

Solid

This is the weakest game of the series so if you didn't play monkey island games please don't start with this one. It's only 3D adventure game in lucas arts MI universe and controls are similar to Grim Fandango. The good: main portagonist with ability to be in a wong place at a wrong time, monkey island feeling, arrrrrr pirates, funny characters, inasne logic to solving puzzles, great music. The bad: there are couple of puzzles that are completly illogical and will spoil overall experiance, monkey combat can be teadius at times and it's not as fun as insult swordfighting, sound effects during monkey combat can drive you crazy eek,aak etc.

Continue reading...

11-06-2019

User
User

Must have!

Monkey Island 1 & 2 - In my opinion, Monkey Island 1 and 2 are the best adventure games ever! I've played them so many times and never get bored. Both (Amiga and PC) versions are great. My favourite quote from Monkey Island 2: "How much wood could a woodchuck chuck if a woodchuck could chuck wood?" :D

Continue reading...

11-06-2019

User
User

Awesome remake

The Secret of The Monkey Island HD - Loved the original, love the remake. The possibility of switching between old/new graphics and UI is really cool. Artwork is great and in my opinion is as closest as they could get to the original. UI can be a bit clumsy at certain points (grog mugs part) but it most certainly isn't a deal-breaker.

Continue reading...

11-06-2019

Mark
User

Brilliant game. Must have.

The Curse of The Monkey Island - One of the best games in the series if not THE best. The animations and graphics are as great today as they were back in 1997. What is truly brilliant, and often overshadowed by other aspects of the game, is the music and soundtrack. I'm of course biased as a huge fan of the series, I'll admit that much. The game is definitely worth playing but if you haven't played 1 or 2, you should do yourself a favour and go through them first. The usual amount of humour, sillyness and witty dialogue is all here. Oh oh! You also have sea battles, log tossing and hairdressing! Look, just play it alright?

Continue reading...

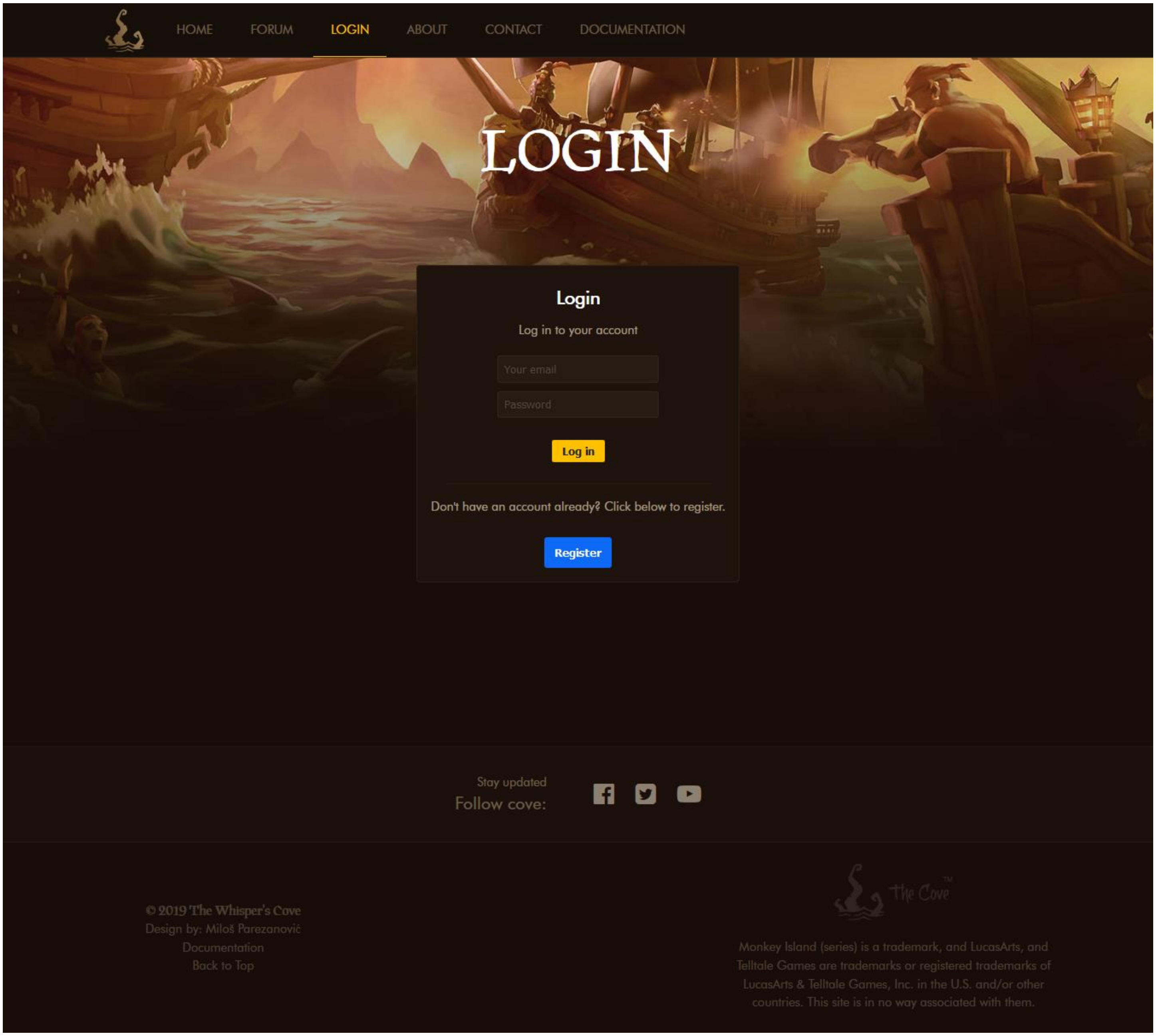
18-04-2019

© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

login

Strana za logovanje, provera unosa i na klijentskoj i na serverskoj strani putem regularnih izraza, posle logovanja user se usmerava na admin ili user stranu u zavisnosti od uloge



Login

Log in to your account

Don't have an account already? Click below to register.

Stay updated

Follow cove:

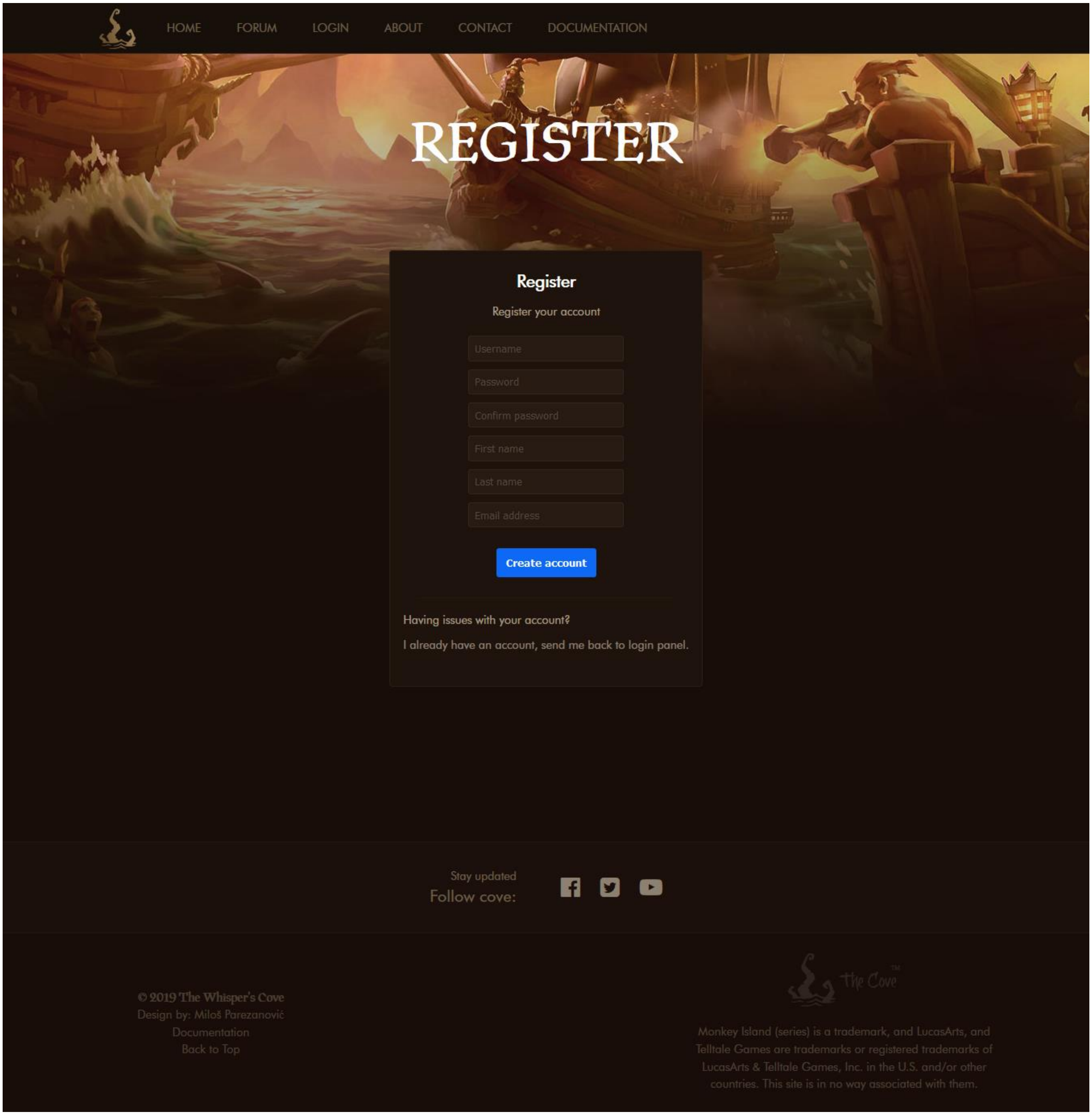
© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

The Cove™

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

register

Strana za registraciju novih korisnika, provera unosa i na klijentskoj i na serverskoj strani putem regularnih izraza, posle uspesne registracije user se salje na login stranu za logovanje, a na mejl dobija link sa aktivacijonim kodom za aktivaciju accounta(kod je napisan za ovu funkcionalnost, sa napomenom da zbog komplikacije sa host providerom ne radi pa je zakomentarisan, a aktivacion kod se salje na register_activation.php, na kojoj se registrovani korisnik aktivira)



Having issues with your account?

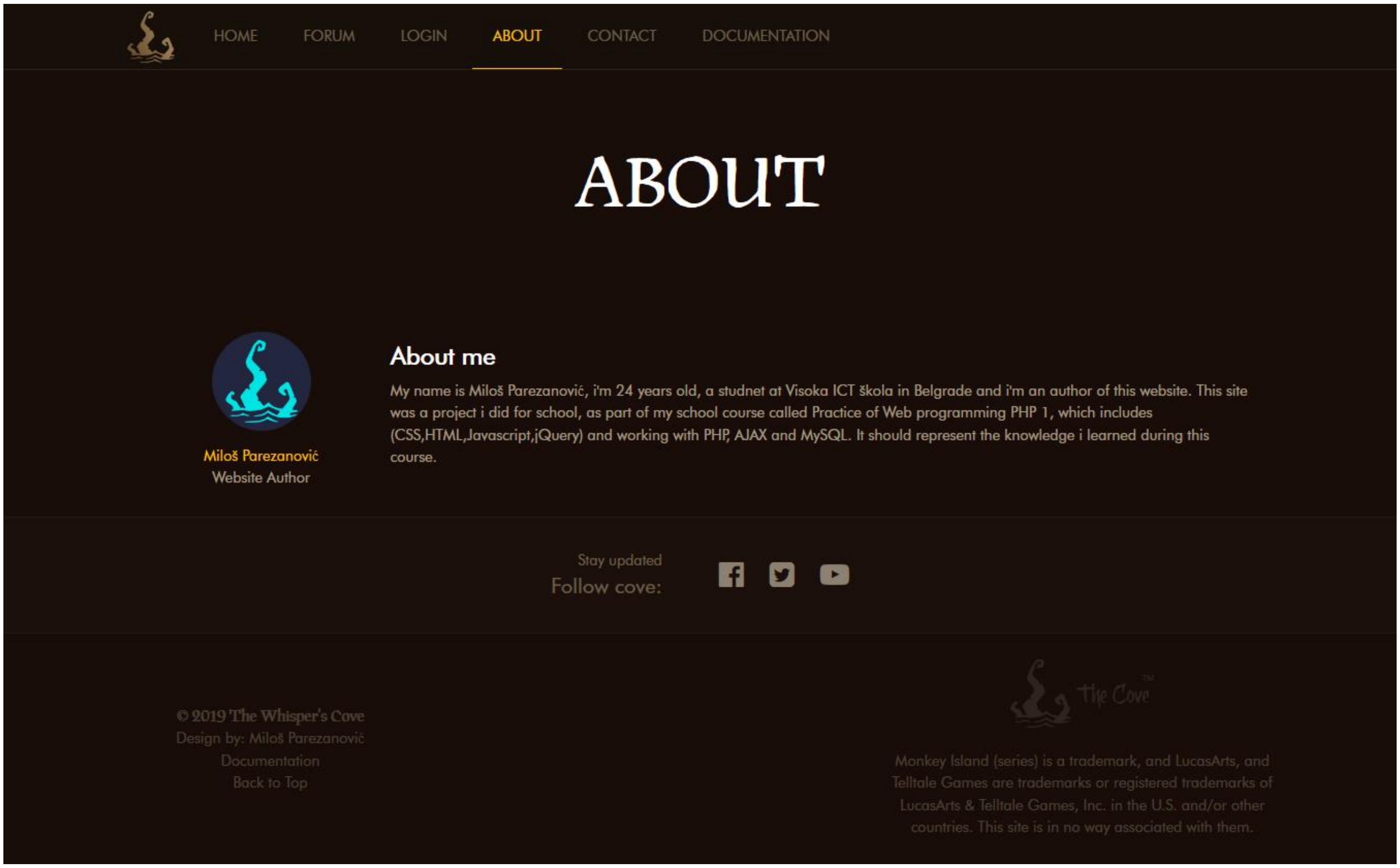
I already have an account, send me back to login panel.

© 2019 The Whisper's Cove
Design by: Miloš Parezanović

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

about

Strana o autoru sa kratkim opisom



HOME FORUM LOGIN ABOUT CONTACT DOCUMENTATION

ABOUT

Miloš Parezanović
Website Author

About me

My name is Miloš Parezanović, i'm 24 years old, a studnet at Visoka ICT škola in Belgrade and i'm an author of this website. This site was a project i did for school, as part of my school course called Practice of Web programming PHP 1, which includes (CSS,HTML,Javascript,jQuery) and working with PHP, AJAX and MySQL. It should represent the knowledge i learned during this course.

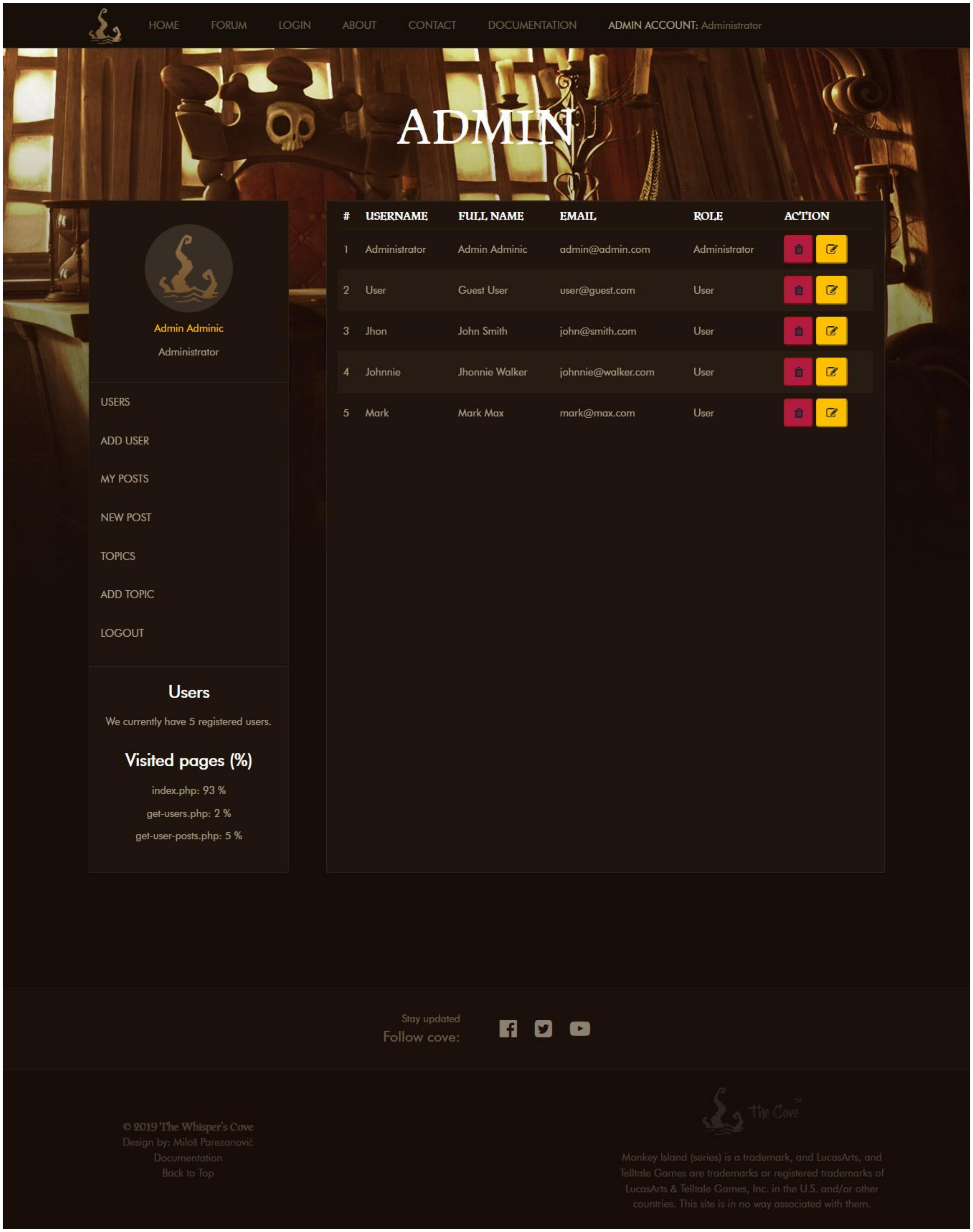
Stay updated
Follow cove:

© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

admin

Admin strana za upravljanje dinamičkim sadržajem sajta, moguće dodavanje usera i postova, kao i kategorija i brisanje istih, pored toga moguća i njihova izmena moguća deaktivacija usera



#	USERNAME	FULL NAME	EMAIL	ROLE	ACTION
1	Administrator	Admin Adminic	admin@admin.com	Administrator	
2	User	Guest User	user@guest.com	User	
3	Jhon	John Smith	john@smith.com	User	
4	Johnnie	Jhonnie Walker	johnnie@walker.com	User	
5	Mark	Mark Max	mark@max.com	User	

Users

We currently have 5 registered users.

Visited pages (%)

index.php: 93 %
get-users.php: 2 %
get-user-posts.php: 5 %

Stay updated
Follow cove:

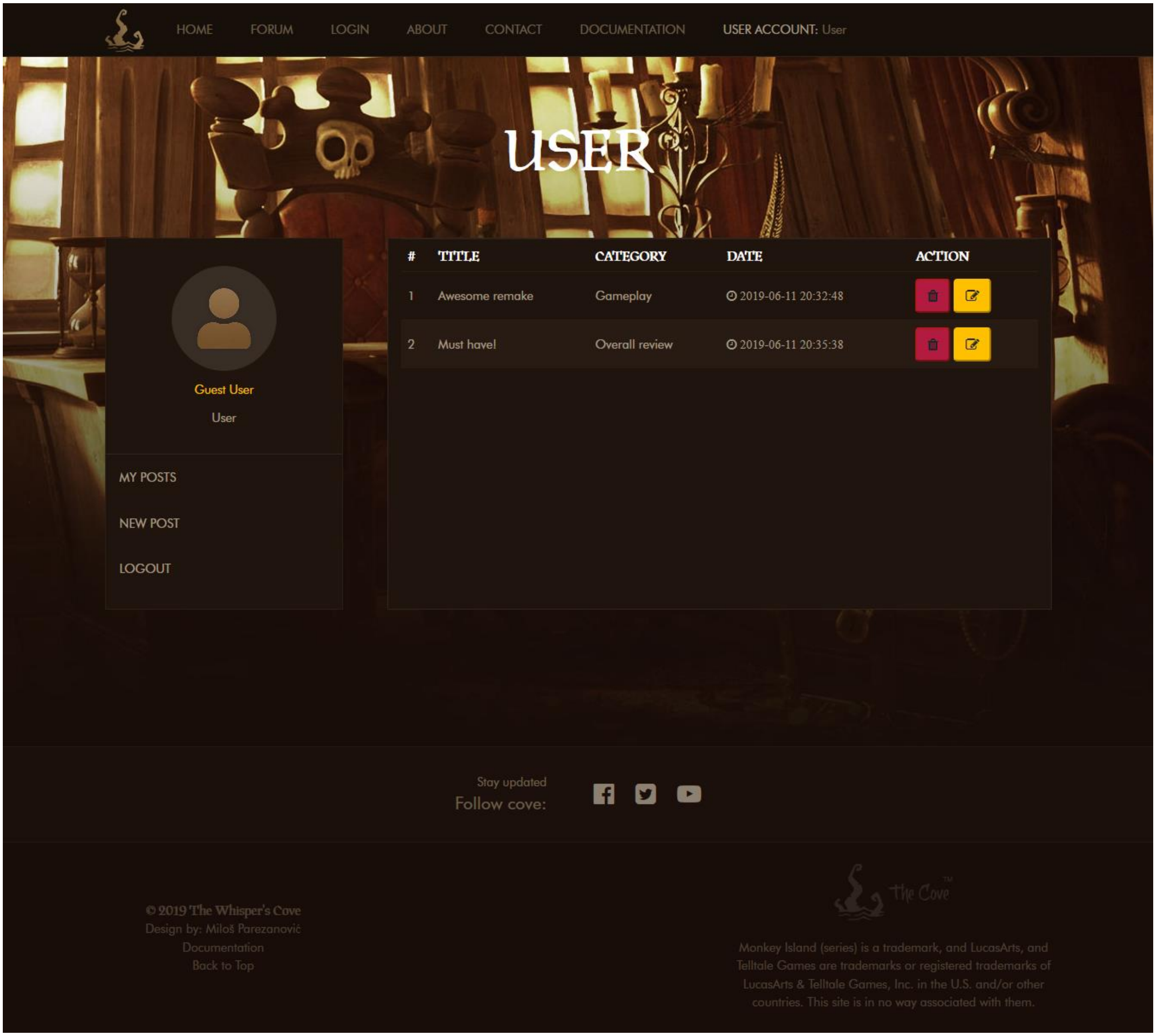
© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

The Cove™

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

user

User strana na kojoj registovani korisnik može dodati, izmeniti ili obrisati neki od svojih postova



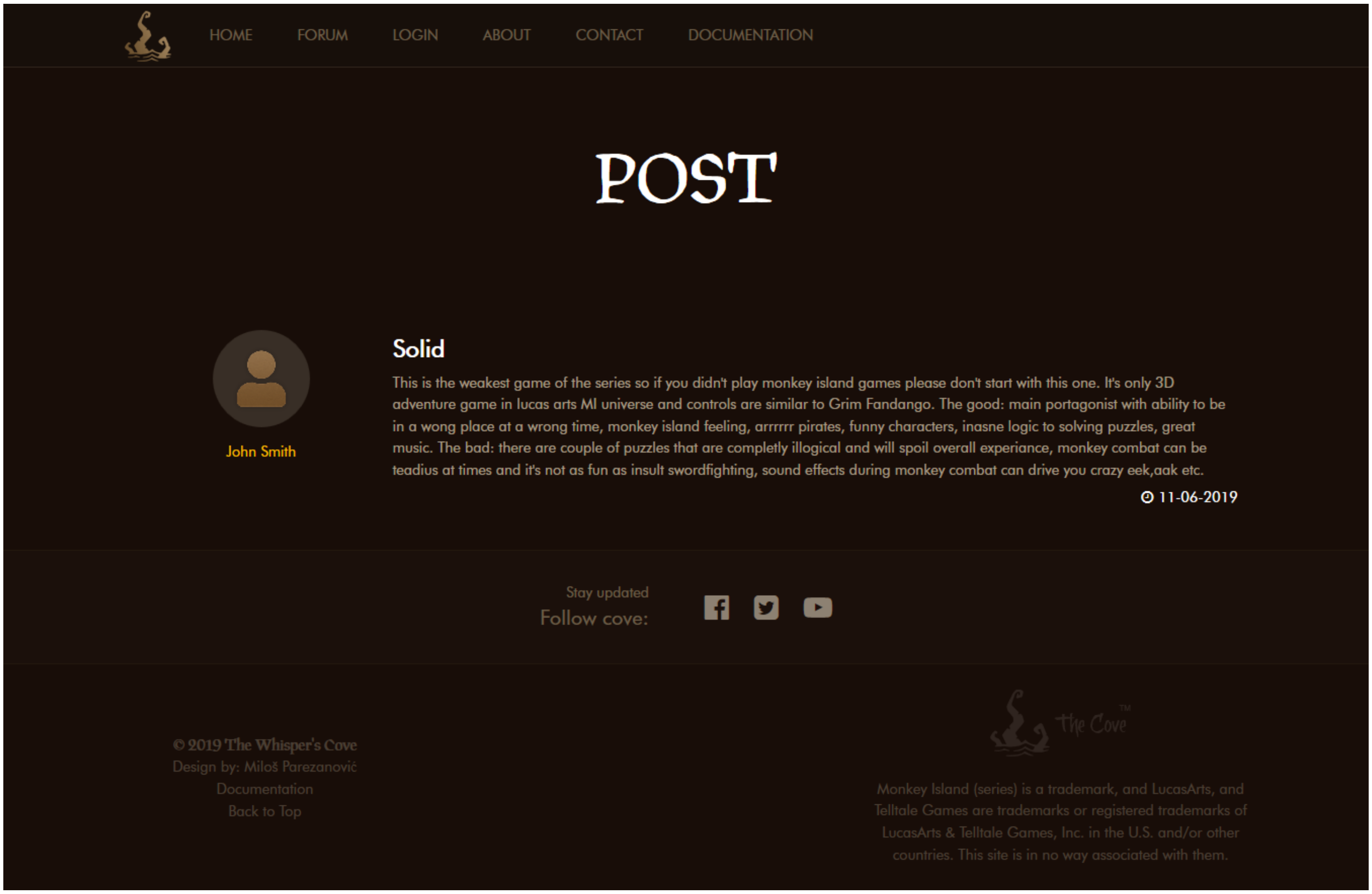
#	TITLE	CATEGORY	DATE	ACTION
1	Awesome remake	Gameplay	2019-06-11 20:32:48	🗑 ✎
2	Must have!	Overall review	2019-06-11 20:35:38	🗑 ✎

© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

post

strana na kojoj mozete procitati celokupan post sa detaljima pri kliku na continue reading na forum ili bilo kojoj drugoj strain



POST

John Smith

Solid

This is the weakest game of the series so if you didn't play monkey island games please don't start with this one. It's only 3D adventure game in lucas arts MI universe and controls are similar to Grim Fandango. The good: main portagonist with ability to be in a wong place at a wrong time, monkey island feeling, arrrrrr pirates, funny characters, inasne logic to solving puzzles, great music. The bad: there are couple of puzzles that are completly illogical and will spoil overall experiance, monkey combat can be teadius at times and it's not as fun as insult swordfighting, sound effects during monkey combat can drive you crazy eek,aak etc.

11-06-2019

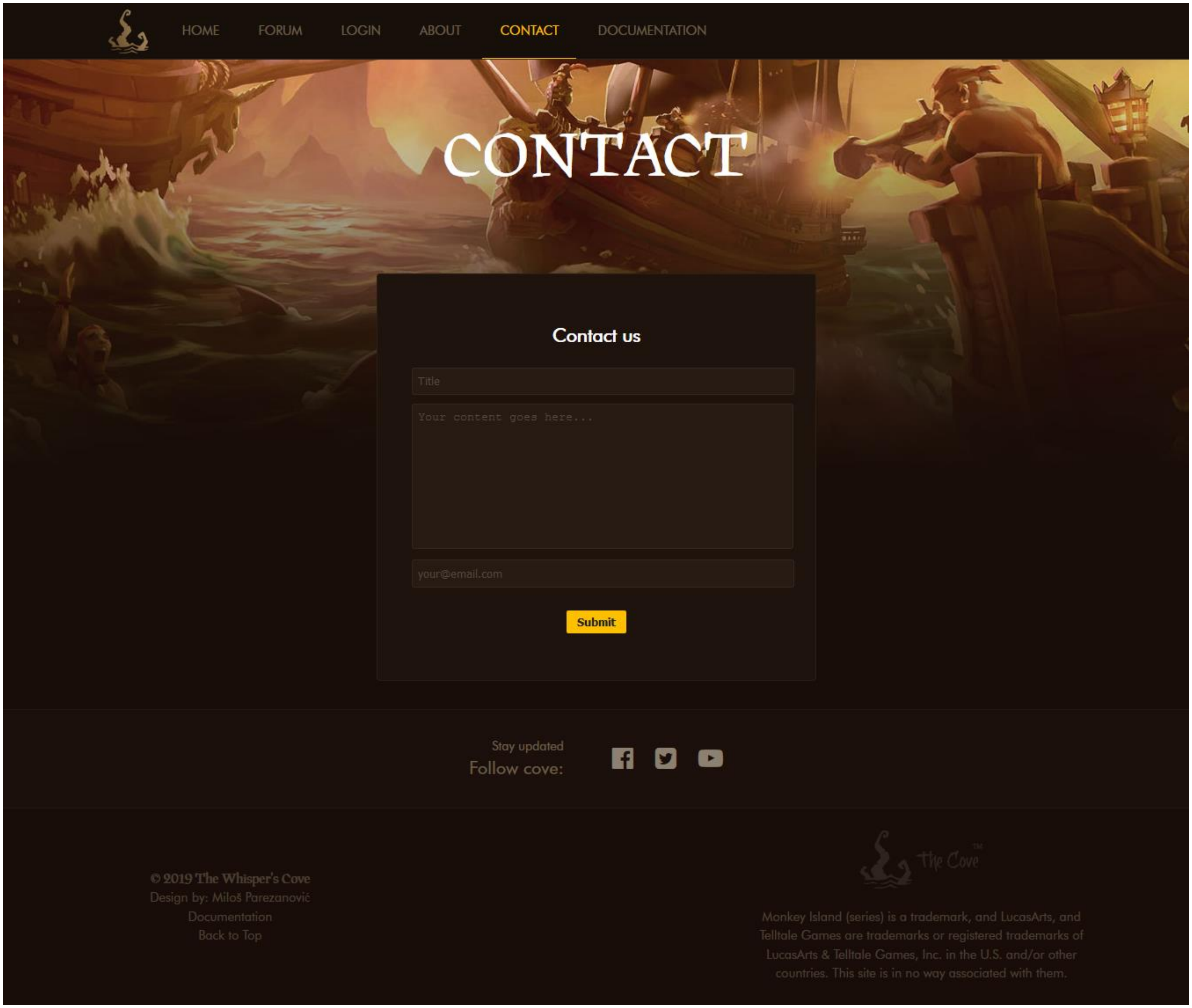
Stay updated
Follow cove:

© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

contact

contact strana na kojoj korisnik salje mejl administratoru sajta unos se proverava



© 2019 The Whisper's Cove
Design by: Miloš Parezanović
Documentation
Back to Top

Monkey Island (series) is a trademark, and LucasArts, and Telltale Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S. and/or other countries. This site is in no way associated with them.

Kodovi

PHP

index.php

```php
<?php

include 'config/init.php';
include 'views/fixed/header.php';

if (isset($_GET['page'])){
    switch ($_GET['page']){
    case "home": include "views/pages/home.php";
    break;
    case "forum": include "views/pages/forum.php";
    break;
    case "post": include "views/pages/post.php";
    break;
    case "login": include "views/partial/login.php";
    break;
    case "register": include "views/partial/register.php";
    break;
    case "user": include "views/pages/user.php";
    break;
    case "admin": include "views/pages/admin.php";
    break;
    case "logout": include "views/partial/logout.php";
    break;
    case "about": include "views/pages/about.php";
    break;
    case "contact": include "views/pages/contact.php";
    break;
    default : include "views/pages/home.php";
    break;
    }
}   else {
        include "views/pages/home.php";
    }

include 'views/fixed/footer.php';

?>
```

header.php

```
<?php

    $pages = array('home','forum', 'login', 'about', 'contact', 'post', 'register', 'admin', 'user');

    if(isset($_GET['page']) && !empty($_GET['page']) && in_array($_GET['page'], $pages)){
        $page = $_GET['page'];
    } else {
        $page="home";
    }

?>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <meta name="description" content="The Whisper's Cove is meant to be a forum dedicated to the Monkey Island game series and the people around it...">
        <meta name="keywords" content="Monkey Island,lucasarts,pirates,blog,forum,review,gameplay">
        <meta name="author" content="Miloš Parezanović">
        <link rel="shortcut icon" href="assets/favicon/favicon.ico">
        <title>The Whisper's Cove</title>
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
        <link rel="stylesheet" type="text/css" href="assets/css/style.css">
    </head>
    <body class="<?php echo $page?>">
    <div id="site">
        <header>
            <?php include 'menu.php'; ?>
        </header>
            <?php
                if(isset($_GET['page']) && !empty($_GET['page'])){
                    $page = $_GET['page'];
                } else {
                    $page="home";
                }
                if($page == "home"){
                    echo "";
                } else if(in_array($page, $pages)) {
                    echo "<div class='main-banner-box'>
                            <span class='page-title'>$page</span>
```

```
                        </div>";
                }
            ?>
        <div class="container">
```

menu.php

```
<?php

  if(isset($_GET['page'])){
    $page = $_GET['page'];
    } else {
        $page="home";
    }

?>

<nav id="main-nav">
    <div class="nav-wrapper">
        <div class="menu-icon">
            <i class="fa fa-bars fa-2x" aria-hidden="true"></i>
        </div>
        <a href="index.php?page=home">
            <div class="head-logo">
                <span class="nav-logo"></span>
            </div>
        </a>
        <div class="nav-menu">
            <ul>
                <?php
                include 'models/functions/get-links.php';

                $menu = get_menu();
                //print_r($menu);

                    foreach($menu as $link) :

                ?>
                    <li class="<?php echo ($page == "$link-
>link_name" ? "active" : "");?>" ><a href="<?= $link->href ?>"><?= $link-
>link_name ?></a></li>

                    <?php endforeach; ?>

                <li><a href="doc.pdf" target="_blank">Documentation</a></li>
                <?php
```

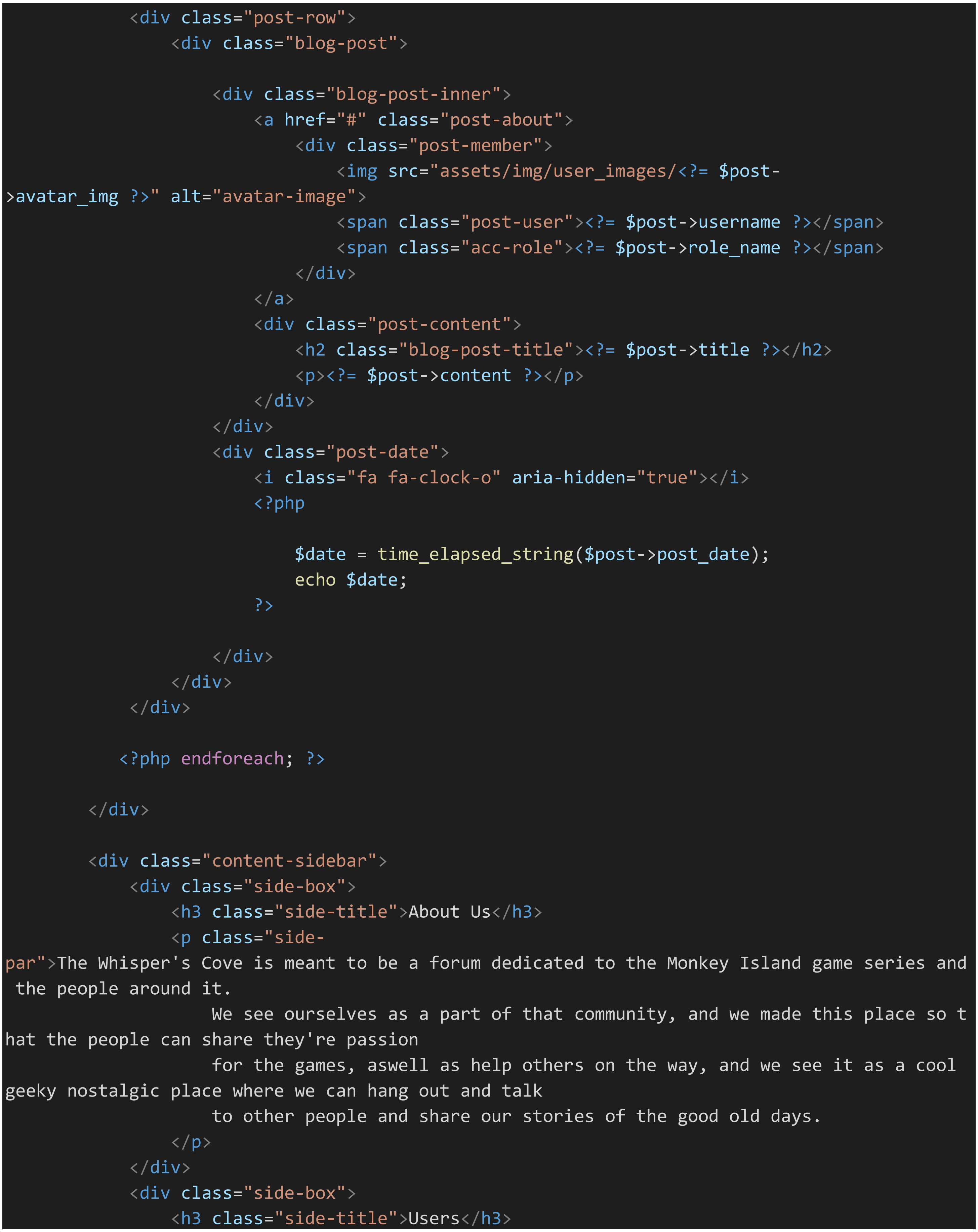
```
                    if(isset($_SESSION['user'])) :
                        $user = $_SESSION['user'];
                        if($_SESSION['user']->role_id == 2) :
                ?>
                    <li class="my-acc <?php echo ($page == "loggedin" ? "active" : "");?>"> <a href="index.php?page=user">My Account</a></li>
                    <li class="acc-nav">
                    <span>User Account:</span>
                    <span class="nav-user"><a href="index.php?page=user"><?= $user->username ?></a></span>
                    </li>
                        <?php endif; ?>

                    <?php if($_SESSION['user']->role_id == 1) : ?>

                    <li class="my-acc <?php echo ($page == "loggedin" ? "active" : "");?>"> <a href="index.php?page=admin">My Account</a></li>
                    <li class="acc-nav">
                    <span>Admin Account:</span>
                    <span class="nav-user"><a href="index.php?page=admin"><?= $user->username ?></a></span>
                    </li>
                        <?php endif; ?>
                <?php endif; ?>
            </ul>
        </div>
    </div>
</nav>
```

footer.php

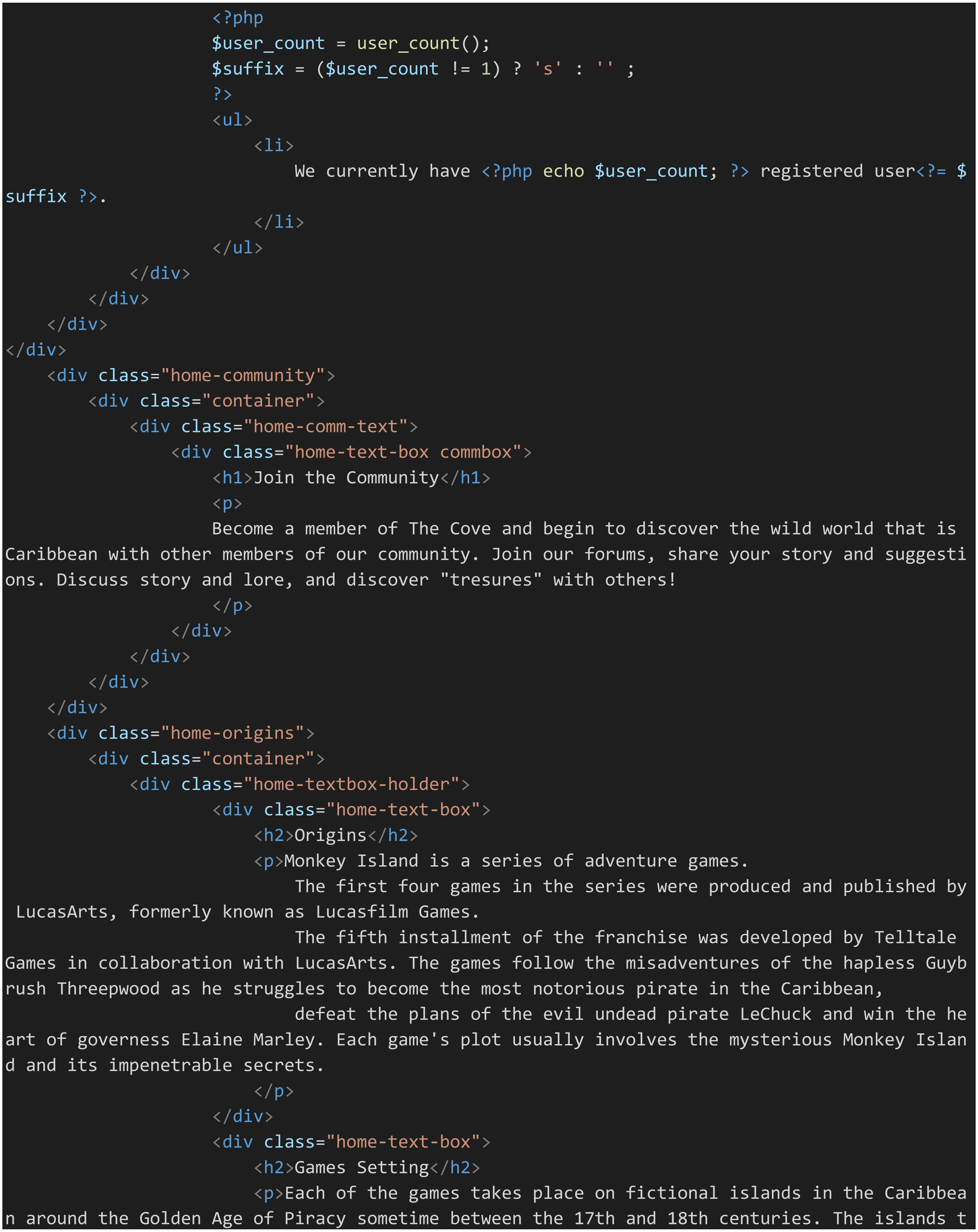
```

        <div class="follower">
            <div class="follower-wrapper">
                <div class="follow-text">
                    Stay updated
                    <span class="follow-title">Follow cove:</span>
                </div>
                <div class="follow-media">
                    <a href="https://www.facebook.com/" target="_blank"><i class="fa fa-facebook-official smedia-face"></i></a>
                    <a href="https://twitter.com/" target="_blank"><i class="fa fa-twitter-square smedia-twit"></i></a>
                    <a href="https://www.youtube.com/" target="_blank"><i class="fa fa-youtube-play smedia-ytube"></i></a>
                </div>
            </div>
        </div>
        <footer>
            <div class="container">
                <div class="footer-row">
                    <div class="footer-left">
                        <div>
                            <p class="tm">© 2019 The Whisper's Cove</p>
                            <p>Design by: <a href="index.php?page=about">Miloš Parezanović</a
></p>

                            <p><a href="doc.pdf" target="_blank">Documentation</a></p>
                            <p><a href="#">Back to Top</a></p>

                        </div>
                    </div>
                    <div class="footer-between"></div>
                    <div class="footer-right">
                        <span class="logo-footer"></span>
                        <p>Monkey Island (series) is a trademark, and LucasArts, and Telltale
 Games are trademarks or registered trademarks of LucasArts & Telltale Games, Inc. in the U.S
. and/or other countries. This site is in no way associated with them.</p>
                    </div>
                </div>
            </div>
        </footer>
    </div>
        <script src="assets/js/main.js"></script>
        <script type="text/javascript">
            $(document).ready(function() {
                $('.menu-icon').click(function() {
                    $('nav ul').toggleClass('show');
                })
            })
        </script>
    </body>
</html>
```

home.php

```
    <div class="main-content">
        <div class="home-left-content-area">
        <?php
            include 'models/functions/functions.php';

            $posts = front_top();
           // print_r($posts);
           foreach($posts as $post):
        ?>
```



```
            <div class="post-row">
                <div class="blog-post">

                    <div class="blog-post-inner">
                        <a href="#" class="post-about">
                            <div class="post-member">
                                <img src="assets/img/user_images/<?= $post-
>avatar_img ?>" alt="avatar-image">
                                <span class="post-user"><?= $post->username ?></span>
                                <span class="acc-role"><?= $post->role_name ?></span>
                            </div>
                        </a>
                        <div class="post-content">
                            <h2 class="blog-post-title"><?= $post->title ?></h2>
                            <p><?= $post->content ?></p>
                        </div>
                    </div>
                    <div class="post-date">
                        <i class="fa fa-clock-o" aria-hidden="true"></i>
                        <?php

                            $date = time_elapsed_string($post->post_date);
                            echo $date;
                        ?>

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

           <?php endforeach; ?>

        </div>

        <div class="content-sidebar">
            <div class="side-box">
                <h3 class="side-title">About Us</h3>
                <p class="side-
par">The Whisper's Cove is meant to be a forum dedicated to the Monkey Island game series and
 the people around it.
                    We see ourselves as a part of that community, and we made this place so t
hat the people can share they're passion
                    for the games, aswell as help others on the way, and we see it as a cool
geeky nostalgic place where we can hang out and talk
                    to other people and share our stories of the good old days.
                </p>
            </div>
            <div class="side-box">
                <h3 class="side-title">Users</h3>
```

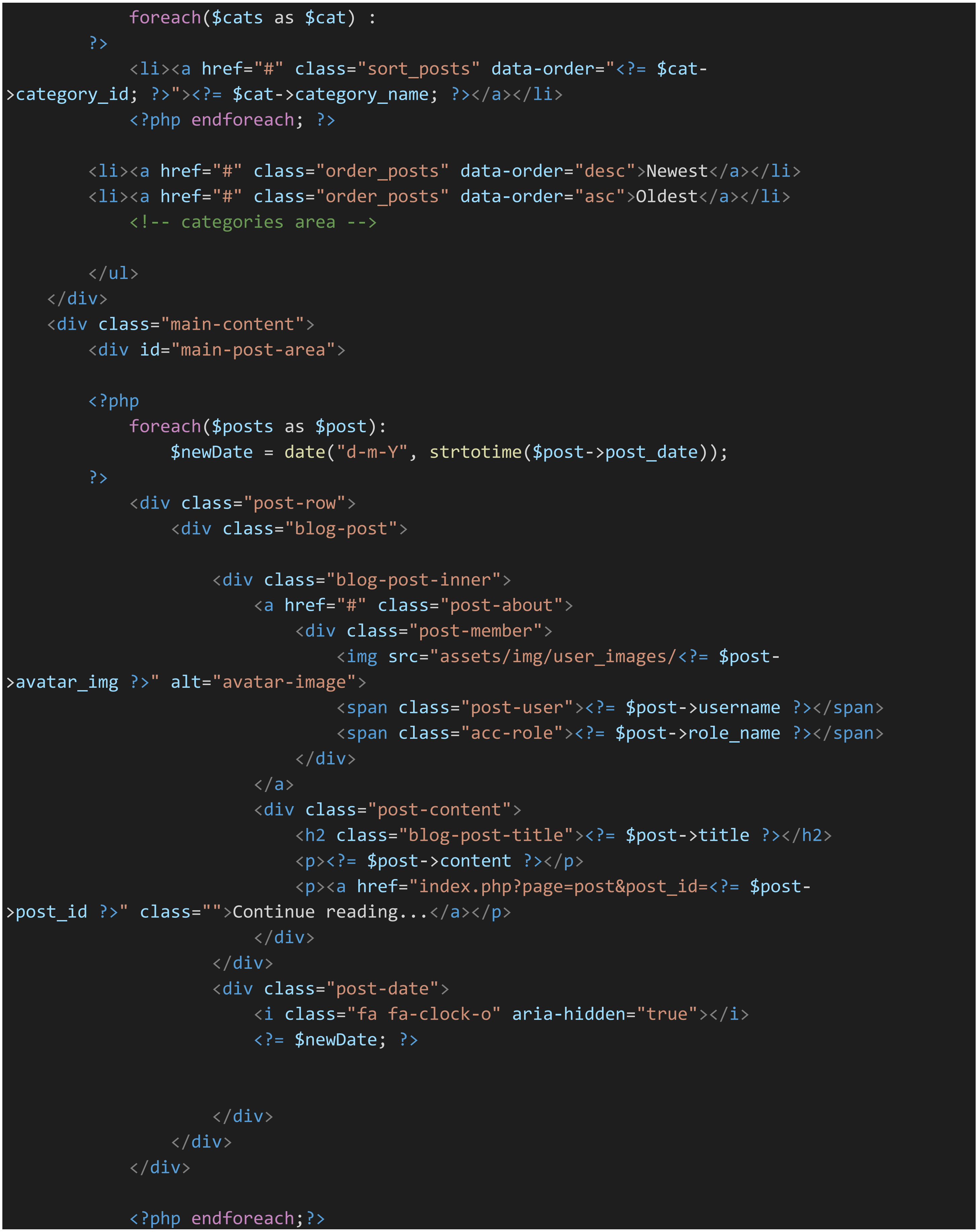


```
                    <?php
                    $user_count = user_count();
                    $suffix = ($user_count != 1) ? 's' : '' ;
                    ?>
                    <ul>
                        <li>
                            We currently have <?php echo $user_count; ?> registered user<?= $
suffix ?>.
                        </li>
                    </ul>
            </div>
        </div>
    </div>
</div>
    <div class="home-community">
        <div class="container">
            <div class="home-comm-text">
                <div class="home-text-box commbox">
                    <h1>Join the Community</h1>
                    <p>
                    Become a member of The Cove and begin to discover the wild world that is
Caribbean with other members of our community. Join our forums, share your story and suggesti
ons. Discuss story and lore, and discover "tresures" with others!
                    </p>
                </div>
            </div>
        </div>
    </div>
    <div class="home-origins">
        <div class="container">
            <div class="home-textbox-holder">
                    <div class="home-text-box">
                        <h2>Origins</h2>
                        <p>Monkey Island is a series of adventure games.
                            The first four games in the series were produced and published by
 LucasArts, formerly known as Lucasfilm Games.
                            The fifth installment of the franchise was developed by Telltale
Games in collaboration with LucasArts. The games follow the misadventures of the hapless Guyb
rush Threepwood as he struggles to become the most notorious pirate in the Caribbean,
                            defeat the plans of the evil undead pirate LeChuck and win the he
art of governess Elaine Marley. Each game's plot usually involves the mysterious Monkey Islan
d and its impenetrable secrets.
                        </p>
                    </div>
                    <div class="home-text-box">
                        <h2>Games Setting</h2>
                        <p>Each of the games takes place on fictional islands in the Caribbea
n around the Golden Age of Piracy sometime between the 17th and 18th centuries. The islands t
```

```
eem with pirates dressed in outfits that seem to come from films and comic books rather than
history, and there are many deliberate anachronisms and references to modern-
day popular culture.  </p>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="home-about-game">
        <div class="container">
            <div class="home-about-text">
                <div class="home-text-box">
                    <h1>Game Info</h1>
                    <ol class="info-list">
                        <li>Developed by - LucasArts (1990–2010), Telltale Games (2009)</li>
                        <li>Platforms: <br/> Atari ST, Amiga, DOS, Microsoft Windows, macOS,
Mega-CD, <br/> PlayStation 2, XBLA, WiiWare, PSN, iOS</li>
                        <li>Release: <br/> First release -
 The Secret of Monkey Island 1990 <br/> Latest release -
 Monkey Island 2: LeChuck's Revenge – Special Edition 2010</li>
                        <li>Genre(s): Graphic adventure</li>
                        <li>Mode(s): Single-player</li>
                        <li>Creator: Ron Gilbert</li>
                        <li>Writer(s): Ron Gilbert, Dave Grossman, Tim Schafer</li>
                        <li>Composer(s): Michael Land, Patrick Mundy</li>
                        <li>Website <a href="https://scummbar.com/" target="_blank"> www.Scum
mbar.com</a></li>
                    </ol>
                </div>
            </div>
        </div>
    </div>
```

forum.php

```
    <?php
        include 'models/functions/functions.php';
        $cats = get_categories();
        //print_r($cats);
        $limit =  isset($_GET['limit'])? $_GET['limit'] : 0;
        $posts = get_posts($limit);
        //print_r($posts);
    ?>
    <div id="sort-content">
        <ul id="sort-ul">
        <li><a href="#" class="reset">All</a></li>
        <?php
```



```
            foreach($cats as $cat) :
        ?>
            <li><a href="#" class="sort_posts" data-order="<?= $cat-
>category_id; ?>"><?= $cat->category_name; ?></a></li>
            <?php endforeach; ?>

        <li><a href="#" class="order_posts" data-order="desc">Newest</a></li>
        <li><a href="#" class="order_posts" data-order="asc">Oldest</a></li>
            <!-- categories area -->

        </ul>
    </div>
    <div class="main-content">
        <div id="main-post-area">

        <?php
            foreach($posts as $post):
                $newDate = date("d-m-Y", strtotime($post->post_date));
        ?>
            <div class="post-row">
                <div class="blog-post">

                    <div class="blog-post-inner">
                        <a href="#" class="post-about">
                            <div class="post-member">
                                <img src="assets/img/user_images/<?= $post-
>avatar_img ?>" alt="avatar-image">
                                <span class="post-user"><?= $post->username ?></span>
                                <span class="acc-role"><?= $post->role_name ?></span>
                            </div>
                        </a>
                        <div class="post-content">
                            <h2 class="blog-post-title"><?= $post->title ?></h2>
                            <p><?= $post->content ?></p>
                            <p><a href="index.php?page=post&post_id=<?= $post-
>post_id ?>" class="">Continue reading...</a></p>
                        </div>
                    </div>
                    <div class="post-date">
                        <i class="fa fa-clock-o" aria-hidden="true"></i>
                        <?= $newDate; ?>


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

            <?php endforeach;?>
            <!-- post area -->

        </div>
    </div>
    <div class="container">
        <div class="page-pag">
            <ul id="pag-list">
            <?php
                $num_of_posts = get_pagination_count();
                for($i = 0; $i < $num_of_posts; $i++):
            ?>
                <li><a href="#" class="post-pag" data-limit="<?= $i ?>"><?= $i+1 ?></a></li>
                <?php endfor; ?>
            </ul>
        </div>
    </div>
</div>
```

about.php

```
    <div class="main-content">
        <div class="post-row">
            <div class="blog-post">

                <div class="blog-post-inner">
                    <div class="post-member">
                        <img src="assets/img/about-avatar.jpg" alt="avatar-image">
                        <span class="post-user">Miloš Parezanović</span>
                        <span class="acc-role">Website Author</span>
                    </div>
                    <div class="post-content">
                        <h2 class="blog-post-title">About me</h2>
                        <p>My name is Miloš Parezanović, i'm 24 years old, a studnet at Visok
a ICT škola in Belgrade and i'm an author of this website. This site was a project i did for
school, as part of my school course called Practice of Web programming PHP 1, which includes
(CSS,HTML,Javascript,jQuery) and working with PHP, AJAX and MySQL. It should represent the kn
owledge i learned during this course. </p>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

admin.php

```
<?php
    if(isset($_SESSION['user'])){
        if($_SESSION['user']->role_id != 1){
            header("Location: index.php");
        }
    } else {
        header("Location: index.php");
    }
?>
<div id="acc-wrapper">
            <div class="account-sidebar">
                <div class="side-box side-avatar">

                    <?php
                        if(isset($_SESSION['user'])) :
                            $user = $_SESSION['user'];
                    ?>

                    <img src="assets/img/user_images/<?= $user->avatar_img ?>">
                    <span class="acc-name"><?php echo $user->first_name . " " . $user-
>last_name; ?></span>
                    <span class="acc-role"><?= $user->role_name; ?></span>
                        <?php endif; ?>
                </div>
                <div class="side-box acc-side-box">
                    <ul>
                        <li><a href="#" class="acc-opt" data-use="users">Users</a></li>
                        <li><a href="#" class="acc-opt" data-use="add">Add user</a></li>
                        <li><a href="#" class="acc-opt" data-use="mpost" data-
user_id="<?= $user->user_id ?>">My Posts</a></li>
                        <li><a href="#" class="acc-opt" data-use="npost" data-
user_id="<?= $user->user_id ?>">New Post</a></li>
                        <li><a href="#" class="acc-opt" data-use="top">Topics</a></li>
                        <li><a href="#" class="acc-opt" data-use="ntop">Add Topic</a></li>
                        <li><a href="index.php?page=logout">Logout</a></li>
                    </ul>
                </div>
                <div class="side-box acc-users">
                    <h3 class="side-title">Users</h3>
                    <?php
                    $user_count = user_count();
                    $suffix = ($user_count != 1) ? 's' : '' ;
                    ?>
                    <ul>
                        <li>
                            We currently have <?php echo $user_count; ?> registered user<?= $
suffix ?>.
```

```
            </li>
          </ul>
          <h3 class="side-title">Visited pages (%)</h3>
          <ul>
          <?php

            include 'views/partial/page-stats.php';
            $stat = ispis_procenta();
            foreach($stat as $item) :
          ?>
            <li><?= $item; ?></li>

            <?php endforeach; ?>
          </ul>
        </div>
      </div>
      <div class="account-content-area">
        <div class="container">
          <div id="acc-content">
              <h1>Action area</h1>
              <p>Place when things happen...</p>


            <!-- acc-content -->

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

contact.php

```
  <div class="form-container acc-form">
    <div class="form-inner">
      <ul class="acc-form">
        <li><h2>Contact us</h2></li>
          <?php
            if(isset($_SESSION['errors'])){
              $error = $_SESSION['error'];
              echo ("<li class='error'>" . $error . "</li>");
              session_destroy();
            }

            if (isset($_SESSION['success'])){
              $success = $_SESSION['success'];
              echo ("<li class='errorOk'>" . $success . "</li>");
```

```
                            session_destroy();
                        }
                    ?>
                <li><form action="" method="POST" id="post-form"></li>
                <li><input id="contact_title" type="text" placeholder="Title" required></li>
                <li><textarea id="contact_body" placeholder="Your content goes here..." required></textarea></li>
                <li><input id="contact_email" type="email" placeholder="your@email.com" required></li>
                <li>
                    <button id="contact_submit" type="submit" class="btn-login" onclick="validate_contact()" >Submit</button>
                    </form>
                </li>
            </ul>
        </div>
    </div>
</div>
```

post.php

```
<?php

    if(isset($_GET['post_id']) && !empty($_GET['post_id'])){

        $post_id = $_GET['post_id'];

        $sql = "SELECT p.*, c.category_name, concat(u.first_name, ' ', u.last_name) AS full_name, u.avatar_img FROM posts p INNER JOIN users u ON p.user_id = u.user_id INNER JOIN category_post cp on p.post_id = cp.post_id INNER JOIN categories c ON cp.category_id = c.category_id WHERE p.post_id = :id";
        $stmt = $conn->prepare($sql);

        $stmt->bindParam(":id", $post_id);

        $stmt->execute();

        $post = $stmt->fetchAll();

        ?>
            <div class="main-content">
                <div id="main-post-area">
                    <?php foreach($post as $row) : ?>
                        <?php $newDate = date("d-m-Y", strtotime($row->post_date)); ?>
                            <div class="post-row">
                                <div class="blog-post">

                                    <div class="blog-post-inner">
```

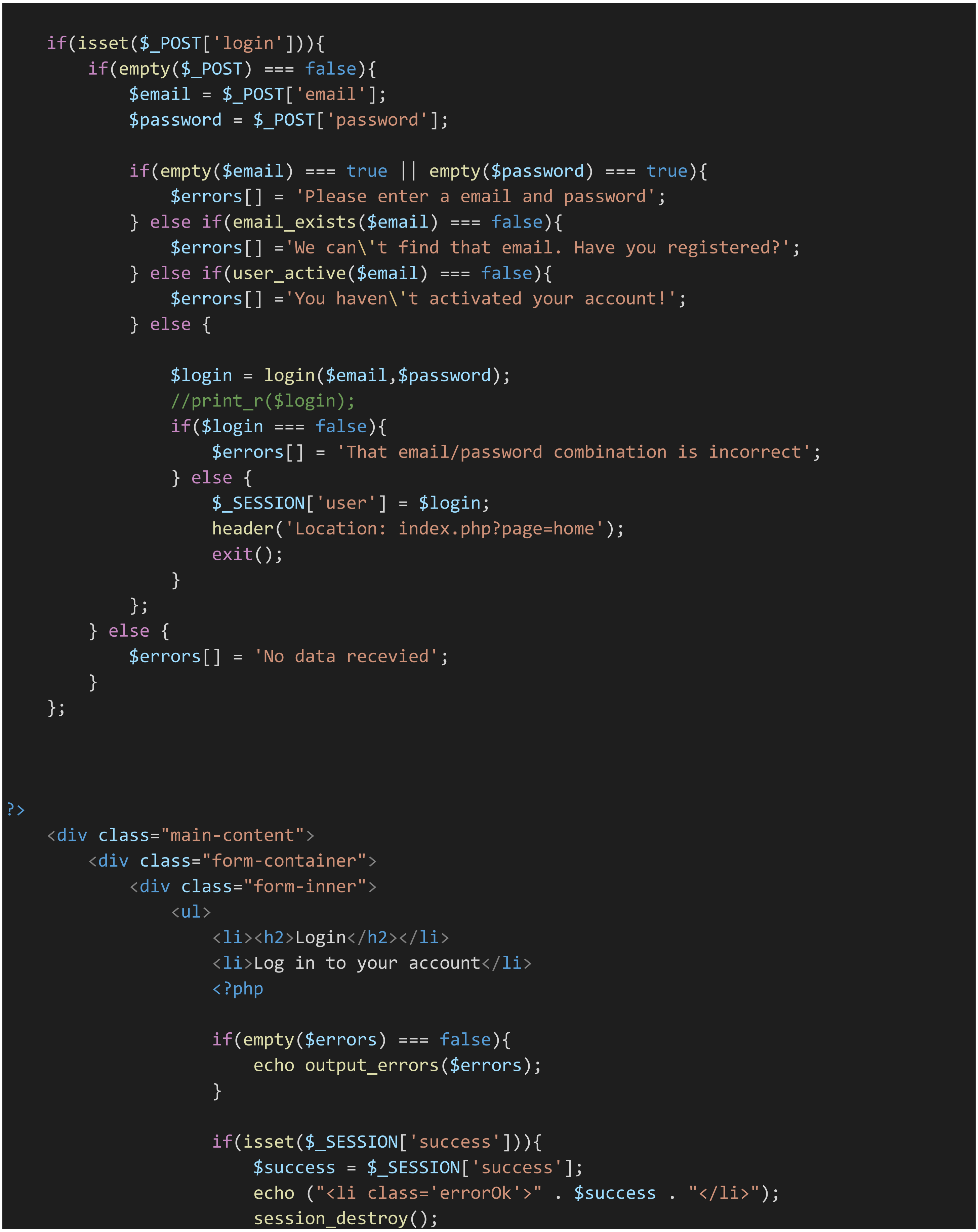
```
                                        <a href="#" class="post-about">
                                            <div class="post-member">
                                                <img src="assets/img/user_images/<?= $row-
>avatar_img ?>" alt="avatar-image">
                                                <span class="post-user"><?= $row-
>full_name ?></span>
                                            </div>
                                        </a>
                                        <div class="post-content">
                                            <h2 class="blog-post-title"><?= $row-
>title ?></h2>
                                            <p><?= $row->content ?></p>
                                        </div>
                                    </div>
                                    <div class="post-date">
                                        <i class="fa fa-clock-o" aria-hidden="true"></i>
                                        <?= $newDate; ?>


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


        <?php

    }else{
        echo "<div class='main-
content'> <p>Can't find your post, it's maybe somewhere else? <a href='index.php?page=forum'>
Go back</a></p></div> </div>";
    }

    //var_dump($post);
```

user.php

```
<?php

if(isset($_SESSION['user'])){
    if($_SESSION['user']->role_id != 2){
        header("Location: index.php");
    }
} else {
    header("Location: index.php");
}
?>
        <div id="acc-wrapper">
            <div class="account-sidebar">
                <div class="side-box side-avatar">

                    <?php
                        if(isset($_SESSION['user'])) :
                            $user = $_SESSION['user'];
                    ?>

                    <img src="assets/img/user_images/<?= $user->avatar_img; ?>">
                    <span class="acc-name"><?php echo $user->first_name .  " " . $user->last_name; ?></span>
                    <span class="acc-role"><?= $user->role_name; ?></span>

                        <?php endif; ?>

                </div>
                <div class="side-box acc-side-box">
                    <ul>
                        <li><a href="#" class="acc-opt" data-use="mpost" data-user_id="<?= $user->user_id ?>">My Posts</a></li>
                        <li><a href="#" class="acc-opt" data-use="npost" data-user_id="<?= $user->user_id ?>">New Post</a></li>
                        <li><a href="index.php?page=logout">Logout</a></li>
                    </ul>
                </div>
            </div>
            <div class="account-content-area">
                <div class="container">
                    <div id="acc-content">

                        <!-- acc-content -->

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

login.php

```
<?php
```



```
    if(isset($_POST['login'])){
        if(empty($_POST) === false){
            $email = $_POST['email'];
            $password = $_POST['password'];

            if(empty($email) === true || empty($password) === true){
                $errors[] = 'Please enter a email and password';
            } else if(email_exists($email) === false){
                $errors[] ='We can\'t find that email. Have you registered?';
            } else if(user_active($email) === false){
                $errors[] ='You haven\'t activated your account!';
            } else {

                $login = login($email,$password);
                //print_r($login);
                if($login === false){
                    $errors[] = 'That email/password combination is incorrect';
                } else {
                    $_SESSION['user'] = $login;
                    header('Location: index.php?page=home');
                    exit();
                }
            };
        } else {
            $errors[] = 'No data recevied';
        }
    };



?>
    <div class="main-content">
        <div class="form-container">
            <div class="form-inner">
                <ul>
                    <li><h2>Login</h2></li>
                    <li>Log in to your account</li>
                    <?php

                    if(empty($errors) === false){
                        echo output_errors($errors);
                    }

                    if(isset($_SESSION['success'])){
                        $success = $_SESSION['success'];
                        echo ("<li class='errorOk'>" . $success . "</li>");
                        session_destroy();
                    }

                    if(isset($_SESSION['error'])){
                        $error = $_SESSION['error'];
                        echo ("<li class='error'>" . $error . "</li>");
                        session_destroy();
                    }

                    ?>
                    <li><form action="" method="post" id="login-form" ></li>
                    <li><input name="email" type="email" id="" class="" placeholder="Your email" autofocus></li>
                    <li><input name="password" type="password" id="" class="" placeholder="Password"></li>
                    <li>
                        <button name="login" class="btn-login" type="submit">Log in</button>
                        </form>
                    </li>
                </ul>
            </div>
            <div class="form-inner">
                <ul>
                    <li class="line"></li>
                    <li>Don't have an account already? Click below to register.</li>
                </ul>
                <a href="index.php?page=register"><button name="register" type="submit" class="btn-reg">Register</button></a>
            </div>
        </div>
    </div>
</div>
```

logout.php

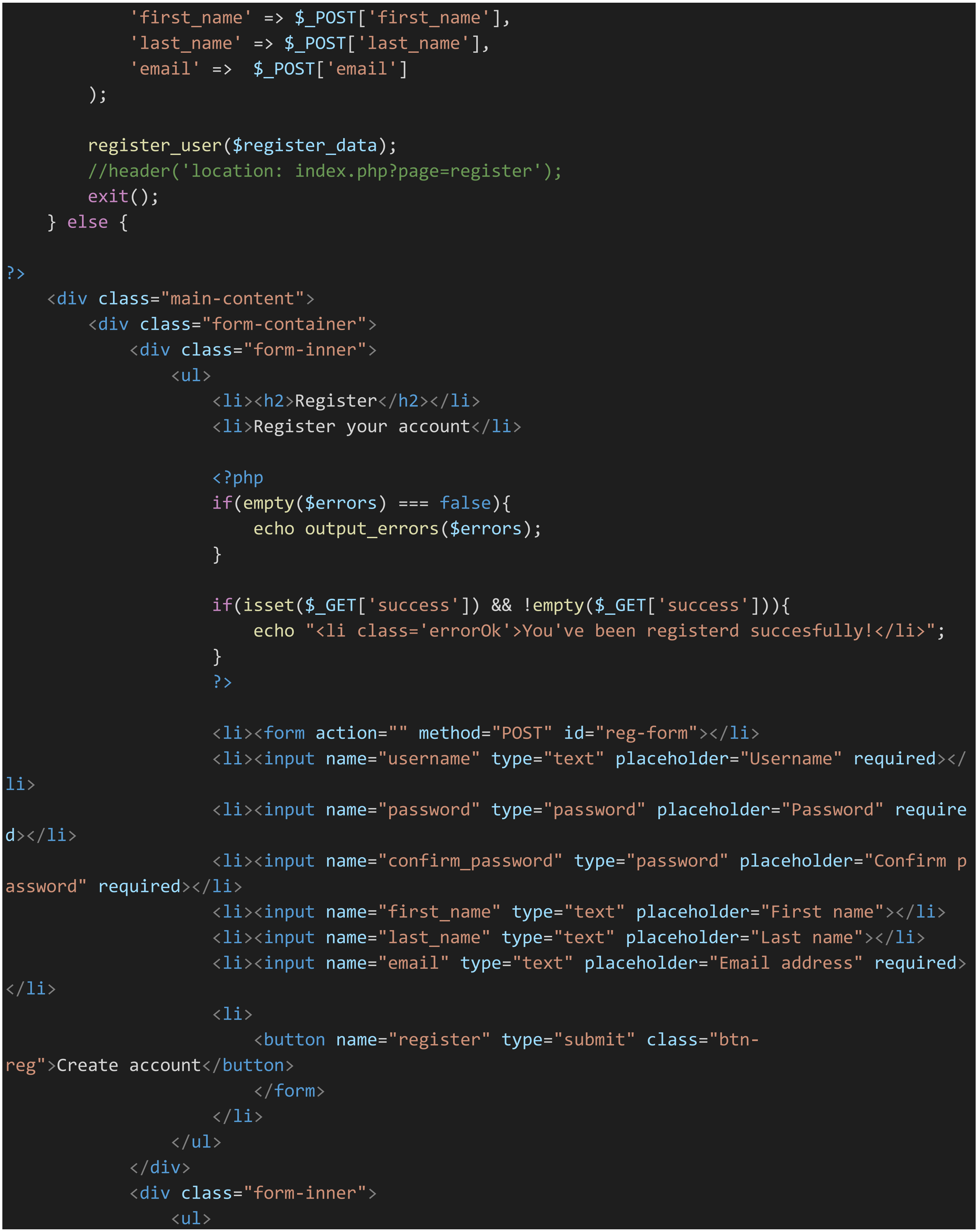
```
<?php

    session_start();
    if($_SESSION['user']){
        session_destroy();
        header('Location: index.php?page=home');
    }

?>
```

register.php

```
<?php
if(isset($_POST['register'])){

    if(empty($errors) === true ){
        if(user_exists($_POST['username']) === true){
            $errors[] = 'Sorry, the username \'' . $_POST['username'] . '\' is already taken.
';
        }

        if(preg_match("/\\s/", $_POST['username']) == true){
            $errors[] = 'Your username must not contain any spaces.';
        }

        if(strlen($_POST['password']) < 6){
            $errors[] = 'Your password must be atleast 6 characters.';
        }

        if($_POST['password'] != $_POST['confirm_password']){
            $errors[] = 'Your passwords do not match';
        }

        if(filter_var($_POST['email'], FILTER_VALIDATE_EMAIL) === false){
            $errors[] = 'A valid email address is required';
        }

        if(email_exists($_POST['email']) === true){
            $errors[] = 'Sorry, the email \'' . $_POST['email'] . '\' is already in use';
        }

    }
}
?>

<?php

    if(empty($_POST) === false && empty($errors) === true){
        $register_data = array (
            'username' =>  $_POST['username'],
            'password' =>   $_POST['password'],
            'first_name' => $_POST['first_name'],
            'last_name' => $_POST['last_name'],
            'email' =>  $_POST['email']
        );

        register_user($register_data);
        //header('location: index.php?page=register');
        exit();
    } else {

?>
    <div class="main-content">
        <div class="form-container">
            <div class="form-inner">
                <ul>
                    <li><h2>Register</h2></li>
                    <li>Register your account</li>

                    <?php
                    if(empty($errors) === false){
                        echo output_errors($errors);
                    }

                    if(isset($_GET['success']) && !empty($_GET['success'])){
                        echo "<li class='errorOk'>You've been registerd succesfully!</li>";
                    }
                    ?>

                    <li><form action="" method="POST" id="reg-form"></li>
                    <li><input name="username" type="text" placeholder="Username" required></
li>
                    <li><input name="password" type="password" placeholder="Password" require
d></li>
                    <li><input name="confirm_password" type="password" placeholder="Confirm p
assword" required></li>
                    <li><input name="first_name" type="text" placeholder="First name"></li>
                    <li><input name="last_name" type="text" placeholder="Last name"></li>
                    <li><input name="email" type="text" placeholder="Email address" required>
</li>
                    <li>
                        <button name="register" type="submit" class="btn-
reg">Create account</button>
                        </form>
                    </li>
                </ul>
            </div>
            <div class="form-inner">
                <ul>
                    <li class="line"></li>
                    <li>Having issues with your account?</li>
                    <li><a href="index.php?page=login">I already have an account, send me bac
k to login panel.</a></li>
                </ul>
            </div>
        </div>
        <?php
        }
        ?>
    </div>
</div>
```

page-stats.php

```
<?php
if(isset($_POST['register'])){

    if(empty($errors) === true ){
        if(user_exists($_POST['username']) === true){
            $errors[] = 'Sorry, the username \'' . $_POST['username'] . '\' is already taken.
';
        }

        if(preg_match("/\\s/", $_POST['username']) == true){
            $errors[] = 'Your username must not contain any spaces.';
        }

        if(strlen($_POST['password']) < 6){
            $errors[] = 'Your password must be atleast 6 characters.';
        }

        if($_POST['password'] != $_POST['confirm_password']){
            $errors[] = 'Your passwords do not match';
        }

        if(filter_var($_POST['email'], FILTER_VALIDATE_EMAIL) === false){
            $errors[] = 'A valid email address is required';
        }

        if(email_exists($_POST['email']) === true){
            $errors[] = 'Sorry, the email \'' . $_POST['email'] . '\' is already in use';
        }

    }
}
?>

<?php

    if(empty($_POST) === false && empty($errors) === true){
        $register_data = array (
            'username' =>  $_POST['username'],
            'password' =>   $_POST['password'],
```



```
            'first_name' => $_POST['first_name'],
            'last_name' => $_POST['last_name'],
            'email' =>  $_POST['email']
        );

        register_user($register_data);
        //header('location: index.php?page=register');
        exit();
    } else {

?>
    <div class="main-content">
        <div class="form-container">
            <div class="form-inner">
                <ul>
                    <li><h2>Register</h2></li>
                    <li>Register your account</li>

                    <?php
                    if(empty($errors) === false){
                        echo output_errors($errors);
                    }

                    if(isset($_GET['success']) && !empty($_GET['success'])){
                        echo "<li class='errorOk'>You've been registerd succesfully!</li>";
                    }
                    ?>

                    <li><form action="" method="POST" id="reg-form"></li>
                    <li><input name="username" type="text" placeholder="Username" required></
li>
                    <li><input name="password" type="password" placeholder="Password" require
d></li>
                    <li><input name="confirm_password" type="password" placeholder="Confirm p
assword" required></li>
                    <li><input name="first_name" type="text" placeholder="First name"></li>
                    <li><input name="last_name" type="text" placeholder="Last name"></li>
                    <li><input name="email" type="text" placeholder="Email address" required>
</li>
                    <li>
                        <button name="register" type="submit" class="btn-
reg">Create account</button>
                        </form>
                    </li>
                </ul>
            </div>
            <div class="form-inner">
                <ul>
```

```
                    <li class="line"></li>
                    <li>Having issues with your account?</li>
                    <li><a href="index.php?page=login">I already have an account, send me back to login panel.</a></li>
                </ul>
            </div>
        </div>

        <?php
        }
        ?>
    </div>
</div>
```

add-categories.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['category_name'])){
    require_once '../../config/connect.php';

    $name = $_POST['category_name'];


    $q = $conn->prepare("INSERT INTO categories (category_name) VALUES (:name)");

    try {

        $q->execute(array(
            ':name' => $name
        ));

        http_response_code(201);
        echo json_encode(["msg"=> "Successufully added category!"]);

    }
    catch(PDOException $ex){
        echo json_encode(['error'=> 'Problem with database: ' . $ex->getMessage()]);

        include_once '../functions/functions.php';
        log_errors($ex->getMessage());

        http_response_code(500);
    }
} else {
    http_response_code(400);
}
```

delete-categories.php

```
<?php

if(isset($_POST['id'])){
    $id = $_POST['id'];

    include "../../config/connect.php";
    $delete = "DELETE FROM categories WHERE category_id = :id";
    $q = $conn->prepare($delete);
    $q->execute(array(
        ':id' => $id
    ));
    $query = "SELECT * FROM categories";
    $cats = executeQuery($query);
    echo json_encode($cats);
} else {
    http_response_code(400);
    echo json_encode(["error"=> "No category_id parameter sent!"]);
}
```

get-categories-opt.php

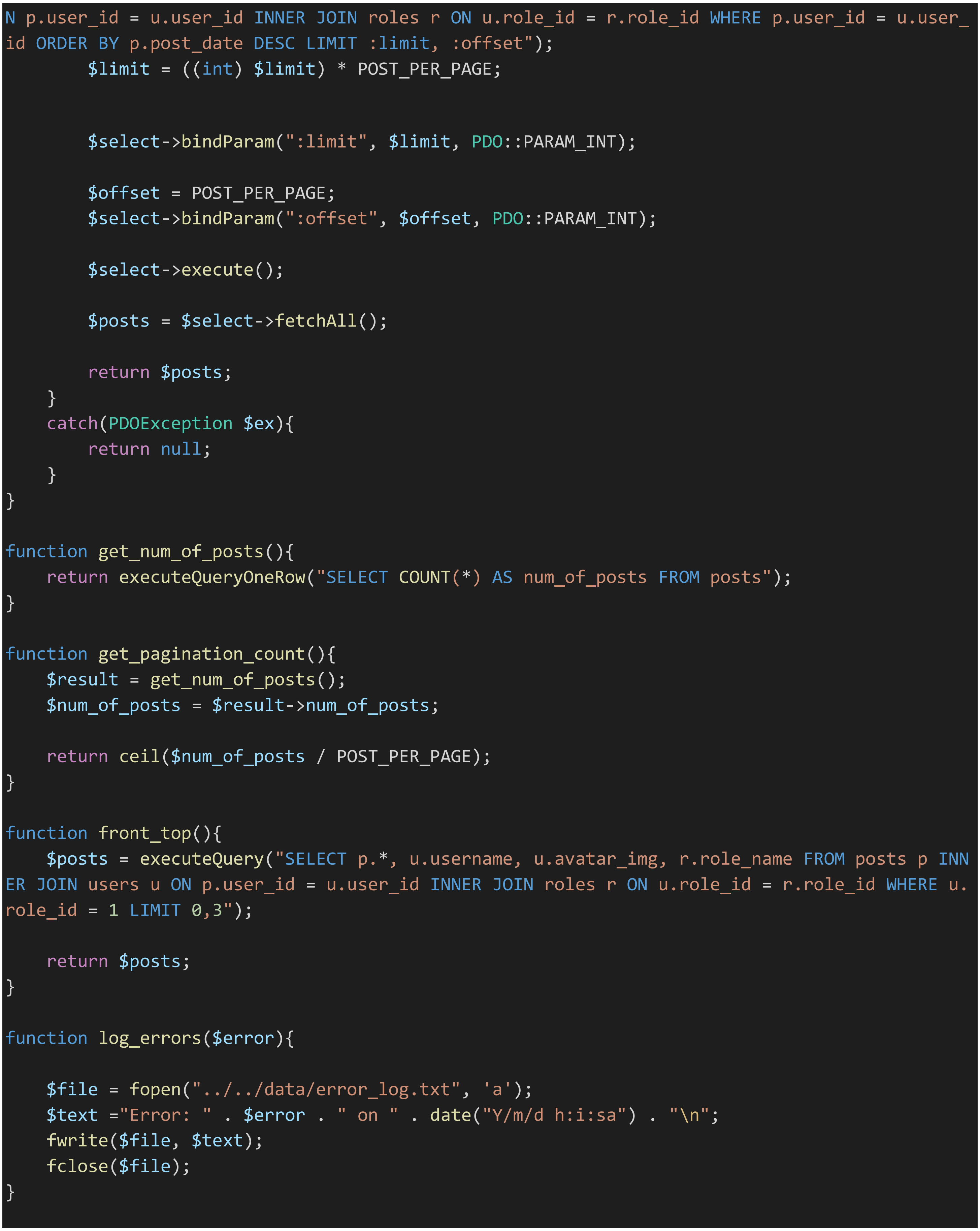
```
<?php

header('Content-Type: application/json');

require_once '../../config/connect.php';

$cats = executeQuery("SELECT * FROM categories");
echo json_encode($cats);
```

contact_validation.php

```
<?php

session_start();


if(isset($_POST['conName']) && isset($_POST['conEmail']) && isset($_POST['message'])){

    $name = $_POST['conName'];
    $email = $_POST['conEmail'];
    $message = $_POST['message'];
    $regexEmail = "/^\w+[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/";
    $regexName = "/^[A-Z][a-z]+(([',. -][A-Z][a-z])?[a-zA-Z]*)*$/";

    $errors = [];

    if(!preg_match($regexName, $name)){
        $errors[] = "Name must start with uppercase";
    }

    if(!preg_match($regexEmail, $email)){
        $errors[] = "Bad email format";
    }

    if(count($errors) == 0) {

        $content="From: $name \n Email: $email \n Message: $message";
        $recipient = "x4blood87@gmail.com";
        $mailheader = "From: $email \r\n";
        mail($recipient, $content, $mailheader) or die("Error!");

        http_response_code(201);
        echo json_encode(["msg"=> "Successufully added user!"]);

    } else {

        echo json_encode(["msg"=> 'Nothing sent!']);
    }
}
```

functions.php

```
<?php

define("POST_PER_PAGE", 5);

function get_categories(){
    $categories = executeQuery("SELECT * FROM categories");
    return $categories;
}

function get_posts($limit = 0){
    global $conn;
    try{

        $select = $conn-
>prepare("SELECT p.*, u.username, u.avatar_img, r.role_name FROM posts p INNER JOIN users u O
```



```
N p.user_id = u.user_id INNER JOIN roles r ON u.role_id = r.role_id WHERE p.user_id = u.user_
id ORDER BY p.post_date DESC LIMIT :limit, :offset");
        $limit = ((int) $limit) * POST_PER_PAGE;


        $select->bindParam(":limit", $limit, PDO::PARAM_INT);

        $offset = POST_PER_PAGE;
        $select->bindParam(":offset", $offset, PDO::PARAM_INT);

        $select->execute();

        $posts = $select->fetchAll();

        return $posts;
    }
    catch(PDOException $ex){
        return null;
    }
}

function get_num_of_posts(){
    return executeQueryOneRow("SELECT COUNT(*) AS num_of_posts FROM posts");
}

function get_pagination_count(){
    $result = get_num_of_posts();
    $num_of_posts = $result->num_of_posts;

    return ceil($num_of_posts / POST_PER_PAGE);
}

function front_top(){
    $posts = executeQuery("SELECT p.*, u.username, u.avatar_img, r.role_name FROM posts p INN
ER JOIN users u ON p.user_id = u.user_id INNER JOIN roles r ON u.role_id = r.role_id WHERE u.
role_id = 1 LIMIT 0,3");

    return $posts;
}

function log_errors($error){

    $file = fopen("../../data/error_log.txt", 'a');
    $text ="Error: " . $error . " on " . date("Y/m/d h:i:sa") . "\n";
    fwrite($file, $text);
    fclose($file);
}
```

general.php

```php
<?php

function output_errors($errors){
    return "<ul <li class='error'>" . implode('</li><li>', $errors) . "</li></ul>";
}

?>
```

get-links.php

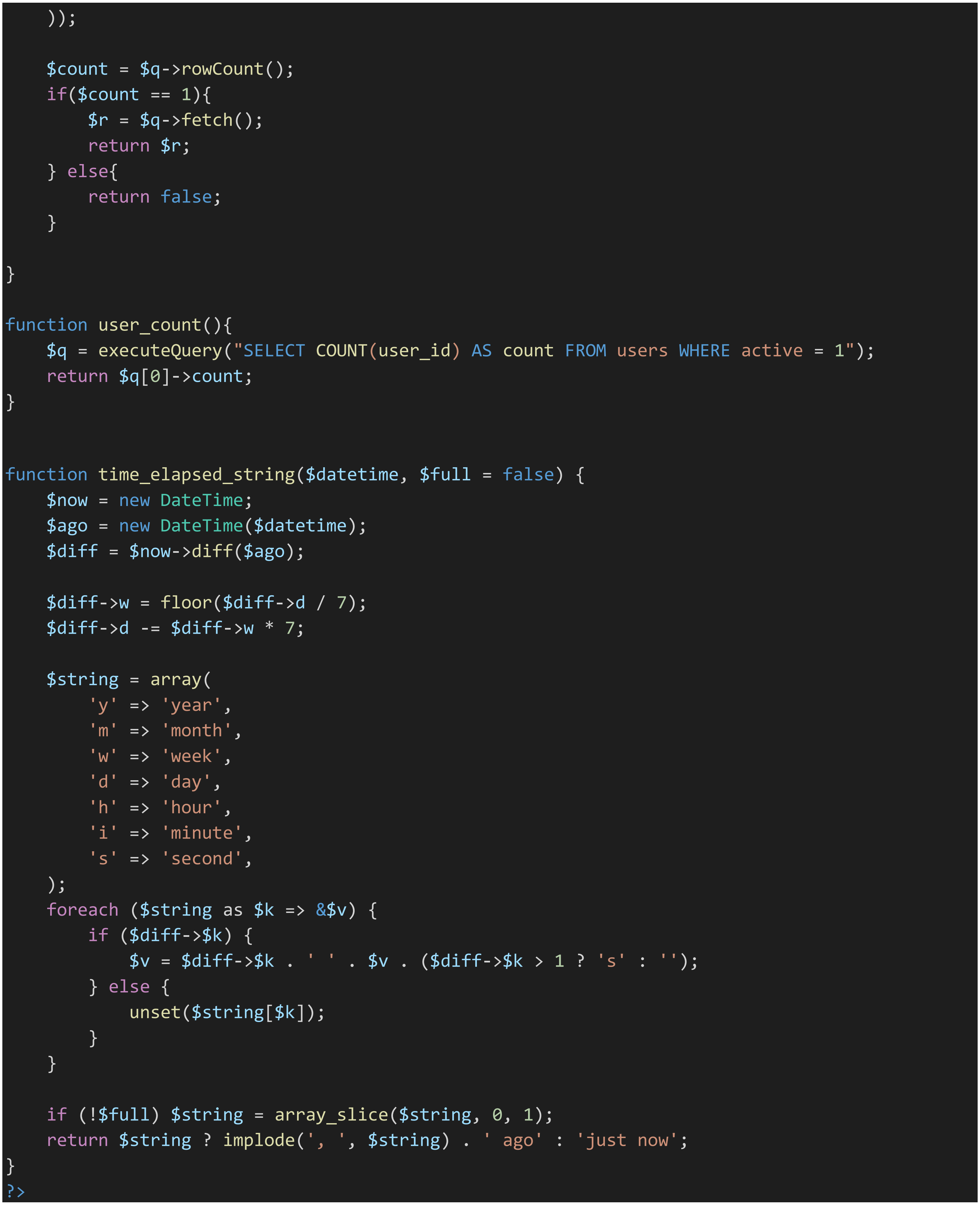
```php
<?php

function get_menu(){
    $menu = executeQuery("SELECT * FROM menu");
    return $menu;
}
```

register_activation.php

```php
<?php

session_start();

if(isset($_GET['code'])){
    $code = $_GET['code'];

    require "../../config/connect.php";

    $sql = "SELECT * FROM users WHERE active_code = :code";
    $stmt = $conn->prepare($sql);
    $stmt->bindParam(":code", $code);

    $stmt->execute();

    if($stmt->rowCount() == 1){
        $user = $stmt->fetch();

        $update = $conn->prepare("UPDATE users SET active = 1 WHERE active_code = :code");
        $update->bindParam(":code", $code);
        $update->execute();

        $_SESSION['success'] = "You have successfully been activated, welcome!";

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

        include_once 'functions.php';
        log_errors("WARNING: wrong activation code used!");

        $_SESSION['error'] = "Wrong activation code!";
        header("Location: ../../index.php?page=login");
    }


}
```

users.php

```
<?php

function register_user($register_data){
    global $conn;
    $username = $register_data['username'];
    $password = md5($register_data['password']);
    $first_name = $register_data['first_name'];
    $last_name = $register_data['last_name'];
    $email = $register_data['email'];

    $active = 0;

    // radjeno kao na vezbama sa primerom konstante
    define("USER_ROLE", 2);

    $role = USER_ROLE;

    // AKTIVACIONI KOD - prepisano sa vezbi
    $code = sha1(md5(md5(time().md5($email).rand(1, 10000000))));

    $sql = "INSERT INTO users (username, password, first_name, last_name, email, active, role_id, active_code) VALUES (:username, :password, :first_name, :last_name, :email, :active, :role, :code)";
    $q = $conn->prepare($sql);

    try{

        $q->execute(array(
            ':username' => $username,
            ':password' => $password,
            ':first_name' => $first_name,
            ':last_name' => $last_name,
            ':email' => $email,
            ':active' => $active,
            ':role' => $role,
            ':code' => $code
        ));
        $_SESSION['success'] = "You have successfully been registered";

        // SLANJE EMAIL-a - prepisano za vezbi
       // mail($email, "Registracija", "Potvrdite registraciju: http://localhost/php1_sajt/lo
gic/register_activation.php?code=".$code);

        // ILI AKO NE RADI MAIL, GADJATI DIREKTNO URL:
        header("location: models/functions/register_activation.php?code=" . $code );

    }catch(PDOException $ex){

        include_once '../functions/functions.php';
        log_errors($ex->getMessage());
    }

}

function logged_in(){
    if(isset($_SESSION['user_id'])){
        return true;
    } else {
        return false;
    }
}


function user_exists($username){
    global $conn;
    $sql="SELECT * FROM users WHERE username = :username";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':username' => $username,
    ));
    $count = $q->rowCount();
    if($count == 1){
        return true;
    } else {
        return false;
    }
}

function email_exists($email){
    global $conn;
    $sql ="SELECT * FROM users WHERE email = :email";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':email' => $email
    ));
    $count = $q->rowCount();
    if($count == 1){
        return true;
    }   else {
        return false;
    }
}

function user_active($email){
    global $conn;
    $sql="SELECT * FROM users WHERE email = :email AND active = 1";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':email' => $email
    ));

    $count = $q->rowCount();
    if($count == 1){
        return true;
    } else{
        return false;
    }
}

function user_id_from_username($email){
    global $conn;
    $sql="SELECT user_id FROM users WHERE email = :email";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':email' => $email
    ));
    $r = $q->fetch();
    return $r->user_id;
}

function login($email, $password){
    global $conn;
    $user_id = user_id_from_username($email);
    $password = md5($password);

    $sql="SELECT user_id, u.username, u.first_name, u.last_name, u.email, u.role_id, u.avatar_img, r.role_name FROM users u INNER JOIN roles r ON u.role_id = r.role_id WHERE email = :email AND password = :password";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':email' => $email,
        'password' => $password
```



```
    ));

    $count = $q->rowCount();
    if($count == 1){
        $r = $q->fetch();
        return $r;
    } else{
        return false;
    }

}

function user_count(){
    $q = executeQuery("SELECT COUNT(user_id) AS count FROM users WHERE active = 1");
    return $q[0]->count;
}


function time_elapsed_string($datetime, $full = false) {
    $now = new DateTime;
    $ago = new DateTime($datetime);
    $diff = $now->diff($ago);

    $diff->w = floor($diff->d / 7);
    $diff->d -= $diff->w * 7;

    $string = array(
        'y' => 'year',
        'm' => 'month',
        'w' => 'week',
        'd' => 'day',
        'h' => 'hour',
        'i' => 'minute',
        's' => 'second',
    );
    foreach ($string as $k => &$v) {
        if ($diff->$k) {
            $v = $diff->$k . ' ' . $v . ($diff->$k > 1 ? 's' : '');
        } else {
            unset($string[$k]);
        }
    }

    if (!$full) $string = array_slice($string, 0, 1);
    return $string ? implode(', ', $string) . ' ago' : 'just now';
}
?>
```

add-post.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['user_id'],$_POST['category_id'],$_POST['title'],$_POST['content'])){
    require_once '../../config/connect.php';

    $user_id = $_POST['user_id'];
    $cat_id = $_POST['category_id'];
    $title = $_POST['title'];
    $content = $_POST['content'];


    $q = $conn-
>prepare("INSERT INTO posts (title, content, user_id) VALUES (:title, :content, :user_id)");



    try {

        if($q->execute( array(
            ':title' => $title,
            ':content' => $content,
            ':user_id' => $user_id
        )) === true){
            $last_id = $conn->lastInsertId();
        }
        $cat_post = $conn-
>prepare("INSERT INTO category_post (post_id, category_id) VALUES (:post_id, :category_id)");
        $cat_post->execute( array(
            ':post_id' => $last_id,
            ':category_id' => $cat_id

        ));

        http_response_code(201);
        echo json_encode(["msg"=> "Successufully added post!"]);

    }
    catch(PDOException $ex){
        echo json_encode(['error'=> 'Problem with database: ' . $ex->getMessage()]);

        include_once '../functions.php';
        log_errors($ex->getMessage());

        http_response_code(500);
    }
} else {
    http_response_code(400);
}
```

delete-post.php

```
<?php

if(isset($_POST['post_id']) && isset($_POST['user_id'])){
    $post_id = $_POST['post_id'];
    $user_id = $_POST['user_id'];

    include "../../config/connect.php";
    $delete = "DELETE FROM posts WHERE post_id = :post_id";
    $q = $conn->prepare($delete);
    $q->execute(array(
        ':post_id' => $post_id
    ));
    $query = "SELECT p.post_id, p.title, p.post_date, c.category_name AS cat_name FROM posts p INNER JOIN category_post cp ON p.post_id = cp.post_id INNER JOIN categories c ON cp.category_id = c.category_id WHERE p.user_id ='$user_id'";
    $posts = executeQuery($query);
    echo json_encode($posts);
} else {
    http_response_code(400);
    echo json_encode(["error"=> "No user_id parameter sent!"]);
}
```

get-post.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['post_id'])){
    $post_id = $_POST['post_id'];

    include "../../config/connect.php";


    try
    {
        $query = "SELECT p.*, c.category_id FROM posts p INNER JOIN category_post cp ON p.post_id = cp.post_id INNER JOIN categories c ON cp.category_id = c.category_id WHERE p.post_id = :post_id";
        $q = $conn->prepare($query);
        $q->execute(array(
            ':post_id' => $post_id
        ));

        $post_data = $q->fetchAll();
        $cat_data = executeQuery("SELECT * FROM categories");
        $data = array('post' => $post_data, 'cats' => $cat_data);
        echo json_encode($data);
    }
    catch(PDOException $ex){
        echo json_encode(['error'=> 'Problem with database: ' . $ex->getMessage()]);

        include_once '../functions/functions.php';
        log_errors($ex->getMessage());

        http_response_code(500);
    }

} else {
    http_response_code(400);
    echo json_encode(["error"=> "No post_id parameter sent!"]);
}
```

get-posts.php

```
<?php

header('Content-Type: application/json');

require_once '../../config/connect.php';

$posts = executeQuery("SELECT p.*, u.username, u.avatar_img, r.role_name FROM posts p INNER JOIN users u ON p.user_id = u.user_id INNER JOIN roles r ON u.role_id = r.role_id WHERE p.user_id = u.user_id");

echo json_encode($posts);
```

get-user-posts.php

```
<?php

if(isset($_POST['user_id'])){
    $user_id = $_POST['user_id'];

    include "../../config/connect.php";
    $query = "SELECT p.user_id, p.post_id, p.title, p.post_date, c.category_name AS cat_name
FROM posts p INNER JOIN category_post cp ON p.post_id = cp.post_id INNER JOIN categories c ON
 cp.category_id = c.category_id WHERE p.user_id ='$user_id'";
    $posts = executeQuery($query);
    echo json_encode($posts);
} else {
    http_response_code(400);
    echo json_encode(["error"=> "No user_id parameter sent!"]);
}
```

post-pagination.php

```
<?php

header("Content-Type: application/json");

if(isset($_GET['limit'])){
    require_once "../../config/connect.php";
    include "../functions/functions.php";

    $limit = $_GET['limit'];
    $posts = get_posts($limit);
    $num_of_pages = get_pagination_count();

    echo json_encode([
        "posts" => $posts,
        "num_of_pages" => $num_of_pages
    ]);
} else {
    echo json_encode(["message"=> "Limit not passed."]);
    http_response_code(400);
}
```

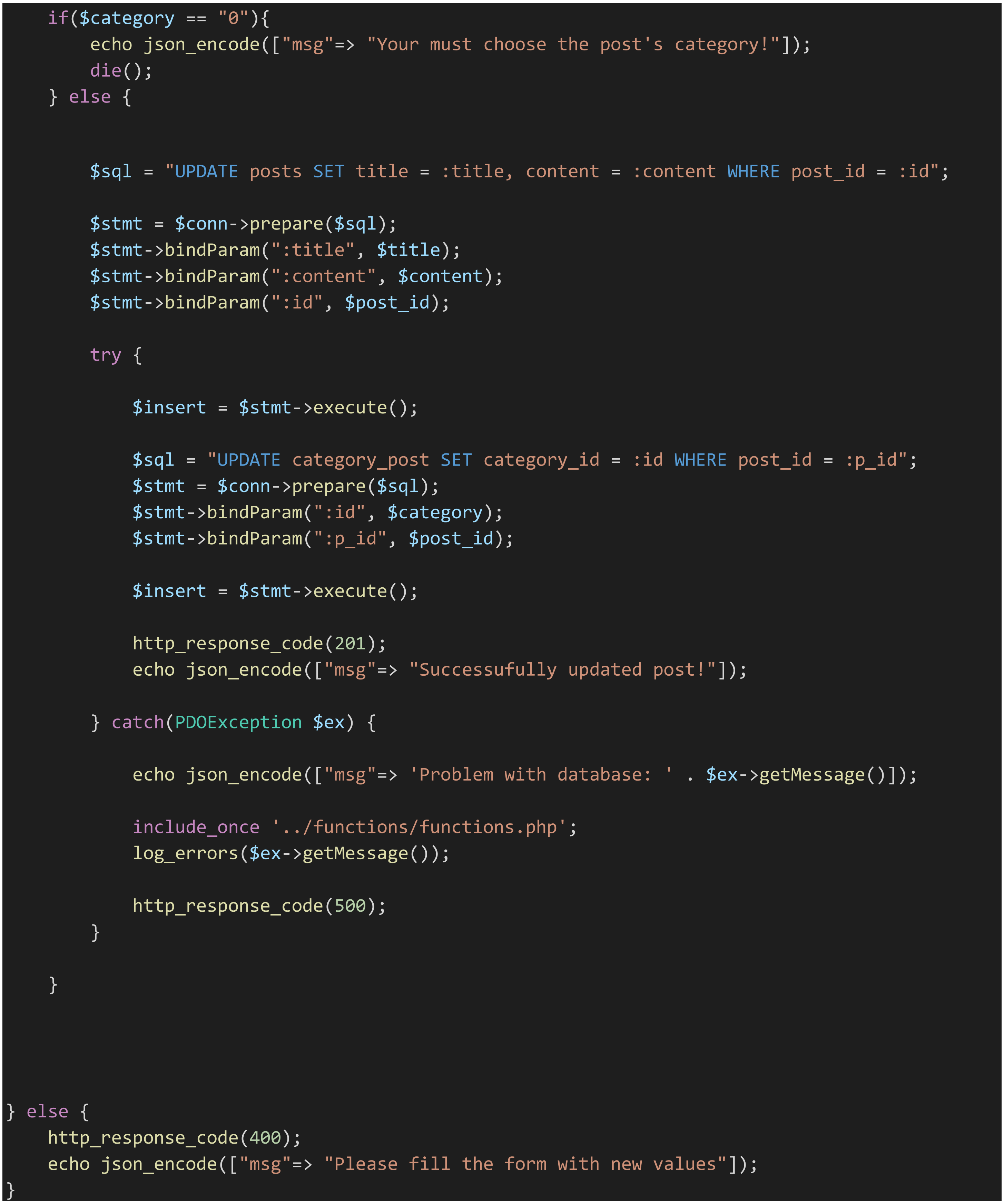
sort-posts.php

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

if(isset($_POST['sort'])){
    $sort = $_POST['sort'];

    include "../../config/connect.php";

    $query = "SELECT p.*, u.username, u.avatar_img, r.role_name, c.category_name FROM posts p
 INNER JOIN users u ON p.user_id = u.user_id INNER JOIN roles r ON u.role_id = r.role_id INNE
R JOIN category_post cp on p.post_id = cp.post_id INNER JOIN categories c on cp.category_id =
 c.category_id";

```

```
    switch($sort){
        case "desc":
            $query .= " WHERE p.user_id = u.user_id ORDER BY post_date DESC";
            break;
        case "asc":
            $query .= " WHERE p.user_id = u.user_id ORDER BY post_date ASC";
            break;
        case 1:
            $query .= " WHERE c.category_id = 1";
            break;
        case 2:
            $query .= " WHERE c.category_id = 2";
            break;
        case 3:
            $query .= " WHERE c.category_id = 3";
            break;
        case 4:
            $query .= " WHERE c.category_id = 4";
            break;
        case 5:
            $query .= " WHERE c.category_id = 5";
            break;

    }

    $posts = executeQuery($query);

    echo json_encode($posts);
} else {
    http_response_code(400); // Bad request
    echo json_encode(["error"=> "No sort parameter sent!"]);
}
```

update-post.php

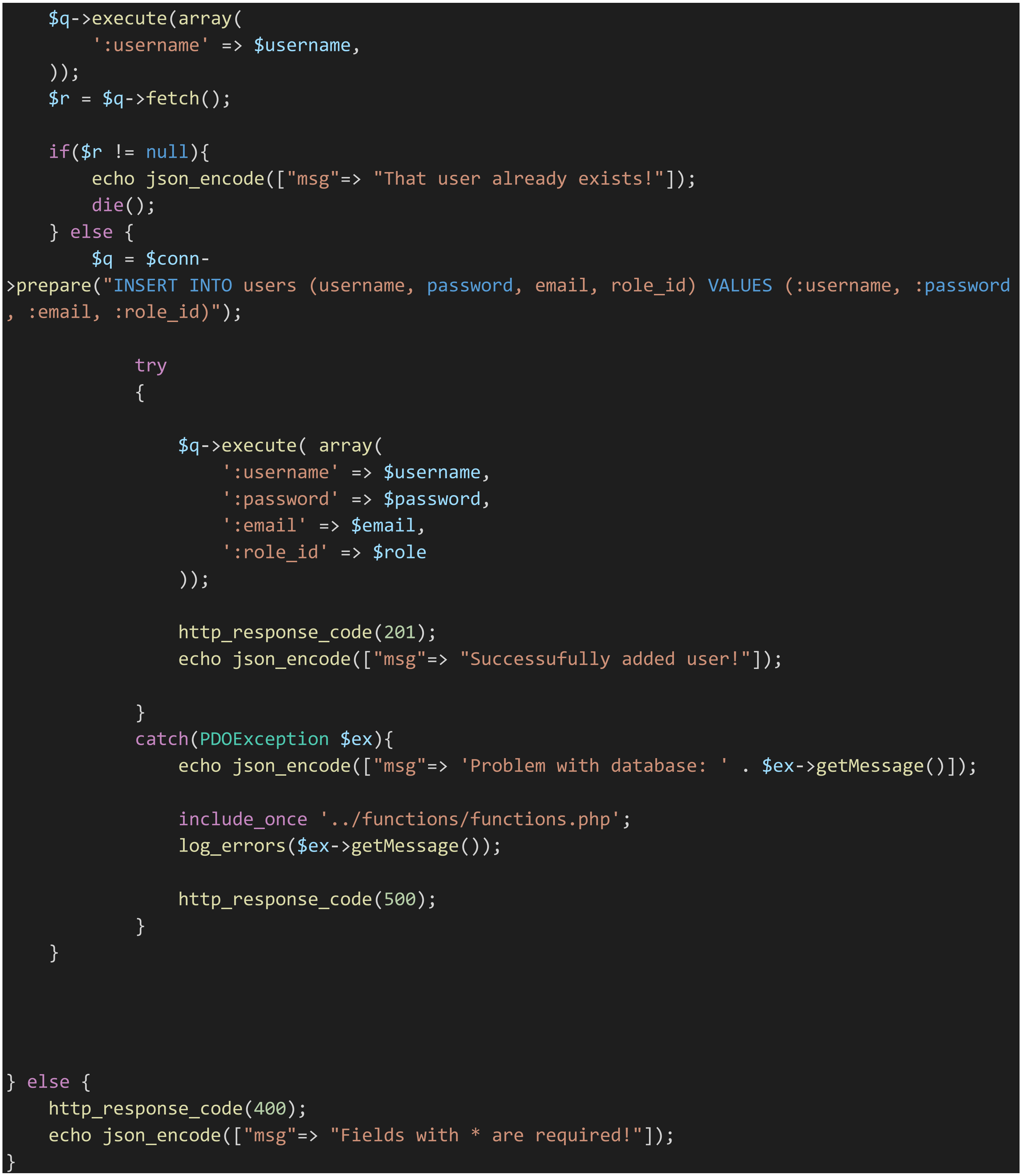
```
<?php

header('Content-Type: application/json');

if(isset($_POST['post_id'],$_POST['title'],$_POST['content'],$_POST['category'])){

    require_once '../../config/connect.php';

    $post_id = $_POST['post_id'];
    $title = $_POST['title'];
    $content = $_POST['content'];
    $category = $_POST['category'];

```



```
    if($category == "0"){
        echo json_encode(["msg"=> "Your must choose the post's category!"]);
        die();
    } else {


        $sql = "UPDATE posts SET title = :title, content = :content WHERE post_id = :id";

        $stmt = $conn->prepare($sql);
        $stmt->bindParam(":title", $title);
        $stmt->bindParam(":content", $content);
        $stmt->bindParam(":id", $post_id);

        try {

            $insert = $stmt->execute();

            $sql = "UPDATE category_post SET category_id = :id WHERE post_id = :p_id";
            $stmt = $conn->prepare($sql);
            $stmt->bindParam(":id", $category);
            $stmt->bindParam(":p_id", $post_id);

            $insert = $stmt->execute();

            http_response_code(201);
            echo json_encode(["msg"=> "Successufully updated post!"]);

        } catch(PDOException $ex) {

            echo json_encode(["msg"=> 'Problem with database: ' . $ex->getMessage()]);

            include_once '../functions/functions.php';
            log_errors($ex->getMessage());

            http_response_code(500);
        }

    }




} else {
    http_response_code(400);
    echo json_encode(["msg"=> "Please fill the form with new values"]);
}
```

add-user.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['username'],$_POST['password'],$_POST['confirm_password'],$_POST['email'],$_P
OST['role'])){

    require_once '../../config/connect.php';

    $username = $_POST['username'];
    $password = $_POST['password'];
    $confirm_password = $_POST['confirm_password'];
    $email = $_POST['email'];
    $role = $_POST['role'];

    if(strlen($password) < 6){
        echo json_encode(["msg"=> "Your password must be atleast 6 characters!"]);
        die();
    }

    if($password == $confirm_password){
        $password = md5($password);
    }else{
        echo json_encode(["msg"=> "Your password's dont match!"]);
        die();
    }

    if($role == "0"){
        echo json_encode(["msg"=> "Your must choose the user's role!"]);
        die();
    }

    $sql="SELECT * FROM users WHERE email = :email";
    $q = $conn->prepare($sql);
    $q->execute(array(
        ':email' => $email,
    ));
    $r = $q->fetch();

    if($r != null){
        echo json_encode(["msg"=> "That email already exists!"]);
        die();
    }


    $sql="SELECT * FROM users WHERE username = :username";
    $q = $conn->prepare($sql);
```



```
    $q->execute(array(
        ':username' => $username,
    ));
    $r = $q->fetch();

    if($r != null){
        echo json_encode(["msg"=> "That user already exists!"]);
        die();
    } else {
        $q = $conn-
>prepare("INSERT INTO users (username, password, email, role_id) VALUES (:username, :password
, :email, :role_id)");

            try
            {

                $q->execute( array(
                    ':username' => $username,
                    ':password' => $password,
                    ':email' => $email,
                    ':role_id' => $role
                ));

                http_response_code(201);
                echo json_encode(["msg"=> "Successufully added user!"]);

            }
            catch(PDOException $ex){
                echo json_encode(["msg"=> 'Problem with database: ' . $ex->getMessage()]);

                include_once '../functions/functions.php';
                log_errors($ex->getMessage());

                http_response_code(500);
            }
    }



} else {
    http_response_code(400);
    echo json_encode(["msg"=> "Fields with * are required!"]);
}
```

delete-user.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['user_id'])){
    $user_id = $_POST['user_id'];

    include "../../config/connect.php";
    $delete = "DELETE FROM users WHERE user_id = :user_id";
    $q = $conn->prepare($delete);
    try{
        $q->execute(array(
        ':user_id' => $user_id
    ));

    $query = "SELECT u.user_id, u.username, u.first_name, u.last_name, u.email, r.role_name F
ROM users u INNER JOIN roles r ON u.role_id = r.role_id";
    $posts = executeQuery($query);
    echo json_encode($posts);
    }
    catch(PDOException $ex){
        echo json_encode(['error'=> 'Problem with database: ' . $ex->getMessage()]);

        include_once '../functions/functions.php';
        log_errors($ex->getMessage());

        http_response_code(500);
    }

} else {
    http_response_code(400);
    echo json_encode(["error"=> "No user_id parameter sent!"]);
}
```

get-roles.php

```
<?php

header('Content-Type: application/json');

require_once '../../config/connect.php';

$roles = executeQuery("SELECT * FROM roles");
echo json_encode($roles);
```

get-user.php

```php
<?php

header('Content-Type: application/json');

if(isset($_POST['user_id'])){
    $user_id = $_POST['user_id'];

    include "../../config/connect.php";

    try
    {
        $query = "SELECT u.user_id, u.username, u.first_name, u.last_name, u.email, r.role_id, r.role_name FROM users u INNER JOIN roles r ON u.role_id = r.role_id WHERE user_id = :user_id";
        $q = $conn->prepare($query);

        $q->execute(array(
            ':user_id' => $user_id
        ));

        $user_data = $q->fetchAll();
        $roles = executeQuery("SELECT * FROM roles");

        $data = array('user' => $user_data, 'roles' => $roles);

        echo json_encode($data);
    }
    catch(PDOException $ex){
        echo json_encode(['error'=> 'Problem with database: ' . $ex->getMessage()]);

        include_once '../functions/functions.php';
        log_errors($ex->getMessage());

        http_response_code(500);
    }

} else {
    http_response_code(400);
    echo json_encode(["error"=> "No user_id parameter sent!"]);
}
```

get-users.php

```php
<?php

header('Content-Type: application/json');
```

```
require_once '../../config/connect.php';

$users = executeQuery("SELECT u.user_id, u.username, u.first_name, u.last_name, u.email, r.role_name FROM users u INNER JOIN roles r ON u.role_id = r.role_id");
echo json_encode($users);
```

update-user.php

```
<?php

header('Content-Type: application/json');

if(isset($_POST['user_id'],$_POST['username'],$_POST['password'],$_POST['confirm_password'],$_POST['email'],$_POST['role'])){

    require_once '../../config/connect.php';

    $user_id = $_POST['user_id'];
    $username = $_POST['username'];
    $password = $_POST['password'];
    $confirm_password = $_POST['confirm_password'];
    $first_name = $_POST['first_name'];
    $last_name = $_POST['last_name'];
    $email = $_POST['email'];
    $role = $_POST['role'];

    if(strlen($password) < 6){
        echo json_encode(["msg"=> "Your password must be atleast 6 characters!"]);
        die();
    }

    if($password == $confirm_password){
        $password = md5($password);
    }else{
        echo json_encode(["msg"=> "Your password's dont match!"]);
        die();
    }

    if($role == "0"){
        echo json_encode(["msg"=> "Your must choose the user's role!"]);
        die();
    } else {


        $q = $conn->prepare("UPDATE users SET username = :username, password = :password, first_name = :first_name, last_name = :last_name, email = :email , role_id = :role_id WHERE user_id = :user_id");
```

```
            try
            {

                $q->execute( array(
                    ':user_id' => $user_id,
                    ':username' => $username,
                    ':password' => $password,
                    ':first_name' => $first_name,
                    ':last_name' => $last_name,
                    ':email' => $email,
                    ':role_id' => $role
                ));

                http_response_code(201);
                echo json_encode(["msg"=> "Successufully updated user!"]);

            }
            catch(PDOException $ex){
                echo json_encode(["msg"=> 'Problem with database: ' . $ex->getMessage()]);

                include_once '../functions/functions.php';
                log_errors($ex->getMessage());

                http_response_code(500);
            }
    }



} else {
    http_response_code(400);
    echo json_encode(["msg"=> "Please fill the form with new values"]);
}
```

config.php

```
<?php
define("ABSOLUTE_PATH", $_SERVER["DOCUMENT_ROOT"]."/pract_sajt");

define("ENV_FAJL", ABSOLUTE_PATH."/config/_env.env");
define("LOG_FAJL", ABSOLUTE_PATH."/data/log.txt");

define("SERVER", env("SERVER"));
define("DATABASE", env("DBNAME"));
define("USERNAME", env("USERNAME"));
define("PASSWORD", env("PASSWORD"));
```

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

connect.php

```
<?php

 require_once 'config.php';

 belezi_pristup();

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

        echo "Something went wrong!";

    }

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

    function executeQueryOneRow($query){
        global $conn;
        return $conn->query($query)->fetch();
    }

    function belezi_pristup(){
        $open = fopen(LOG_FAJL, "a");
        if($open) {
            if(empty($_SESSION['user'])){
                $user = "Guest";
            }
            else {
                $user = $_SESSION['user']->username;
            }
            $date = date('d-m-Y H:i:s');
            fwrite($open, "{$_SERVER['PHP_SELF']}\t{$user}\t{$date}\t{$_SERVER['REMOTE_ADDR']
}\t\n");
            fclose($open);
        }
    }

?>
```

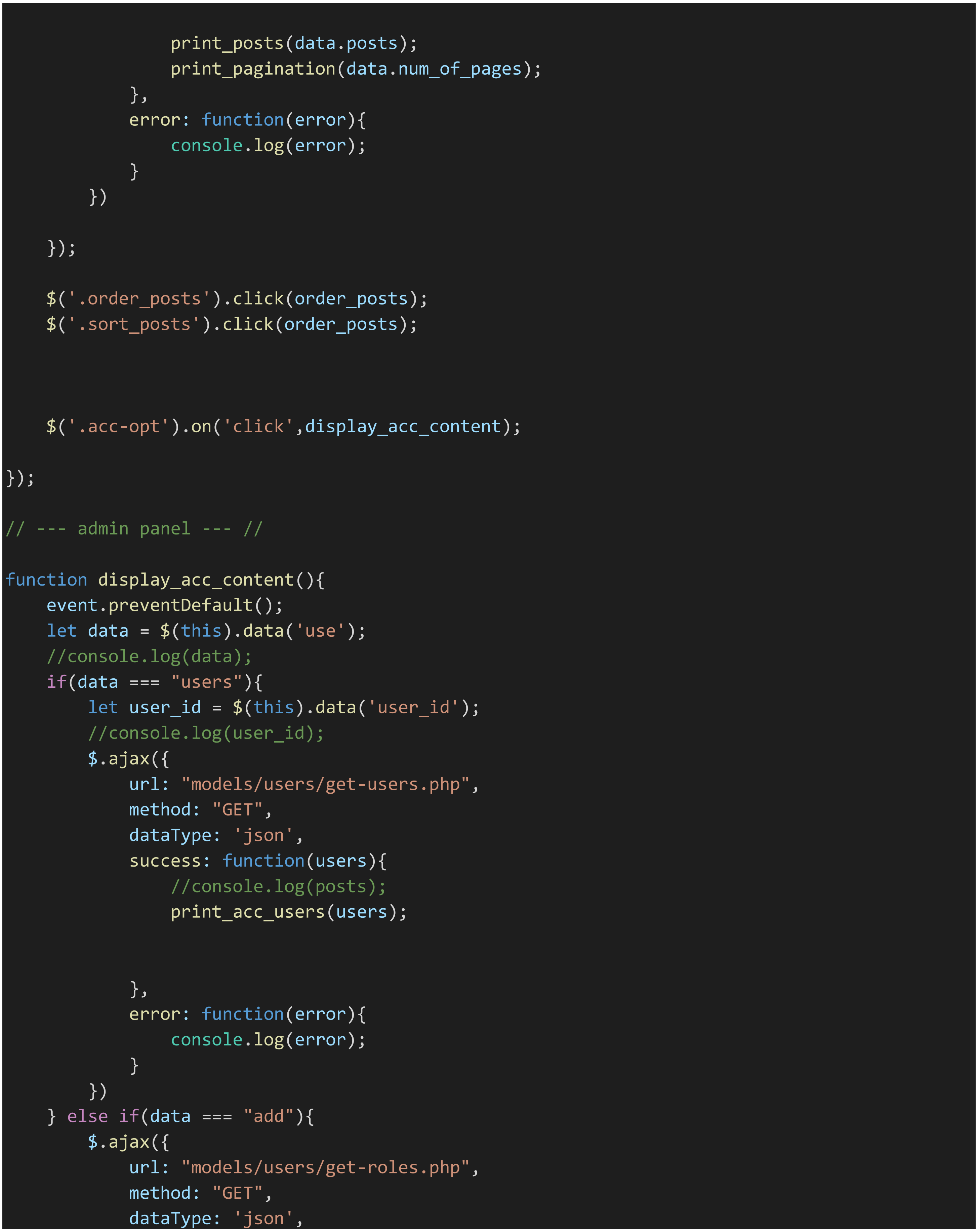
init.php

```
<?php
session_start();

require 'connect.php';
require 'models/functions/general.php';
require 'models/functions/users.php';




$errors = array();
?>
```

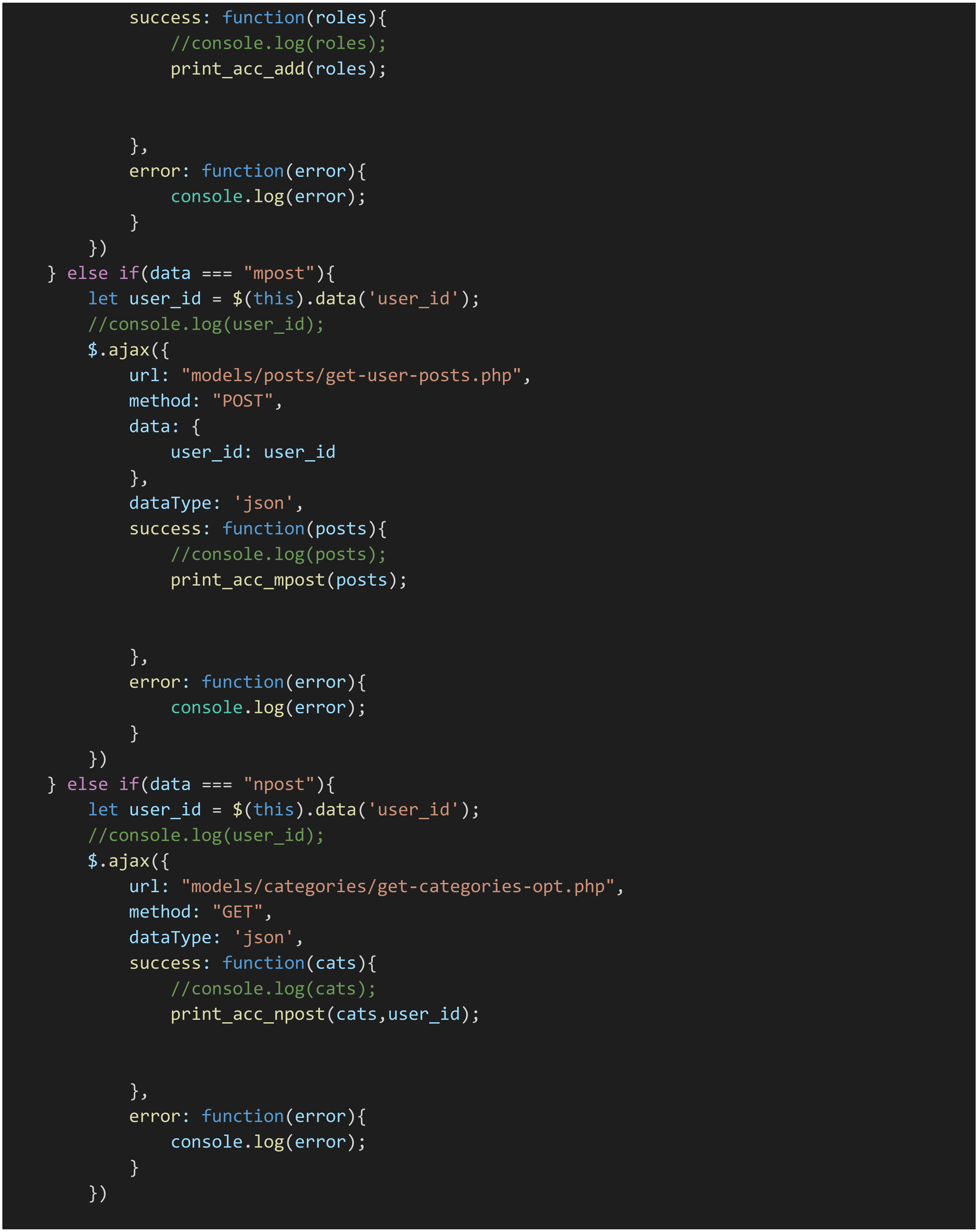
JavaScript

main.js

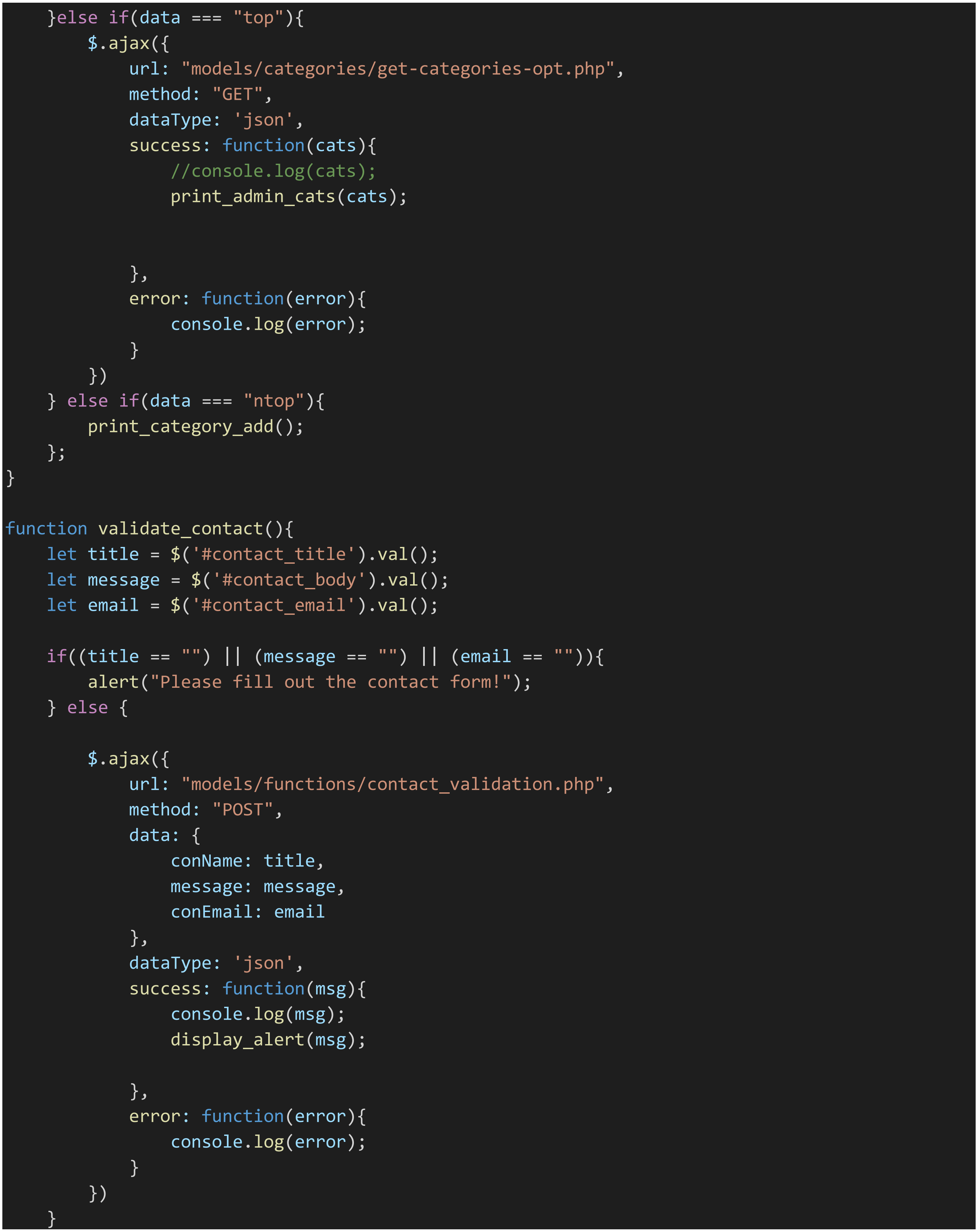
```
$(document).ready(function(){

    $("body").on("click", ".post-pag", function(){

        event.preventDefault();
        let limit = $(this).data("limit");

        $.ajax({
            url: "models/posts/post-pagination.php",
            method: "GET",
            data: {
                limit: limit
            },
            success: function(data){
                //console.log(data);
```



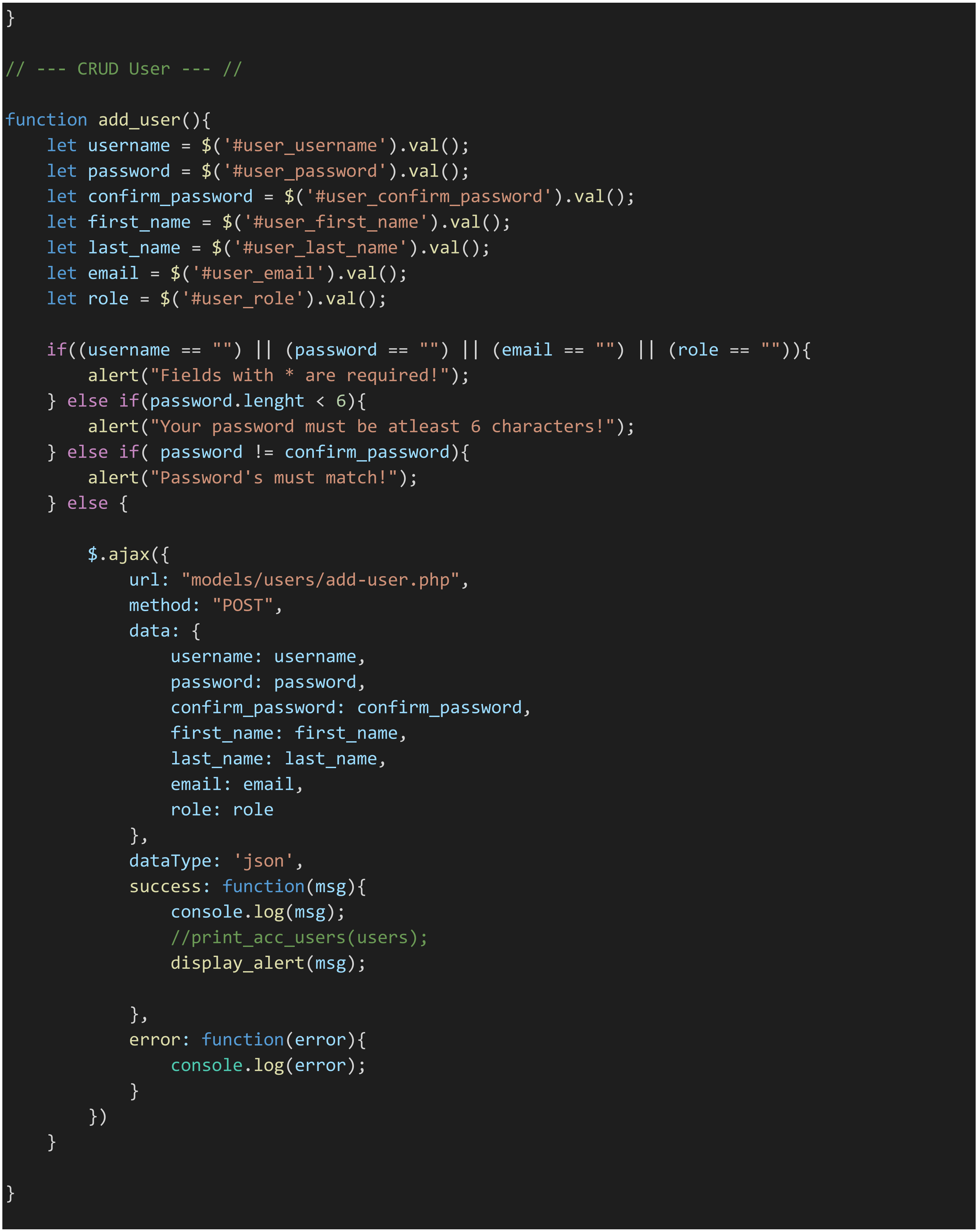
```
                print_posts(data.posts);
                print_pagination(data.num_of_pages);
            },
            error: function(error){
                console.log(error);
            }
        })

    });

    $('.order_posts').click(order_posts);
    $('.sort_posts').click(order_posts);



    $('.acc-opt').on('click',display_acc_content);

});

// --- admin panel --- //

function display_acc_content(){
    event.preventDefault();
    let data = $(this).data('use');
    //console.log(data);
    if(data === "users"){
        let user_id = $(this).data('user_id');
        //console.log(user_id);
        $.ajax({
            url: "models/users/get-users.php",
            method: "GET",
            dataType: 'json',
            success: function(users){
                //console.log(posts);
                print_acc_users(users);


            },
            error: function(error){
                console.log(error);
            }
        })
    } else if(data === "add"){
        $.ajax({
            url: "models/users/get-roles.php",
            method: "GET",
            dataType: 'json',
```



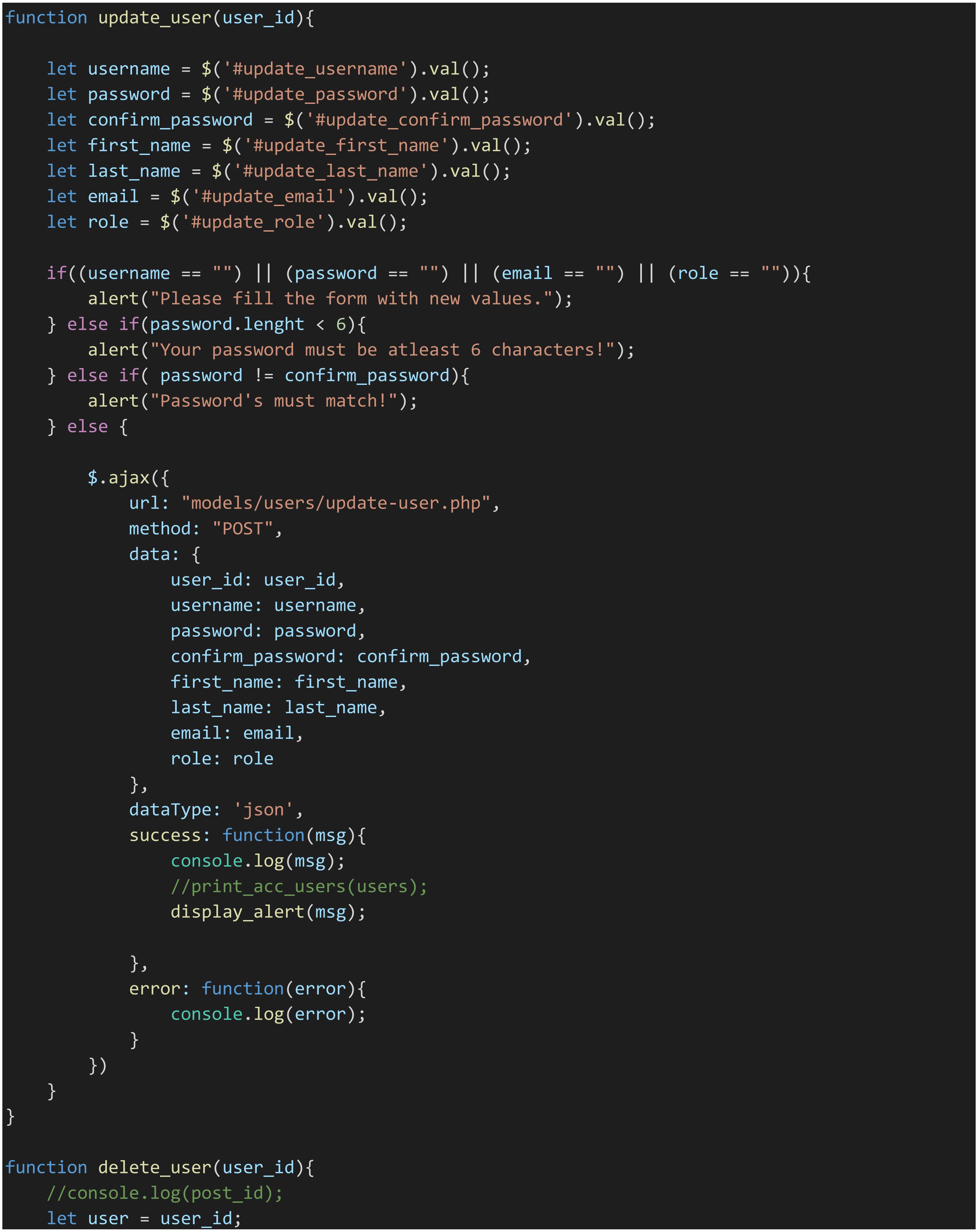
```
            success: function(roles){
                //console.log(roles);
                print_acc_add(roles);


            },
            error: function(error){
                console.log(error);
            }
        })
    } else if(data === "mpost"){
        let user_id = $(this).data('user_id');
        //console.log(user_id);
        $.ajax({
            url: "models/posts/get-user-posts.php",
            method: "POST",
            data: {
                user_id: user_id
            },
            dataType: 'json',
            success: function(posts){
                //console.log(posts);
                print_acc_mpost(posts);


            },
            error: function(error){
                console.log(error);
            }
        })
    } else if(data === "npost"){
        let user_id = $(this).data('user_id');
        //console.log(user_id);
        $.ajax({
            url: "models/categories/get-categories-opt.php",
            method: "GET",
            dataType: 'json',
            success: function(cats){
                //console.log(cats);
                print_acc_npost(cats,user_id);


            },
            error: function(error){
                console.log(error);
            }
        })
```



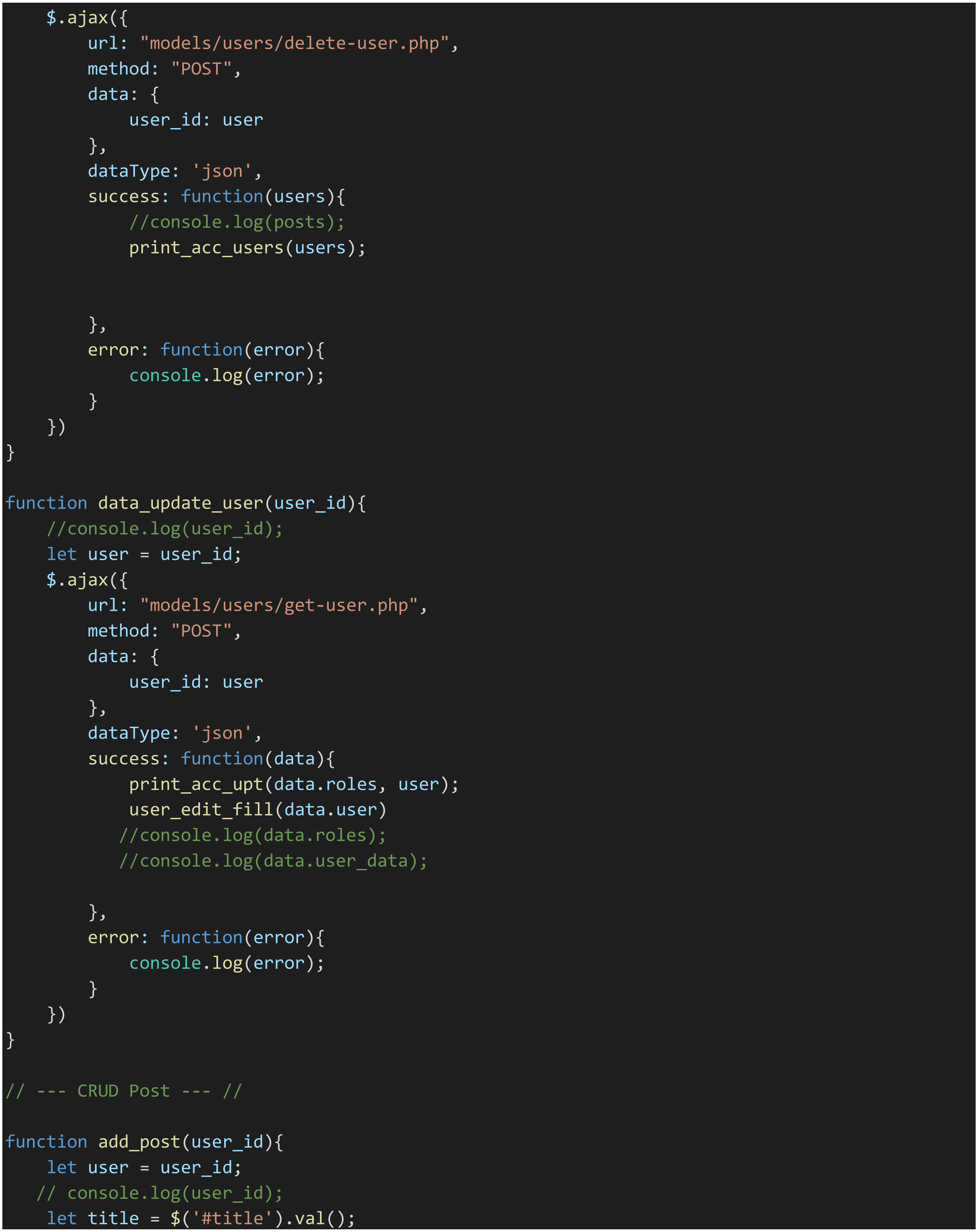
```
    }else if(data === "top"){
        $.ajax({
            url: "models/categories/get-categories-opt.php",
            method: "GET",
            dataType: 'json',
            success: function(cats){
                //console.log(cats);
                print_admin_cats(cats);


            },
            error: function(error){
                console.log(error);
            }
        })
    } else if(data === "ntop"){
        print_category_add();
    };
}

function validate_contact(){
    let title = $('#contact_title').val();
    let message = $('#contact_body').val();
    let email = $('#contact_email').val();

    if((title == "") || (message == "") || (email == "")){
        alert("Please fill out the contact form!");
    } else {

        $.ajax({
            url: "models/functions/contact_validation.php",
            method: "POST",
            data: {
                conName: title,
                message: message,
                conEmail: email
            },
            dataType: 'json',
            success: function(msg){
                console.log(msg);
                display_alert(msg);

            },
            error: function(error){
                console.log(error);
            }
        })
    }
```



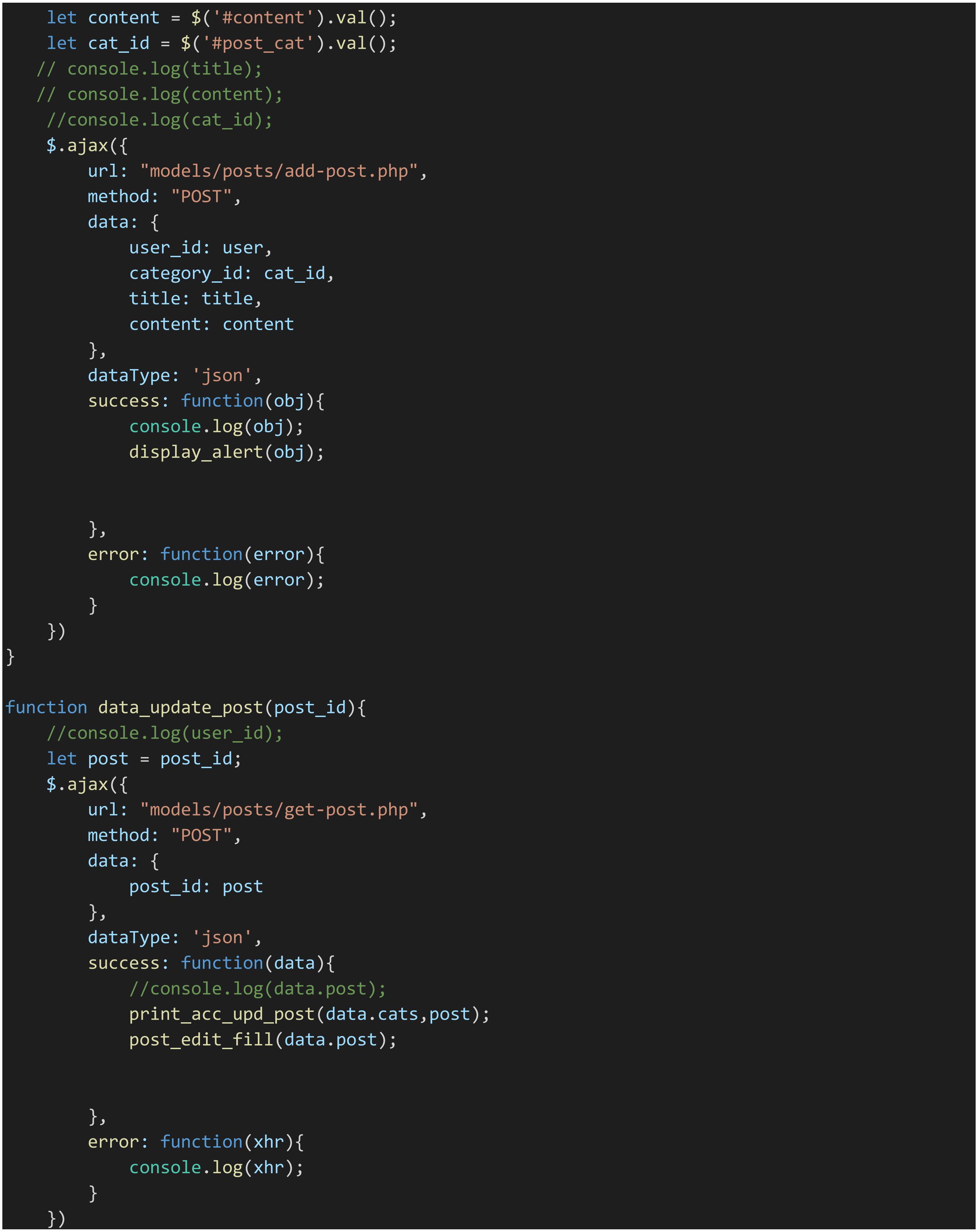
```
}

// --- CRUD User --- //

function add_user(){
    let username = $('#user_username').val();
    let password = $('#user_password').val();
    let confirm_password = $('#user_confirm_password').val();
    let first_name = $('#user_first_name').val();
    let last_name = $('#user_last_name').val();
    let email = $('#user_email').val();
    let role = $('#user_role').val();

    if((username == "") || (password == "") || (email == "") || (role == "")){
        alert("Fields with * are required!");
    } else if(password.lenght < 6){
        alert("Your password must be atleast 6 characters!");
    } else if( password != confirm_password){
        alert("Password's must match!");
    } else {

        $.ajax({
            url: "models/users/add-user.php",
            method: "POST",
            data: {
                username: username,
                password: password,
                confirm_password: confirm_password,
                first_name: first_name,
                last_name: last_name,
                email: email,
                role: role
            },
            dataType: 'json',
            success: function(msg){
                console.log(msg);
                //print_acc_users(users);
                display_alert(msg);

            },
            error: function(error){
                console.log(error);
            }
        })
    }

}
```



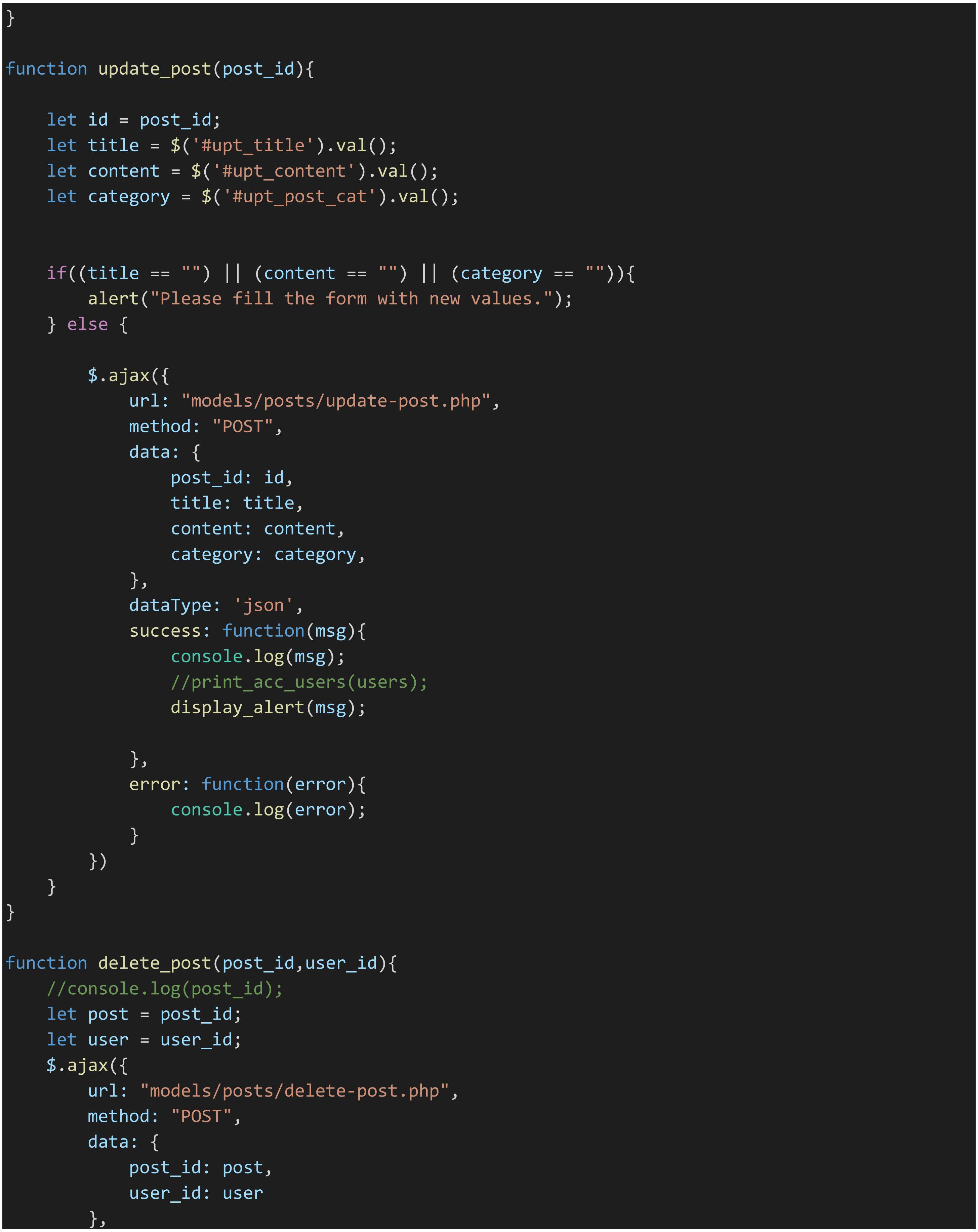
```
function update_user(user_id){

    let username = $('#update_username').val();
    let password = $('#update_password').val();
    let confirm_password = $('#update_confirm_password').val();
    let first_name = $('#update_first_name').val();
    let last_name = $('#update_last_name').val();
    let email = $('#update_email').val();
    let role = $('#update_role').val();

    if((username == "") || (password == "") || (email == "") || (role == "")){
        alert("Please fill the form with new values.");
    } else if(password.lenght < 6){
        alert("Your password must be atleast 6 characters!");
    } else if( password != confirm_password){
        alert("Password's must match!");
    } else {

        $.ajax({
            url: "models/users/update-user.php",
            method: "POST",
            data: {
                user_id: user_id,
                username: username,
                password: password,
                confirm_password: confirm_password,
                first_name: first_name,
                last_name: last_name,
                email: email,
                role: role
            },
            dataType: 'json',
            success: function(msg){
                console.log(msg);
                //print_acc_users(users);
                display_alert(msg);

            },
            error: function(error){
                console.log(error);
            }
        })
    }
}

function delete_user(user_id){
    //console.log(post_id);
    let user = user_id;
```



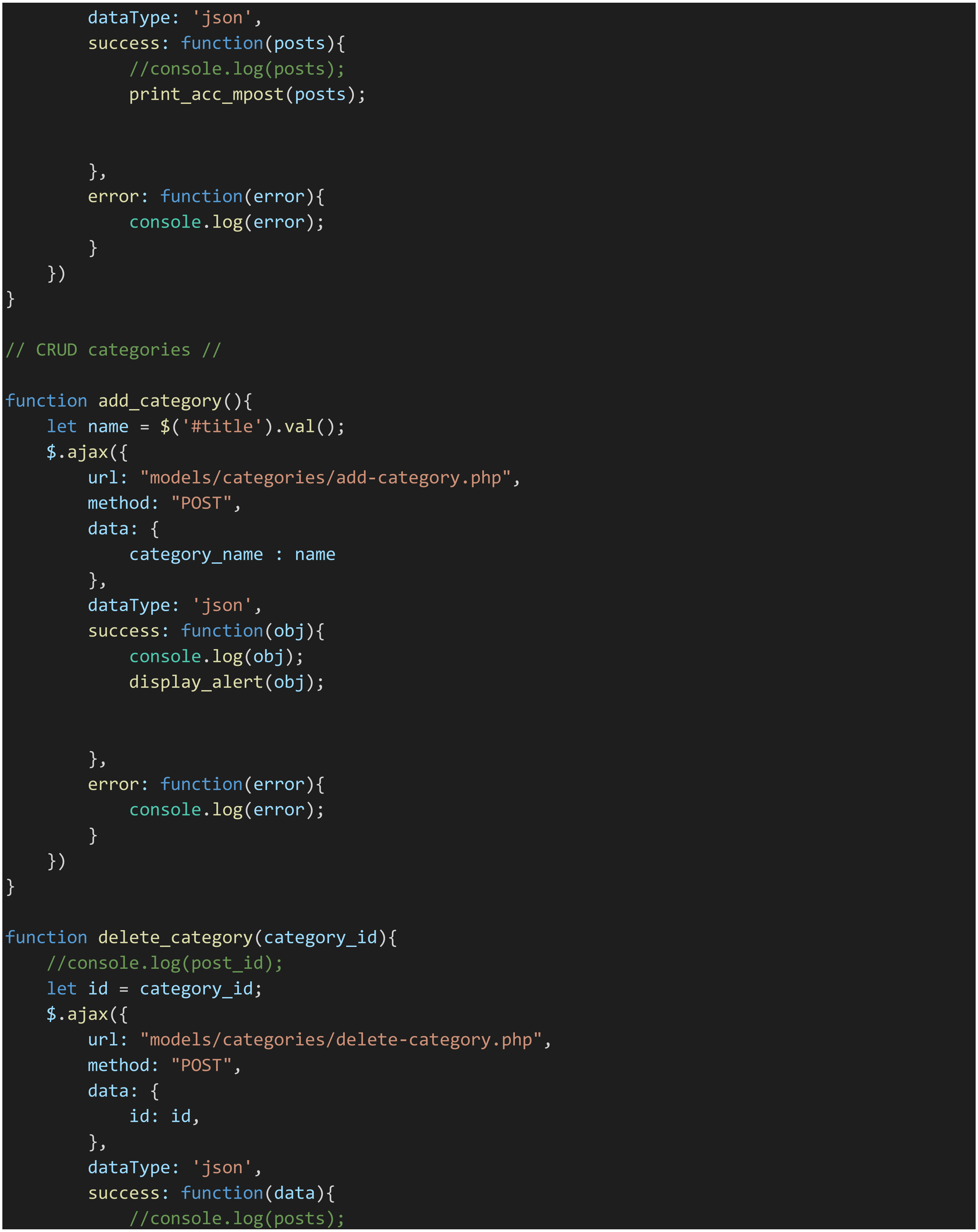
```
    $.ajax({
        url: "models/users/delete-user.php",
        method: "POST",
        data: {
            user_id: user
        },
        dataType: 'json',
        success: function(users){
            //console.log(posts);
            print_acc_users(users);


        },
        error: function(error){
            console.log(error);
        }
    })
}

function data_update_user(user_id){
    //console.log(user_id);
    let user = user_id;
    $.ajax({
        url: "models/users/get-user.php",
        method: "POST",
        data: {
            user_id: user
        },
        dataType: 'json',
        success: function(data){
            print_acc_upt(data.roles, user);
            user_edit_fill(data.user)
           //console.log(data.roles);
           //console.log(data.user_data);

        },
        error: function(error){
            console.log(error);
        }
    })
}

// --- CRUD Post --- //

function add_post(user_id){
    let user = user_id;
   // console.log(user_id);
    let title = $('#title').val();
```



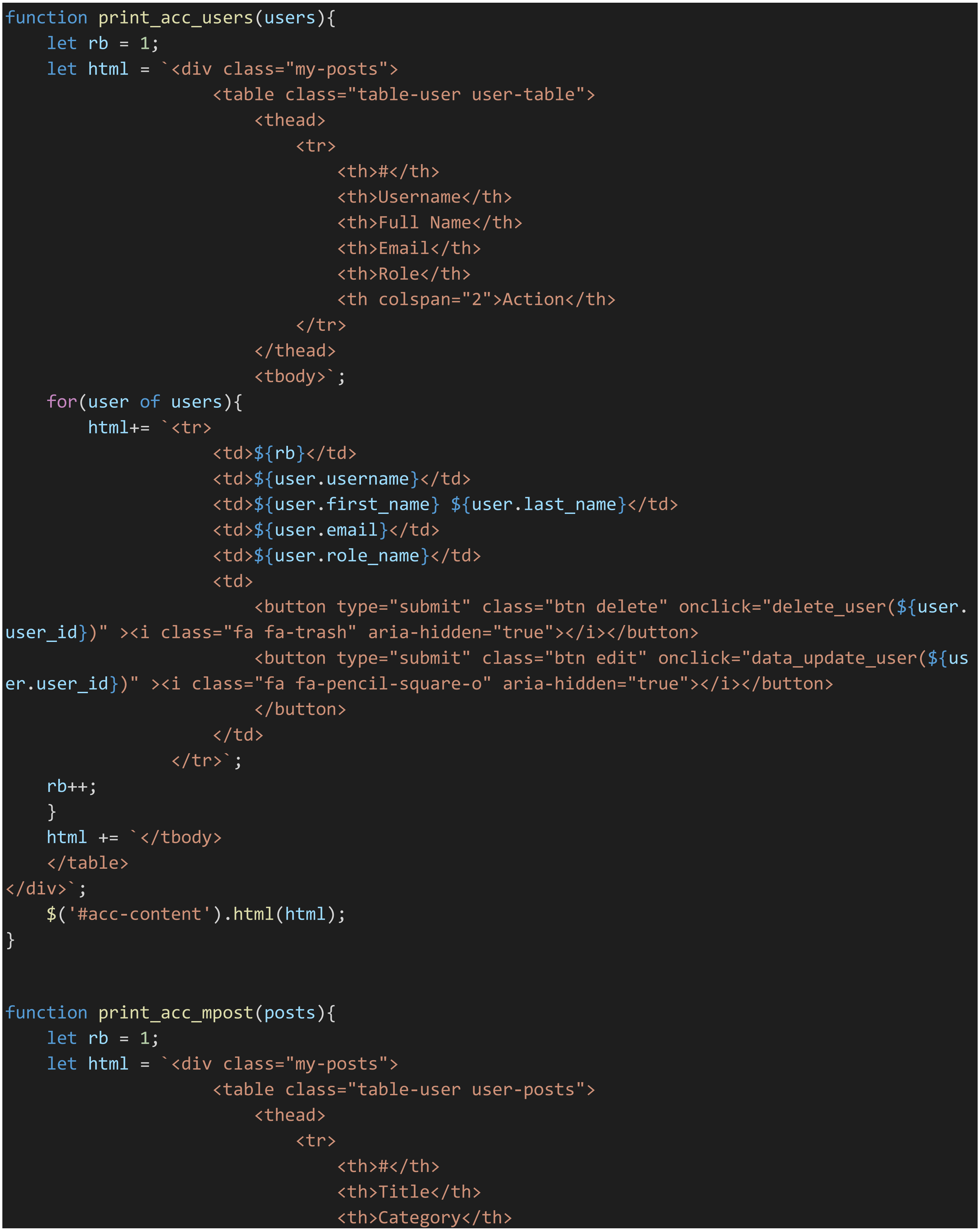
```
    let content = $('#content').val();
    let cat_id = $('#post_cat').val();
   // console.log(title);
   // console.log(content);
    //console.log(cat_id);
    $.ajax({
        url: "models/posts/add-post.php",
        method: "POST",
        data: {
            user_id: user,
            category_id: cat_id,
            title: title,
            content: content
        },
        dataType: 'json',
        success: function(obj){
            console.log(obj);
            display_alert(obj);


        },
        error: function(error){
            console.log(error);
        }
    })
}

function data_update_post(post_id){
    //console.log(user_id);
    let post = post_id;
    $.ajax({
        url: "models/posts/get-post.php",
        method: "POST",
        data: {
            post_id: post
        },
        dataType: 'json',
        success: function(data){
            //console.log(data.post);
            print_acc_upd_post(data.cats,post);
            post_edit_fill(data.post);


        },
        error: function(xhr){
            console.log(xhr);
        }
    })
```



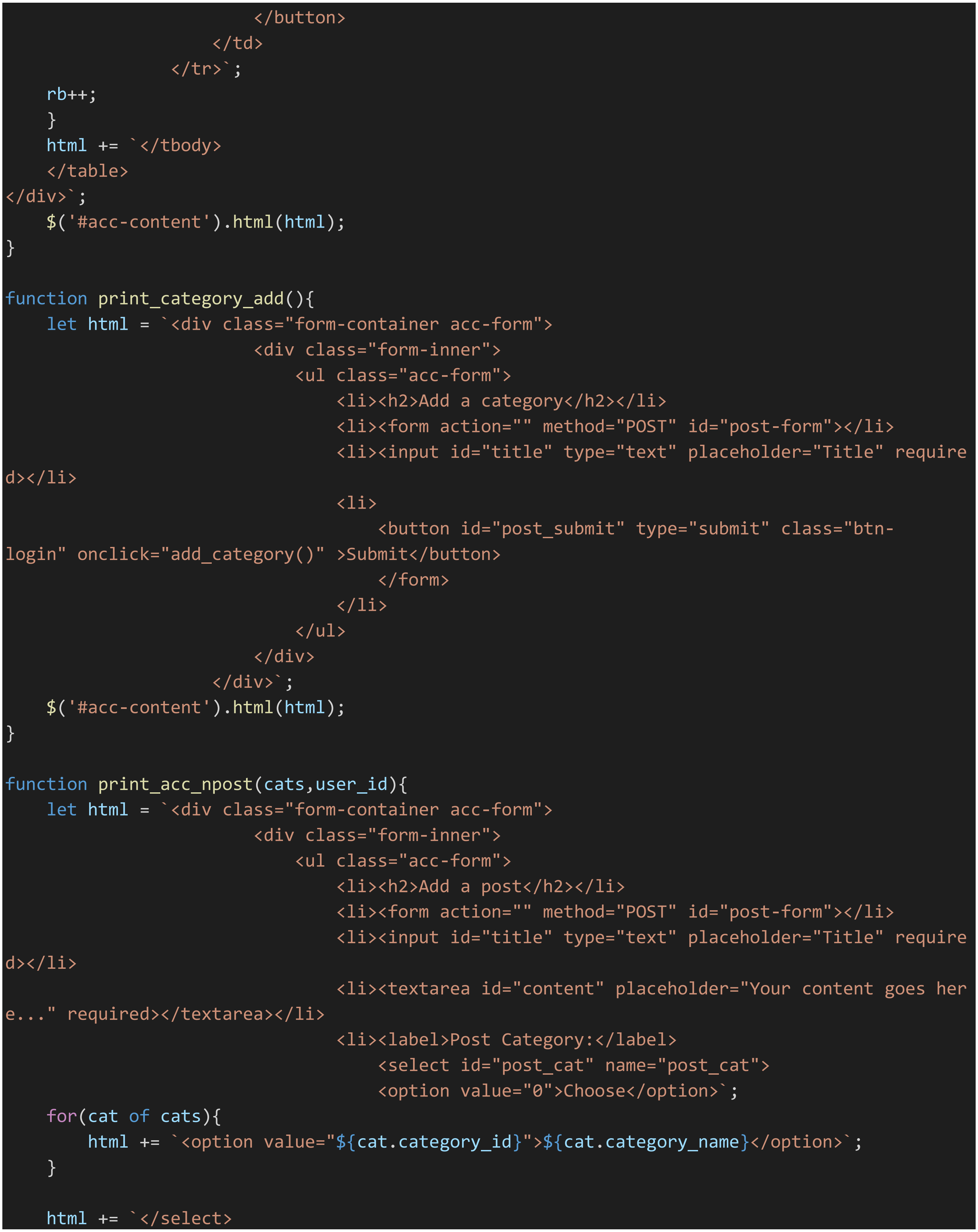
```
}

function update_post(post_id){

    let id = post_id;
    let title = $('#upt_title').val();
    let content = $('#upt_content').val();
    let category = $('#upt_post_cat').val();


    if((title == "") || (content == "") || (category == "")){
        alert("Please fill the form with new values.");
    } else {

        $.ajax({
            url: "models/posts/update-post.php",
            method: "POST",
            data: {
                post_id: id,
                title: title,
                content: content,
                category: category,
            },
            dataType: 'json',
            success: function(msg){
                console.log(msg);
                //print_acc_users(users);
                display_alert(msg);

            },
            error: function(error){
                console.log(error);
            }
        })
    }
}

function delete_post(post_id,user_id){
    //console.log(post_id);
    let post = post_id;
    let user = user_id;
    $.ajax({
        url: "models/posts/delete-post.php",
        method: "POST",
        data: {
            post_id: post,
            user_id: user
        },
```



```
        dataType: 'json',
        success: function(posts){
            //console.log(posts);
            print_acc_mpost(posts);


        },
        error: function(error){
            console.log(error);
        }
    })
}

// CRUD categories //

function add_category(){
    let name = $('#title').val();
    $.ajax({
        url: "models/categories/add-category.php",
        method: "POST",
        data: {
            category_name : name
        },
        dataType: 'json',
        success: function(obj){
            console.log(obj);
            display_alert(obj);


        },
        error: function(error){
            console.log(error);
        }
    })
}

function delete_category(category_id){
    //console.log(post_id);
    let id = category_id;
    $.ajax({
        url: "models/categories/delete-category.php",
        method: "POST",
        data: {
            id: id,
        },
        dataType: 'json',
        success: function(data){
            //console.log(posts);
            print_admin_cats(data);


        },
        error: function(error){
            console.log(error);
        }
    })
}

// ** ORDER and SORT ** //

function order_posts(){
    event.preventDefault()
    let sort = $(this).data('order');
    //console.log(sort);

    $('#pag-list').addClass('hidden');

    $.ajax({
        url: "models/posts/sort-posts.php",
        method: "POST",
        data: {
            sort: sort
        },
        dataType: 'json',
        success: function(data){
            //console.log(data);
            //console.log(data.posts);
            print_posts(data);

        },
        error: function(error){
            console.log(error);
        }
    })
}

$("body").on("click", ".reset", function(){

    event.preventDefault();

    $('#pag-list').removeClass('hidden');

    $.ajax({
        url: "models/posts/post-pagination.php",
        method: "GET",
        data: {
            limit: 0
        },
        success: function(data){
            //console.log(data);

            print_posts(data.posts);
            print_pagination(data.num_of_pages);
        },
        error: function(error){
            console.log(error);
        }
    })

});

function display_alert(obj){
    alert(obj.msg);
}

// --- print content --- //

function print_posts(posts){
    let html = "";
    for(let post of posts){
        let date = new Date(post.post_date);
        date = date.getDate() + "-" + (date.getMonth() + 1) + "-" + date.getFullYear();
        html += `
        <div class="post-row">
                <div class="blog-post">

                    <div class="blog-post-inner">
                        <a href="#" class="post-about">
                            <div class="post-member">
                                <img src="assets/img/user_images/${post.avatar_img}" alt="avatar-image">

                                <span class="post-user">${post.username}</span>
                                <span class="acc-role">${post.role_name}</span>
                            </div>
                        </a>
                        <div class="post-content">
                            <h2 class="blog-post-title">${post.title}</h2>
                            <p>${post.content}</p>
                            <p><a href="index.php?page=post&post_id=${post.post_id}" class="">Continue reading...</a></p>
                        </div>
                    </div>
                    <div class="post-date">
                        <i class="fa fa-clock-o" aria-hidden="true"></i>
                        ${date}

                    </div>
                </div>
            </div> `;
    }
    $('#main-post-area').html(html);
}

function print_acc_add(roles){
    let html = `<div class="form-container acc-form">
    <div class="form-inner">
        <ul class="edit-form">
            <li><h2>Add user</h2></li>
            <li><form action="" method="POST" id="user-form"></li>
            <li><label>Username * :</label></li>
            <li><input id="user_username" name="user_username" type="text" required></li>
            <li><label>Password * :</label></li>
            <li><input id="user_password" name="user_password" type="password" required></li>

            <li><label>Confirm password * :</label></li>
            <li><input id="user_confirm_password" name="user_confirm_password" type="password
" required></li>
            <li><label>First name:</label></li>
            <li><input id="user_first_name" name="user_first_name" type="text" ></li>
            <li><label>Last name:</label></li>
            <li><input id="user_last_name" name="user_last_name" type="text" ></li>
            <li><label>Email * :</label></li>
            <li><input id="user_email" name="email" type="email" required></li>
            <li><label>Role * :</label></li>
            <li><select id="user_role" name="user_role">
                <option value="0">Choose</option>`;
    for(role of roles){
        html += `<option value="${role.role_id}">${role.role_name}</option>`;
    }
    html += `</select></li>
            <li>
                <button id="user_submit" type="submit" class="btn-
reg" onclick=add_user()>Add user</button>
                </form>
            </li>
        </ul>
    </div>
</div>`;
$('#acc-content').html(html);
}

function print_acc_upt(roles,user){
    let html = `<div class="form-container acc-form">
    <div class="form-inner">
        <ul class="edit-form">
            <li><h2>Edit user info</h2></li>
            <li><form action="" method="POST" id="user-form"></li>
            <li><label>Username:</label></li>
            <li><input id="update_username" name="user_username" type="text" required></li>
            <li><label>Password:</label></li>
            <li><input id="update_password" name="user_password" type="password" required></li>
            <li><label>Confirm password:</label></li>
            <li><input id="update_confirm_password" name="user_confirm_password" type="password" required></li>
            <li><label>First name:</label></li>
            <li><input id="update_first_name" name="user_first_name" type="text" ></li>
            <li><label>Last name:</label></li>
            <li><input id="update_last_name" name="user_last_name" type="text" ></li>
            <li><label>Email:</label></li>
            <li><input id="update_email" name="email" type="email" required></li>
            <li><label>Role:</label></li>
            <li><select id="update_role" name="user_role">
                <option value="0">Choose</option>`;
    for(role of roles){
        html += `<option value="${role.role_id}">${role.role_name}</option>`;
    }
    html += `</select></li>
            <li>
                <button id="user_submit" type="submit" class="btn-login" onclick=update_user(${user})>Change</button>
                </form>
            </li>
        </ul>
    </div>
</div>`;
$('#acc-content').html(html);
}

function user_edit_fill(user_data){
    //console.log(user_data);
    for(data of user_data){
        $('#update_username').val(data.username);
        $('#update_first_name').val(data.first_name);
        $('#update_last_name').val(data.last_name);
        $('#update_email').val(data.email);
        $('#update_role').val(data.role_id);
    }
}
```



```
function print_acc_users(users){
    let rb = 1;
    let html = `<div class="my-posts">
                    <table class="table-user user-table">
                        <thead>
                            <tr>
                                <th>#</th>
                                <th>Username</th>
                                <th>Full Name</th>
                                <th>Email</th>
                                <th>Role</th>
                                <th colspan="2">Action</th>
                            </tr>
                        </thead>
                        <tbody>`;
    for(user of users){
        html+= `<tr>
                    <td>${rb}</td>
                    <td>${user.username}</td>
                    <td>${user.first_name} ${user.last_name}</td>
                    <td>${user.email}</td>
                    <td>${user.role_name}</td>
                    <td>
                        <button type="submit" class="btn delete" onclick="delete_user(${user.
user_id})" ><i class="fa fa-trash" aria-hidden="true"></i></button>
                        <button type="submit" class="btn edit" onclick="data_update_user(${us
er.user_id})" ><i class="fa fa-pencil-square-o" aria-hidden="true"></i></button>
                        </button>
                    </td>
                </tr>`;
    rb++;
    }
    html += `</tbody>
    </table>
</div>`;
    $('#acc-content').html(html);
}


function print_acc_mpost(posts){
    let rb = 1;
    let html = `<div class="my-posts">
                    <table class="table-user user-posts">
                        <thead>
                            <tr>
                                <th>#</th>
                                <th>Title</th>
                                <th>Category</th>
                                <th>Date</th>
                                <th colspan="2">Action</th>
                            </tr>
                        </thead>
                        <tbody>`;
    for(post of posts){
        html+= `<tr>
                    <td>${rb}</td>
                    <td>${post.title}</td>
                    <td>${post.cat_name}</td>
                    <td>
                        <i class="fa fa-clock-o" aria-hidden="true">
                        ${post.post_date}
                    </td>
                    <td>
                        <button type="submit" class="btn delete" onclick="delete_post(${post.
post_id},${post.user_id})" ><i class="fa fa-trash" aria-hidden="true"></i></button>
                        <button type="submit" class="btn edit" onclick="data_update_post(${po
st.post_id})" ><i class="fa fa-pencil-square-o" aria-hidden="true"></i></button>
                    </td>
                </tr>`;
        rb++;
    }
    html += `</tbody>
    </table>
</div>`;
    $('#acc-content').html(html);
}

function print_admin_cats(cats){
    let rb = 1;
    let html = `<div class="my-posts">
                    <table class="table-user user-table">
                        <thead>
                            <tr>
                                <th>#</th>
                                <th>Name</th>
                                <th>Action</th>
                            </tr>
                        </thead>
                        <tbody>`;
    for(cat of cats){
        html+= `<tr>
                    <td>${rb}</td>
                    <td>${cat.category_name}</td>
                    <td>
                        <button type="submit" class="btn delete" onclick="delete_category(${c
at.category_id})" ><i class="fa fa-trash" aria-hidden="true"></i></button>
```



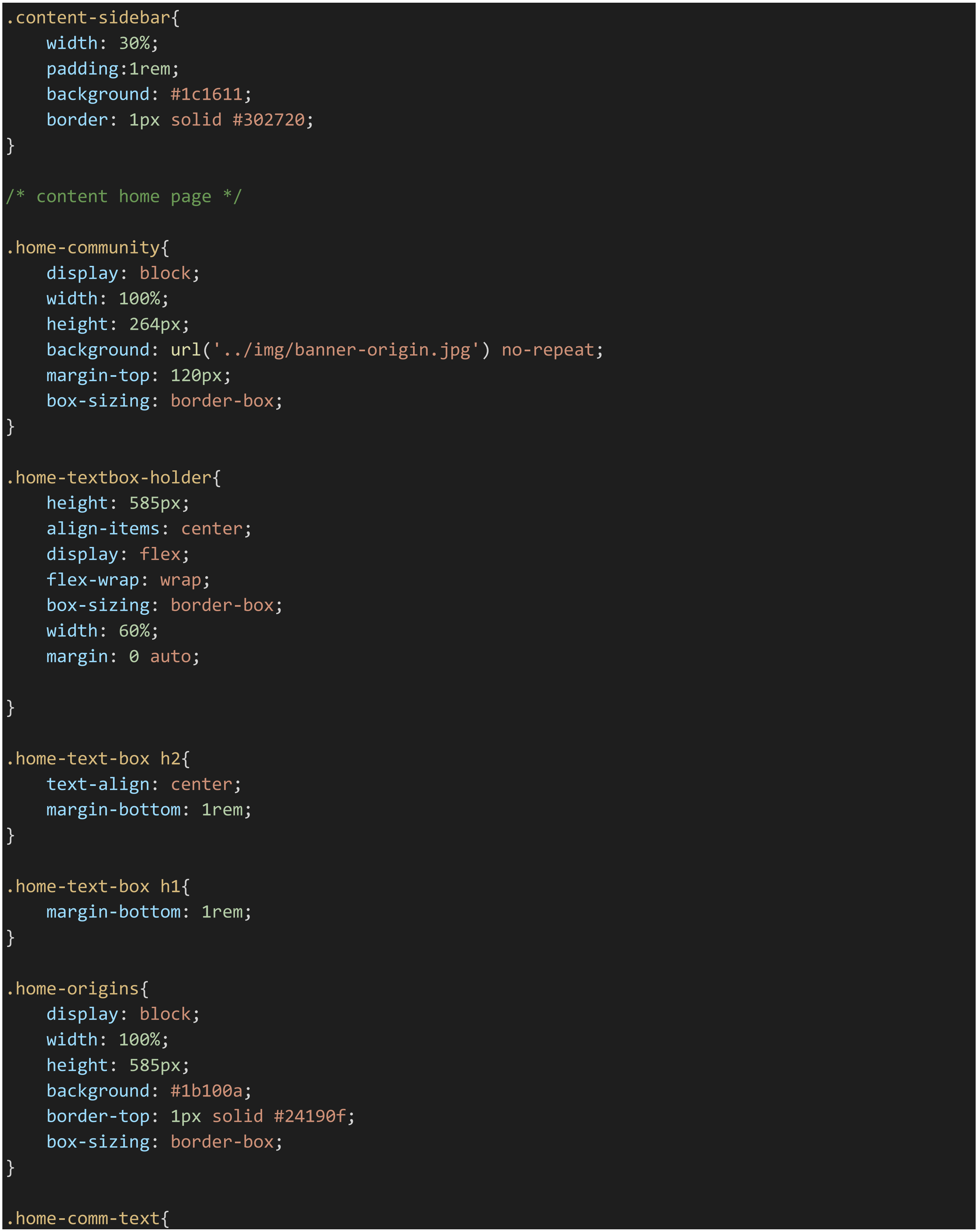
```
                        </button>
                    </td>
                </tr>`;
    rb++;
    }
    html += `</tbody>
    </table>
</div>`;
    $('#acc-content').html(html);
}

function print_category_add(){
    let html = `<div class="form-container acc-form">
                        <div class="form-inner">
                            <ul class="acc-form">
                                <li><h2>Add a category</h2></li>
                                <li><form action="" method="POST" id="post-form"></li>
                                <li><input id="title" type="text" placeholder="Title" require
d></li>
                                <li>
                                    <button id="post_submit" type="submit" class="btn-
login" onclick="add_category()" >Submit</button>
                                    </form>
                                </li>
                            </ul>
                        </div>
                    </div>`;
    $('#acc-content').html(html);
}

function print_acc_npost(cats,user_id){
    let html = `<div class="form-container acc-form">
                        <div class="form-inner">
                            <ul class="acc-form">
                                <li><h2>Add a post</h2></li>
                                <li><form action="" method="POST" id="post-form"></li>
                                <li><input id="title" type="text" placeholder="Title" require
d></li>
                                <li><textarea id="content" placeholder="Your content goes her
e..." required></textarea></li>
                                <li><label>Post Category:</label>
                                    <select id="post_cat" name="post_cat">
                                    <option value="0">Choose</option>`;
    for(cat of cats){
        html += `<option value="${cat.category_id}">${cat.category_name}</option>`;
    }

    html += `</select>
```

```
            </li>
            <li>
                <button id="post_submit" type="submit" class="btn-login" onclick="add_post(${user_id})" >Submit post</button>
                </form>
            </li>
        </ul>
    </div>
</div>`;
    $('#acc-content').html(html);
}

function print_acc_upd_post(cats,post){

    let html = `<div class="form-container acc-form">
                        <div class="form-inner">
                            <ul class="acc-form">
                                <li><h2>Edit your post</h2></li>
                                <li><form action="" method="POST" id="post-form"></li>
                                <li><input id="upt_title" type="text" placeholder="Title" required></li>
                                <li><textarea id="upt_content" placeholder="Your content goes here..." required></textarea></li>
                                <li><label>Post Category:</label>
                                    <select id="upt_post_cat" name="post_cat">
                                    <option value="0">Choose</option>`;
    for(cat of cats){
        html += `<option value="${cat.category_id}">${cat.category_name}</option>`;
    }

    html += `</select>
            </li>
            <li>
                <button id="post_submit" type="submit" class="btn-login" onclick="update_post(${post})" >Submit post</button>
                </form>
            </li>
        </ul>
    </div>
</div>`;
    $('#acc-content').html(html);
}

function post_edit_fill(post_data){
    //console.log(user_data);
    for(data of post_data){
        $('#upt_title').val(data.title);
        $('#upt_content').val(data.content);
        $('#upt_post_cat').val(data.category_id);
    }
}

function print_pagination(num_of_pages){
    let html = "";
    for(let i = 0; i < num_of_pages; i++){
        html += `<li>
                <a href="#" class="post-pag" data-limit="${ i }">
                    ${ i + 1 }
                </a>
            </li>`;
    }
    $("#pag-list").html(html);
}
```

CSS

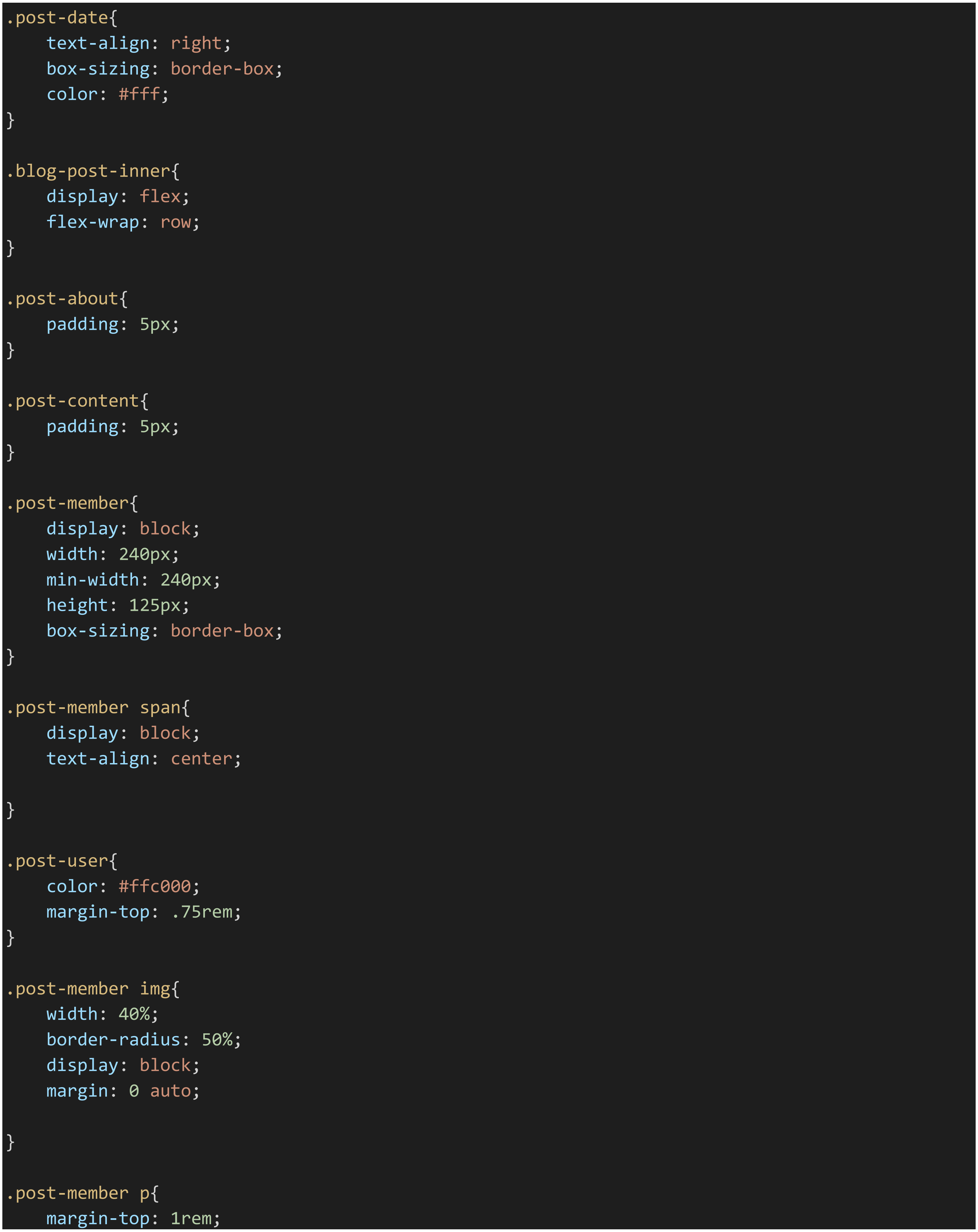
style.css

```
@font-face{
    font-family: Futura;
    src: url('fonts/FuturaBook.otf');
}
@font-face{
    font-family: Antika;
    src: url('fonts/ManuskriptAntD-Reg.otf');
}
*{
    margin: 0;
    padding: 0;
}
a{
    text-decoration: none;
    color:#fff;
}
ul,ol {
    padding:0;
    margin:0 0 20px 0;
    list-style:none;
}

body{
    font-family: 'Futura';
    font-size: 0.9rem;
    font-weight: 400;
    line-height: 1.5;
    color: #aa9c87;
    text-align: left;
    background-color: #180d08;
}

h1,h2,h3,h4{
    color: #fff;
}

#site{
    padding-top: 66px;
}

.home{
    background: url('../img/banner-home-opc.jpg') no-repeat;
    background-position: center 20px;
    background-color: #180d08;
}


.forum{
    background: url('../img/banner-forum.jpg') no-repeat;
    background-position: center 20px;
    background-color: #180d08;
}

.login, .register, .contact{
    background: url('../img/banner-log.jpg') no-repeat;
    background-position: center 20px;
    background-color: #180d08;
}

.admin, .user{
    background: url('../img/banner-admin.jpg') no-repeat;
    background-position: center 20px;
    background-color: #180d08;
}


ul li {
    padding:5px 0;
}


/* navigation */

header{
    width:100%;

}

.nav-wrapper{
    width: 100%;
    padding-right: 15px;
    padding-left: 15px;
    margin-right: auto;
    margin-left: auto;
    box-sizing: border-box;
}

.head-logo{
    position: fixed;
    float: left;
    padding: 7px 0;
}

.nav-user a{
    text-transform: none;
}

.my-acc{
    display: none;
}

.acc-nav{
    display:inline-block;
}

.nav-logo{
    width: 50px;
    height: 50px;
    background: url('../img/logo.png') no-repeat;
    position: center;
    display: block;
    box-sizing: border-box;
}

.nav-menu {
    margin-left: 4rem;
}

nav{
    position: fixed;
    top: 0;
    z-index: 10;
    width: 100%;
    background: #170f0a;
    border-bottom: 1px solid #30241b !important;
}

nav ul{
    list-style:none;
    overflow: hidden;
    padding: 0;
    margin:0;
    text-align: left;
    transition: 1s;
}

nav ul li{
    display: inline-block;
    padding: 20px;
    text-transform: uppercase;
}

nav ul li a{
    color: #6c5d48;
    text-decoration: none;
}

nav ul li a:hover{
    color: #ffc000;
}

.menu-icon {
    width: 100%;
    text-align: right;
    box-sizing: border-box;
    padding: 15px 24px;
    cursor: pointer;
    display: none;
    color: #6c5d48;
    padding-right: 0;
}

/* content */

.main-banner-box{
    width: 100%;
    height: 32vh;
    display: flex;
    -webkit-box-pack: center;
    justify-content: center;
    -webkit-box-orient: vertical;
    -webkit-box-direction: normal;
    flex-direction: column;
    -webkit-box-align: center;
    align-items: center;
    text-align: center;
    width: 100%;
    box-sizing: border-box;
}

.page-title{
    font-size: 4rem;
    color: #fff;
    box-sizing: border-box;
    text-align: center;
    font-family: 'Antika';
    text-transform: uppercase;
}

.container{
    width: 100%;
    padding-right: 15px;
    padding-left: 15px;
    margin-right: auto;
    margin-left: auto;
    box-sizing: border-box;
}

#page-content-wrapper{
    width: 100%;
    box-sizing: border-box;

}

.content-wrapper-inner{
    width: 100%;
    box-sizing: border-box;

}

#sort-content{
    width: 100%;
}

#sort-ul{
    width: 100%;
    order: 6;
    margin-bottom: 0;
    margin-top: 15.4px;
    height: auto;
    float: none;
    margin-right: 0;
    position: relative;
    font-size: 15px;
    font-weight: 400;
    border-bottom: solid 1px rgba(255,255,255,0.15);
    text-align: center;
}

#sort-ul li {
    margin-right: 0.75em;
    font-size: 18px;
    display: inline-block;
    text-transform: uppercase;
    border-bottom: 1px solid transparent;
    cursor: pointer;

}

#sort-ul li a{
    display: block;
    padding: 20px;
}

#sort-ul li:hover {
    border-bottom: 1px solid #ffc000;

}

#sort-ul li a:hover {
    color:#ffc000;

}

.main-content{
    display: flex;
    box-sizing: border-box;
    margin: 2rem 0;
}
#main-post-area{
    width: 100%;
}
.home-left-content-area{
    width: 100%;
    padding: 0 1rem;
}
```

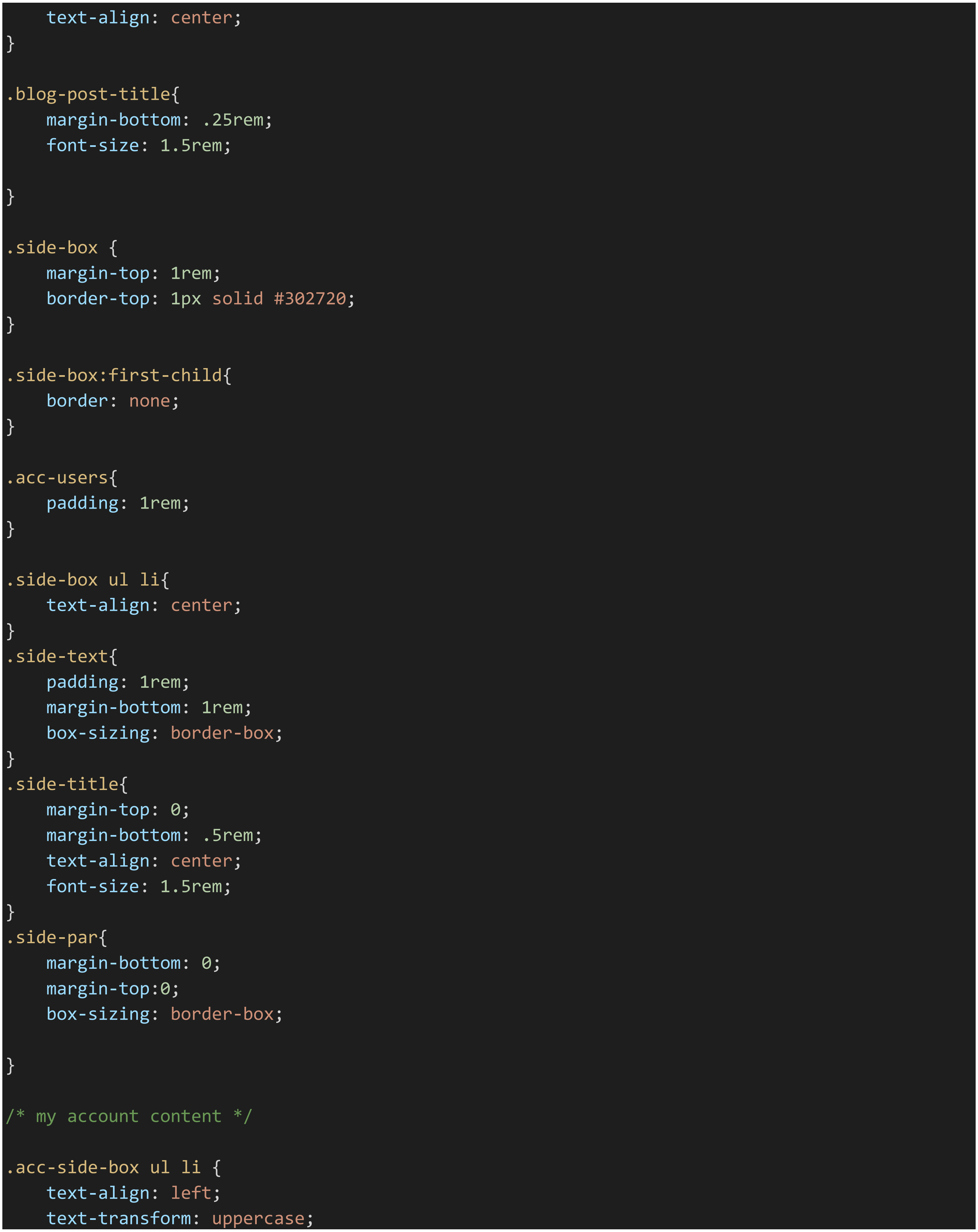


```
.content-sidebar{
    width: 30%;
    padding:1rem;
    background: #1c1611;
    border: 1px solid #302720;
}

/* content home page */

.home-community{
    display: block;
    width: 100%;
    height: 264px;
    background: url('../img/banner-origin.jpg') no-repeat;
    margin-top: 120px;
    box-sizing: border-box;
}

.home-textbox-holder{
    height: 585px;
    align-items: center;
    display: flex;
    flex-wrap: wrap;
    box-sizing: border-box;
    width: 60%;
    margin: 0 auto;

}

.home-text-box h2{
    text-align: center;
    margin-bottom: 1rem;
}

.home-text-box h1{
    margin-bottom: 1rem;
}

.home-origins{
    display: block;
    width: 100%;
    height: 585px;
    background: #1b100a;
    border-top: 1px solid #24190f;
    box-sizing: border-box;
}

.home-comm-text{
```

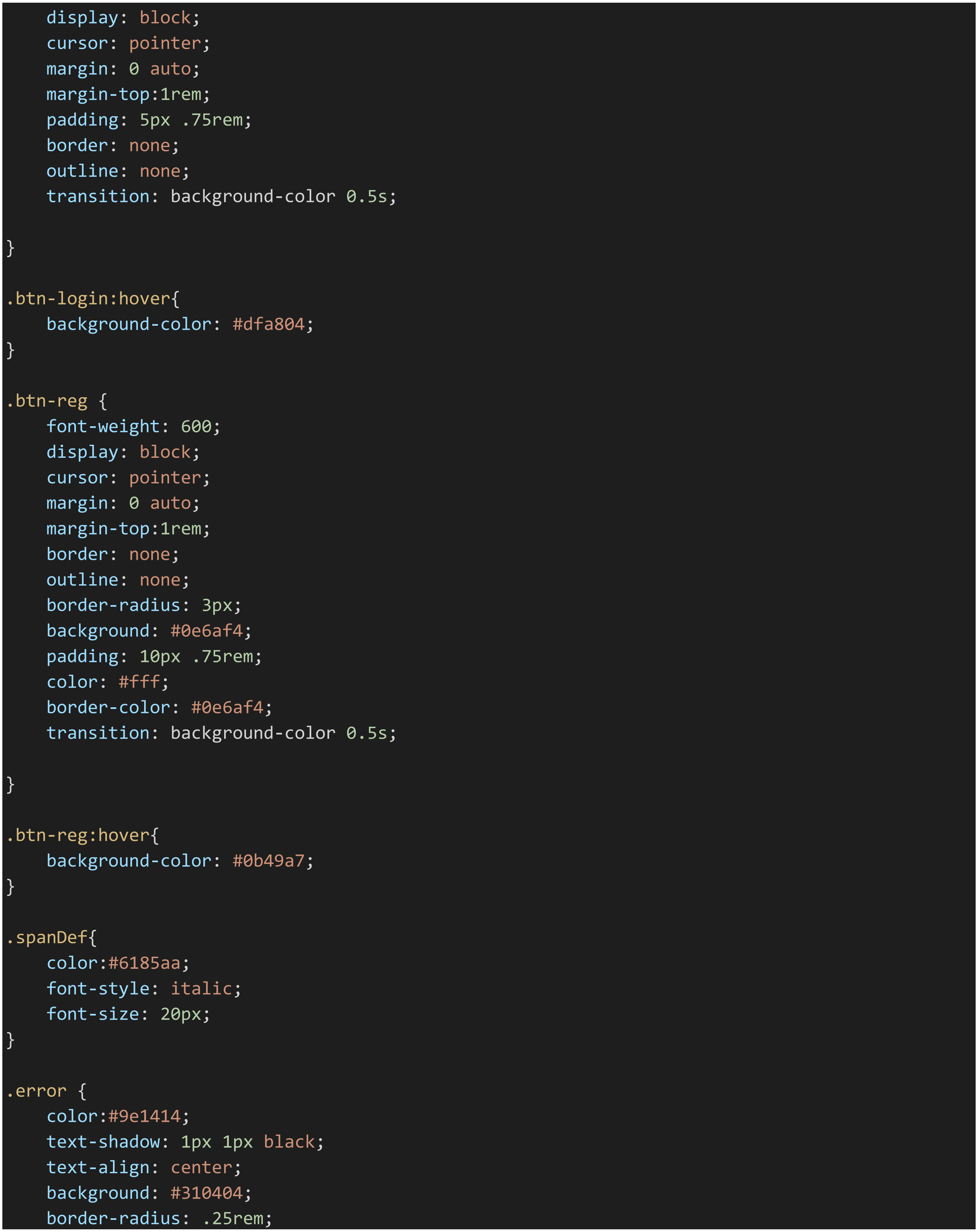
```css
    width: 100%;
    height: 264px;
    padding: 0;
    margin: 0;
    display: flex;
    align-items: center;
    justify-content: right;
}

.commbox {
    width: 40%;
}

.home-about-game{
    display: block;
    width: 100%;
    background: url('../img/banner-aboutg.jpg') no-repeat top;
    height: 608px;
    box-sizing: border-box;
}

.home-about-text{
    height: 608px;
    align-items: center;
    display: flex;
    flex-wrap: wrap;
    box-sizing: border-box;
    width: 60%;

}

/* Posts */

.post-row{
    box-sizing: border-box;
    text-align: left;
    padding: 10px;
    border-top: 1px solid #302720
}

.post-row:first-child{
    border:none;
}

.blog-post{
    box-sizing: border-box;
}
```



```css
.post-date{
    text-align: right;
    box-sizing: border-box;
    color: #fff;
}

.blog-post-inner{
    display: flex;
    flex-wrap: row;
}

.post-about{
    padding: 5px;
}

.post-content{
    padding: 5px;
}

.post-member{
    display: block;
    width: 240px;
    min-width: 240px;
    height: 125px;
    box-sizing: border-box;
}

.post-member span{
    display: block;
    text-align: center;

}

.post-user{
    color: #ffc000;
    margin-top: .75rem;
}

.post-member img{
    width: 40%;
    border-radius: 50%;
    display: block;
    margin: 0 auto;

}

.post-member p{
    margin-top: 1rem;
```

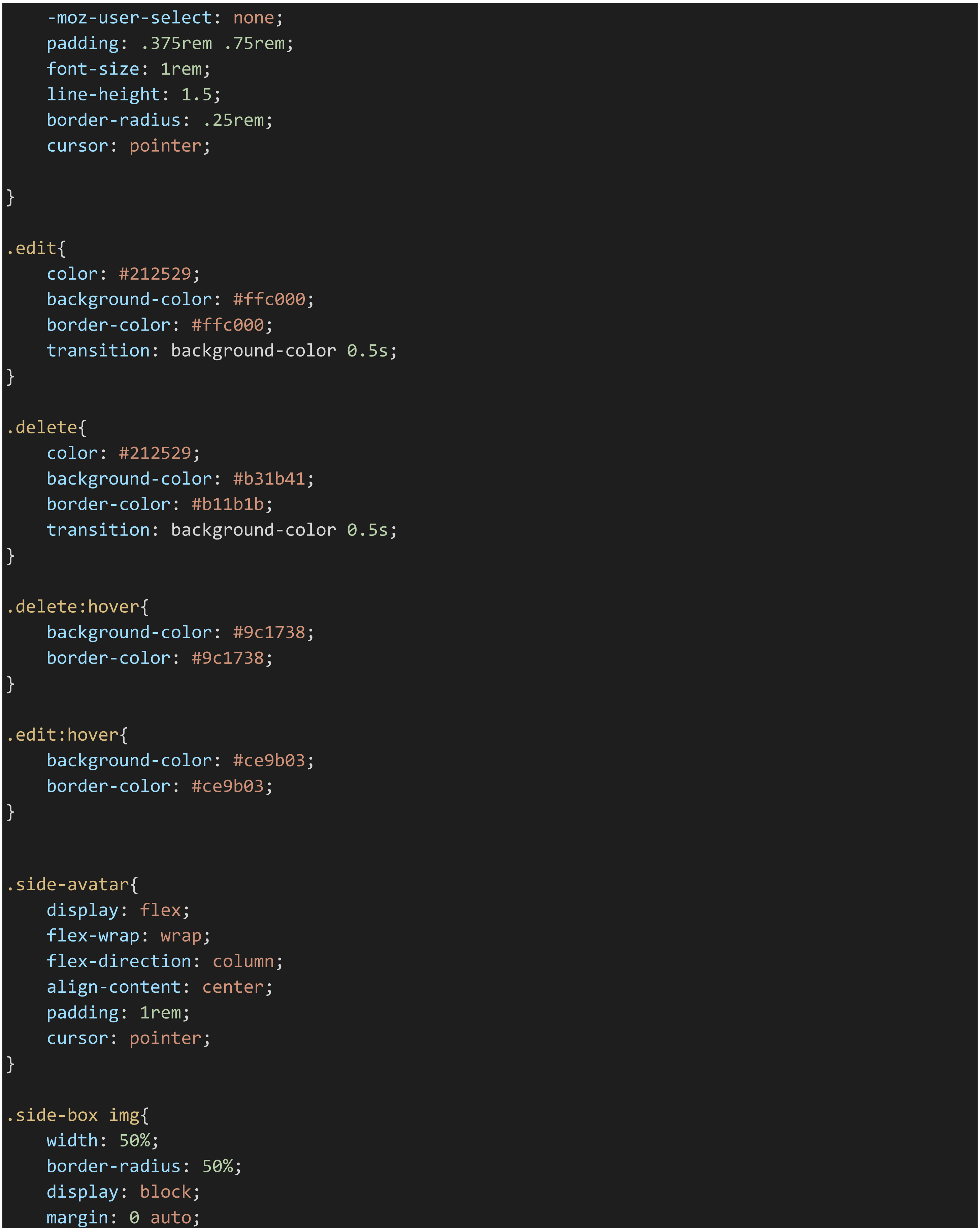


```
    text-align: center;
}

.blog-post-title{
    margin-bottom: .25rem;
    font-size: 1.5rem;

}

.side-box {
    margin-top: 1rem;
    border-top: 1px solid #302720;
}

.side-box:first-child{
    border: none;
}

.acc-users{
    padding: 1rem;
}

.side-box ul li{
    text-align: center;
}
.side-text{
    padding: 1rem;
    margin-bottom: 1rem;
    box-sizing: border-box;
}
.side-title{
    margin-top: 0;
    margin-bottom: .5rem;
    text-align: center;
    font-size: 1.5rem;
}
.side-par{
    margin-bottom: 0;
    margin-top:0;
    box-sizing: border-box;

}

/* my account content */

.acc-side-box ul li {
    text-align: left;
    text-transform: uppercase;
    cursor: pointer;
    padding: 0;
    position: relative;
}

.acc-side-box ul li:hover{
    border-left: 1px solid #ffc000;
    background: #180f0a;
}

.acc-side-box ul li a{
    color: #aa9c87;
    display: block;
    padding: 1rem;
}

.side-box ul li a:hover{
    color: #ffc000;
}

.user-info{
    max-width: 500px;
    margin: 0 auto 10rem auto;
    background-color: #1f140d;
    border: 1px solid #2c1f15;
    border-radius: .25rem;
    font-weight: 400;
    color: #aa9c87;
    text-align: left;
    display: flex;
    flex-direction: column;
    padding: 1rem;
    box-sizing: border-box;
}

.user-info-list li h2{
    text-align: center;
}

.user-info-list li span{
    margin-left: 1rem;
    text-align: right;
}

/* forms */

ul li input[type="text"], input[type="password"], input[type="email"], textarea {
    width:50%;
    padding: 0.5em;
    border: none;
    outline:none;
    border-radius: 3px;
    color: #7f6f65;
    background-color: #271b14;
    border: 1px solid #332721;
    border-collapse: collapse;
    border-spacing: 0;

}

ul li input[type="text"]:focus, input[type="password"]:focus, input[type="email"]:focus ,textarea:focus{
    box-shadow: 0 0 5px #724500;
    transition: 0.5s;

}

ul li textarea{
    resize: none;
    min-height: 150px;
}

ul li select{
    width: 50%;
    padding: 0.5em;
    border: none;
    outline:none;
    border-radius: 3px;
    color: #7f6f65;
    background-color: #271b14;
    border: 1px solid #332721;
    border-spacing: 0;
}


.form-container{
    max-width: 500px;
    margin: 0 auto 10rem auto;
    background-color: #1f140d;
    border: 1px solid #2c1f15;
    border-radius: .25rem;
    font-weight: 400;
    color: #aa9c87;
    text-align: left;
    display: flex;
    flex-direction: column;
    padding: 1rem;
    box-sizing: border-box;
}

.form-inner{
    width:100%;
}


.form-inner ul li{
    text-align: center;
    margin: 0 auto;
}

.form-inner:nth-child(2) ul li{
    text-align: left;

}

.form-inner:nth-child(2) ul li a{
    color: #8d8172;

}

.form-inner:nth-child(2) ul li a:hover{
    text-decoration: underline;
}

.line{
    width: 90%;
    margin: 0 auto;
    border-top: 1px solid #281a10;
}

.reg-inner{
    padding: 1rem;
}

.login-form span{
    text-align: left;
    width: 80%;
}
.btn-login {
    border-radius: 2px;
    font-weight: 600;
    color: #212529;
    background-color: #ffc000;
    border-color: #ffc000;
```



```css
    display: block;
    cursor: pointer;
    margin: 0 auto;
    margin-top:1rem;
    padding: 5px .75rem;
    border: none;
    outline: none;
    transition: background-color 0.5s;

}

.btn-login:hover{
    background-color: #dfa804;
}

.btn-reg {
    font-weight: 600;
    display: block;
    cursor: pointer;
    margin: 0 auto;
    margin-top:1rem;
    border: none;
    outline: none;
    border-radius: 3px;
    background: #0e6af4;
    padding: 10px .75rem;
    color: #fff;
    border-color: #0e6af4;
    transition: background-color 0.5s;

}

.btn-reg:hover{
    background-color: #0b49a7;
}

.spanDef{
    color:#6185aa;
    font-style: italic;
    font-size: 20px;
}

.error {
    color:#9e1414;
    text-shadow: 1px 1px black;
    text-align: center;
    background: #310404;
    border-radius: .25rem;
    width: 60%;
    margin: 0 auto;
    padding: 5px;
    margin-top: .75em;
}

.errorOk{
    color: #069943;
    text-shadow: 1px 1px black;
    text-align: center;
    background: #012711;
    border-radius: .25rem;
    width: 60%;
    margin-top: 2em;
    padding: 5px;

}


#acc-wrapper{
    margin-bottom: 10rem;
    width: 100%;
    display: flex;
    justify-content: space-between;
    box-sizing: border-box;
}

.account-sidebar{
    width: 25%;
    background: rgb(31, 20, 14) none repeat scroll 0% 0%;
    border: 1px solid rgb(44, 31, 21);
}

.account-content-area{
    width: 70%;
    border : 1px solid #302720;
    background: rgb(31, 20, 14) none repeat scroll 0% 0%;
}


.acc-form input[type="text"], textarea{
    width: 100%;
    max-width: 500px;
}

.acc-form input[type="email"]{
    width: 100%;
    max-width: 500px;
```

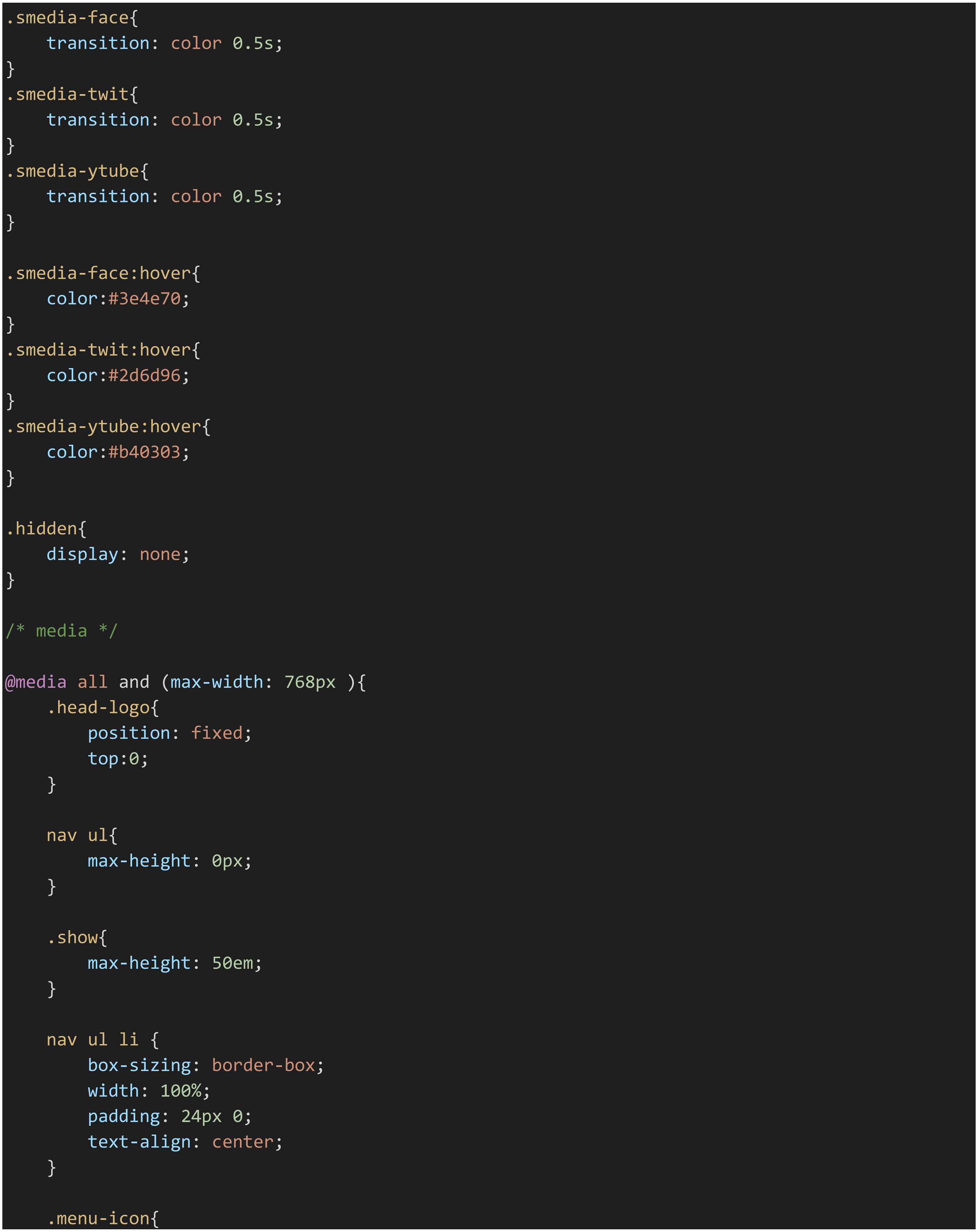
```
}

.acc-form input[type="password"]{
    width: 100%;
    max-width: 500px;
}

.acc-form ul li{
    text-align: left;
    max-width: 90%;

}

.acc-form ul li h2{
    text-align: center;

}

/* tables */

.table-user{
    width: 100%;
    margin-bottom: 20px;
    border-spacing: 0;
    border-collapse: collapse;
    color: #aa9c87;
}

.table-user tr:nth-child(even){
    background:#271b14;
}

.table-user th,td {
    padding: 8px;
    line-height: 1.42857143;
    border-top:1px solid #2c1f15;
}
.table-user th{
    text-transform: uppercase;
    font-family: 'Antika';
    color: #fff;
}

.btn{
    display: inline-block;
    font-weight: 400;
    text-align: center;
    vertical-align: middle;
```



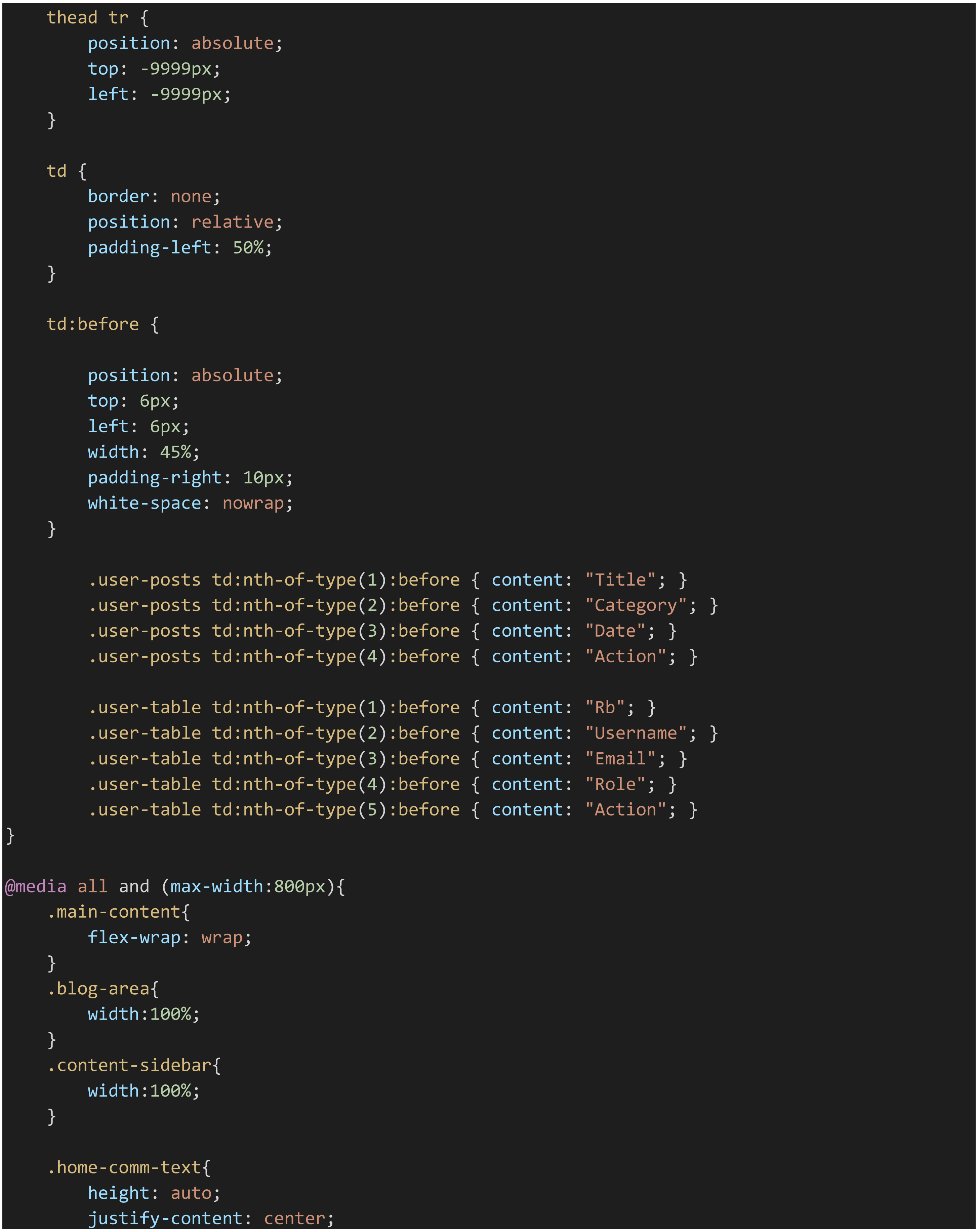
```
    -moz-user-select: none;
    padding: .375rem .75rem;
    font-size: 1rem;
    line-height: 1.5;
    border-radius: .25rem;
    cursor: pointer;

}

.edit{
    color: #212529;
    background-color: #ffc000;
    border-color: #ffc000;
    transition: background-color 0.5s;
}

.delete{
    color: #212529;
    background-color: #b31b41;
    border-color: #b11b1b;
    transition: background-color 0.5s;
}

.delete:hover{
    background-color: #9c1738;
    border-color: #9c1738;
}

.edit:hover{
    background-color: #ce9b03;
    border-color: #ce9b03;
}


.side-avatar{
    display: flex;
    flex-wrap: wrap;
    flex-direction: column;
    align-content: center;
    padding: 1rem;
    cursor: pointer;
}

.side-box img{
    width: 50%;
    border-radius: 50%;
    display: block;
    margin: 0 auto;
```

```
}

.side-box span{
    margin-top: .75em;
    text-align: center;
}



/* footer */

footer{
    margin-top: 25px;
    margin-bottom: 25px;
    display: block;
    box-sizing: border-box;
}

.follower{
    display: block;
    width: 100%;
    background: #1b100b;
    border-top: 1px solid #24190f;
    border-bottom: 1px solid #24190f;
    margin: 0;
    padding: 30px 0;
    box-sizing: border-box;
}

.follower-wrapper{
    margin: 0 auto;
    display: flex;
    -webkit-box-orient: horizontal;
    -webkit-box-direction: normal;
    flex-direction: row;box-sizing: border-box;
}

.follow-text{
    text-align: right;
    color: #6c5d48;
    margin-left: auto;
    box-sizing: border-box;
}

.follow-title{
    display: block;
    font-size: 20px;
    box-sizing: border-box;
    text-align: right;
    color: #6c5d48;
}

.follow-media{
    display: flex;
    -webkit-box-orient: horizontal;
    -webkit-box-direction: normal;
    flex-direction: row;
    -webkit-box-align: center;
    align-items: center;margin-left: 30px;
    margin-right: auto;
    box-sizing: border-box;
}

.follow-media a{
    font-size: 28px;
    margin-left: 25px;
    color: #8d8172;
    text-decoration: none;
    background-color: transparent;
    box-sizing: border-box;
}

.footer-row{
    display: flex;
}

.footer-left{
    display: flex;
    -webkit-box-orient: vertical;
    -webkit-box-direction: normal;
    flex-direction: column;
    -webkit-box-pack: justify;
    justify-content: space-between;
    color: #4b3f34;
    -webkit-box-flex: 0;
    flex: 0 0 25%;
    max-width: 50%;
    margin: auto 0;
    text-align: center;
}

.footer-left p a {
    color:#4b3f34;
}

.footer-left p a:hover{
    text-decoration: underline;
}

.footer-between{
    display: flex;
    -webkit-box-orient: horizontal;
    -webkit-box-direction: normal;
    flex-direction: row;
    -webkit-box-pack: justify;
    justify-content: space-between;
    -webkit-box-flex: 0;
    flex: 0 0 41.66666667%;
    max-width: 41.66666667%;
}

.footer-right{
    color: #4b3f34;
    box-sizing: border-box;
    text-align: center;
    -webkit-box-flex: 0;
    flex: 0 0 33.33333333%;
    max-width: 50%;
}

.footer-right p{
    margin-top: 1rem;
}

.logo-footer{
    background: url('../img/logo-footer.png') no-repeat;
    background-position: center;
    display: block;
    margin: 0 auto;
    height: 70px;
    width: 140px;
    box-sizing: border-box;
    text-align: center;
}

/* pagination */

.page-pag{
    border-top: 1px solid #302720;
    text-align: center;
}

#pag-list li{
    display: inline-block;
```

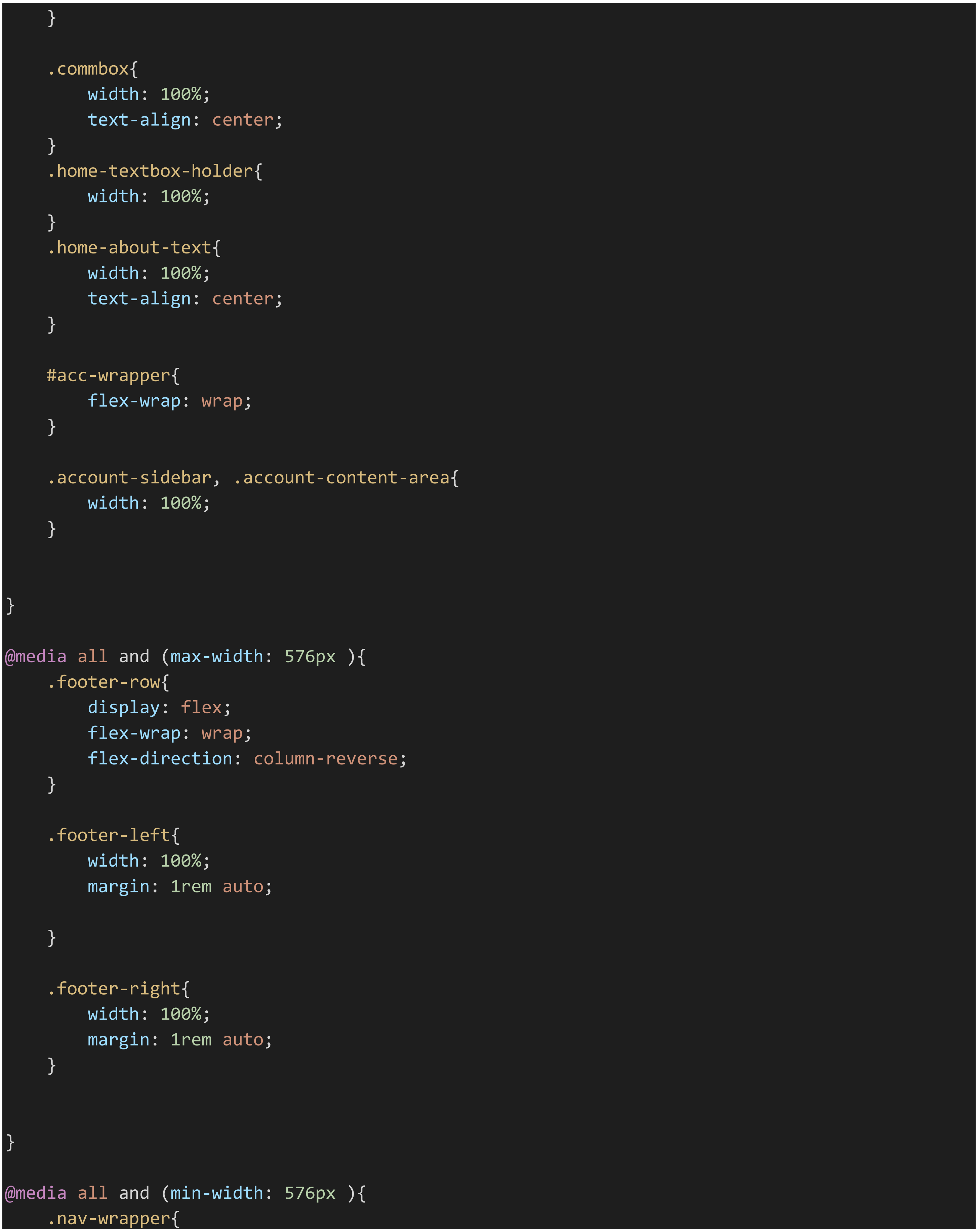
```
}

#pag-list li a{
    color: #fff;
    float: left;
    padding: 8px 16px;
    text-decoration: none;
    transition: background-color .3s;
    font-family: 'Antika';
    font-size: 18px;
}

#pag-list li a:hover:not(.active-page) {
    background-color: #3f3a32;
}

#pag-list li a.active-page {
    background-color: #ff8c00;
    color: #000;
}

/* Misc */

.active{
    border-bottom: 1px solid #ffc000;
}

.active a {
    color: #ffc000;
}

.tm{
    font-family: 'Antika';
    font-weight: bolder;
}

.acc-name{
    color:#ffc000;
}

.acc-role{
    text-transform: capitalize;
}

.acc-form{
    margin: 2rem auto;
}
```



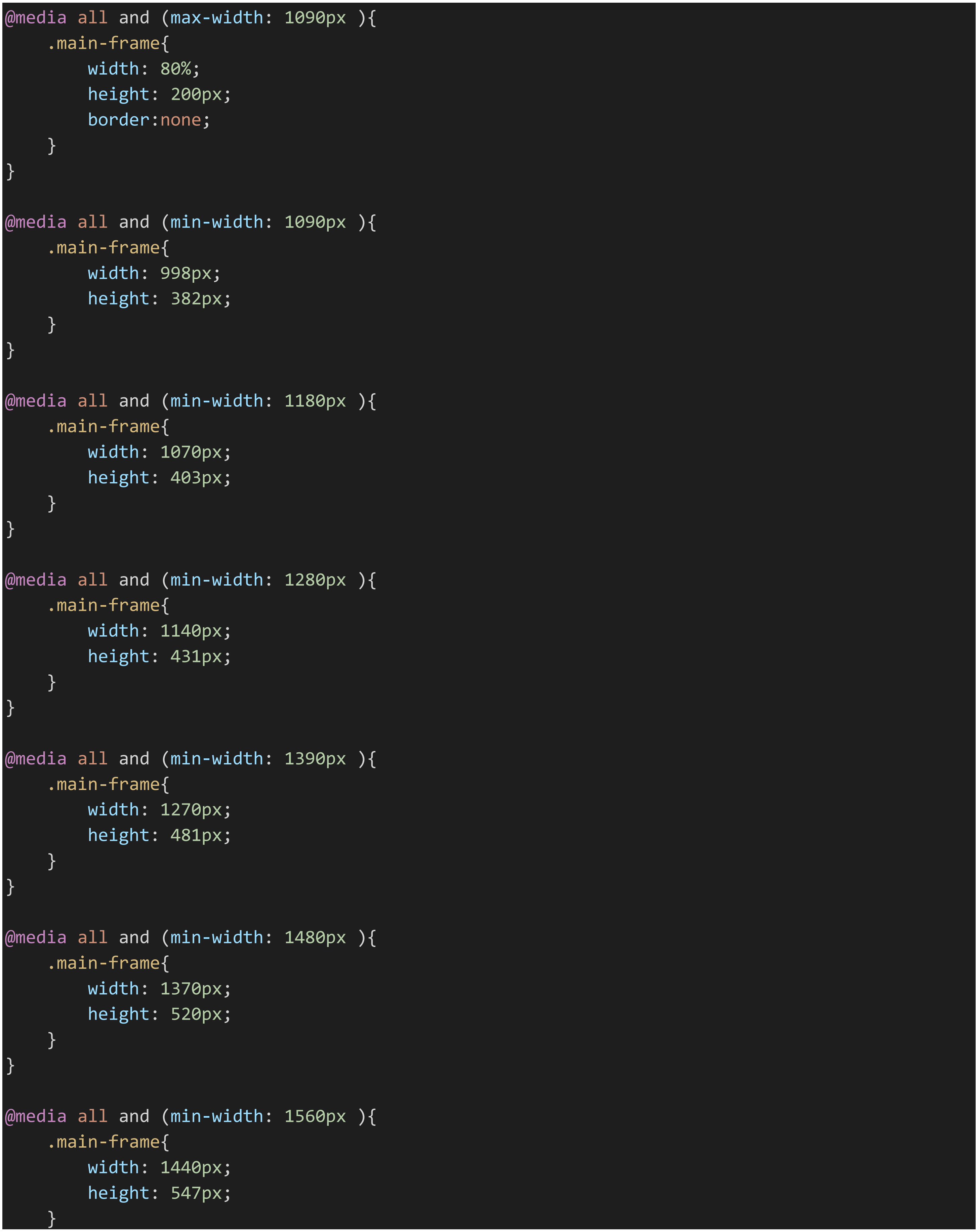
```css
.smedia-face{
    transition: color 0.5s;
}
.smedia-twit{
    transition: color 0.5s;
}
.smedia-ytube{
    transition: color 0.5s;
}

.smedia-face:hover{
    color:#3e4e70;
}
.smedia-twit:hover{
    color:#2d6d96;
}
.smedia-ytube:hover{
    color:#b40303;
}

.hidden{
    display: none;
}

/* media */

@media all and (max-width: 768px ){
    .head-logo{
        position: fixed;
        top:0;
    }

    nav ul{
        max-height: 0px;
    }

    .show{
        max-height: 50em;
    }

    nav ul li {
        box-sizing: border-box;
        width: 100%;
        padding: 24px 0;
        text-align: center;
    }

    .menu-icon{
        display: block;
    }

    .nav-menu{
        margin-left: 0;
    }

    .blog-post-inner{
        flex-wrap: wrap;
    }

    .post-about{
        width: 90%;
        margin: 0 auto;
        margin-bottom: 2rem;
    }
    .post-member{
        width: 100%;
        margin-bottom: 1rem;
    }
    .post-member img{
        width: 30%;
    }
    .post-content{
        margin-top: 2rem;
    }

    .my-acc{
        display: block;
    }

    .acc-nav{
        display: none;
    }

    #sort-ul li{
        display: block;
    }

}

@media only screen and (max-width: 760px), (min-device-width: 768px) and (max-device-width: 1024px)  {

    table, thead, tbody, th, td, tr {
        display: block;
    }
```



```css
    thead tr {
        position: absolute;
        top: -9999px;
        left: -9999px;
    }

    td {
        border: none;
        position: relative;
        padding-left: 50%;
    }

    td:before {

        position: absolute;
        top: 6px;
        left: 6px;
        width: 45%;
        padding-right: 10px;
        white-space: nowrap;
    }

        .user-posts td:nth-of-type(1):before { content: "Title"; }
        .user-posts td:nth-of-type(2):before { content: "Category"; }
        .user-posts td:nth-of-type(3):before { content: "Date"; }
        .user-posts td:nth-of-type(4):before { content: "Action"; }

        .user-table td:nth-of-type(1):before { content: "Rb"; }
        .user-table td:nth-of-type(2):before { content: "Username"; }
        .user-table td:nth-of-type(3):before { content: "Email"; }
        .user-table td:nth-of-type(4):before { content: "Role"; }
        .user-table td:nth-of-type(5):before { content: "Action"; }
}

@media all and (max-width:800px){
    .main-content{
        flex-wrap: wrap;
    }
    .blog-area{
        width:100%;
    }
    .content-sidebar{
        width:100%;
    }

    .home-comm-text{
        height: auto;
        justify-content: center;
```



```
    }

    .commbox{
        width: 100%;
        text-align: center;
    }
    .home-textbox-holder{
        width: 100%;
    }
    .home-about-text{
        width: 100%;
        text-align: center;
    }

    #acc-wrapper{
        flex-wrap: wrap;
    }

    .account-sidebar, .account-content-area{
        width: 100%;
    }


}

@media all and (max-width: 576px ){
    .footer-row{
        display: flex;
        flex-wrap: wrap;
        flex-direction: column-reverse;
    }

    .footer-left{
        width: 100%;
        margin: 1rem auto;

    }

    .footer-right{
        width: 100%;
        margin: 1rem auto;
    }


}

@media all and (min-width: 576px ){
    .nav-wrapper{
        max-width: 540px;
    }
    .container{
        max-width: 540px;
    }
    .content-wrapper{
        max-width: 540px;
    }
}

@media all and (min-width: 768px ){
    .nav-wrapper{
        max-width: 720px;
    }

    .container{
        max-width: 720px;
    }
    .content-wrapper{
        max-width: 720px;
    }

}
@media all and (min-width: 992px ){

    .nav-wrapper{
        max-width: 960px;
    }
    .container{
        max-width: 960px;
    }
    .content-wrapper{
        max-width: 960px;
    }
}

@media all and (min-width: 1200px ){
    .nav-wrapper{
        max-width: 1140px;
    }
    .container{
        max-width: 1140px;
    }
    .content-wrapper{
        max-width: 1140px;
    }
}
```



```
@media all and (max-width: 1090px ){
    .main-frame{
        width: 80%;
        height: 200px;
        border:none;
    }
}

@media all and (min-width: 1090px ){
    .main-frame{
        width: 998px;
        height: 382px;
    }
}

@media all and (min-width: 1180px ){
    .main-frame{
        width: 1070px;
        height: 403px;
    }
}

@media all and (min-width: 1280px ){
    .main-frame{
        width: 1140px;
        height: 431px;
    }
}

@media all and (min-width: 1390px ){
    .main-frame{
        width: 1270px;
        height: 481px;
    }
}

@media all and (min-width: 1480px ){
    .main-frame{
        width: 1370px;
        height: 520px;
    }
}

@media all and (min-width: 1560px ){
    .main-frame{
        width: 1440px;
        height: 547px;
    }
}

@media all and (min-width: 1635px ){
    .main-frame{
        width: 1520px;
        height: 579px;
    }
}

@media all and (min-width: 1610px ){
    .main-frame{
        width: 1610px;
        height: 626px;
    }
}
```