Visoka škola strukovnih studija za informacione i komunikacione tehnologije

https://gigatronic-shop.000webhostapp.com/

Predmet: Web programiranje PHP 1

Student: Marko Gačanović 38/17

Beograd, mart 2020

1 Uvod

1.1 Korišćeni programski jezici i tehnologije

Korišćeni programski jezici su: HTML, CSS, JavaScript, jQuery, Bootstrap, JSON, AJAX, PHP, MySQL

Sajt je rađen u Visual Studio Codu

1.2 Opis funkcionalnosti

1.2.1 Admin Panel

Adminu je omogućeno dodavanje novih proizvoda, kao i brisanje i ažuriranje. Admin takođe može da unosi nove ankete, da aktivira neke od postojećih kao i da vidi rezultate za sve ankete.

1.2.2 Proizvodi

Sav sadržaj stranice 'Products' je učitan dinamički putem ajax-a iz baze podataka. Moguće je pogledati detalje o proizvodu čim se otvori modal. Na stranici postoji dinamička dropdown lista za filtriranje proizvoda po kategorijama. Takođe je moguće vršiti pretragu po karakterima u nazivu proizvoda. Odnosno kada se klikne na dugme 'Search' u pretrazi, koja se nalazi u meniju, prikazaće se svi proizvodi koji sadrže u svom imenu karaktere koje smo ukucali za pretragu.

1.2.3 Straničenje

Straničenje se koristi na stranici za proizvode, prikazuju se 4 proizvoda po stranici.

1.2.4 Kontakt stranica

Na kontakt stranici korisnik može poslati poruku administratoru sajta. Neophodno je samo ispravno da popuni podatke.

1.2.5 Registracija i logovanje

Provera podataka prilikom registracije je urađena na serverskoj i klijentskoj strani, odnosno ajax-om. Prilikom uspešne registracije korisnik dobija ulogu korisnika, koju može da menja administrator. Provera podataka prilikom logovanja vrši se na serverskoj strani i ako podaci nisu dobri pravi se sesija koja obaveštava korisnika gde je pogrešio prilikom logovanja. Ako su email i lozinka ispravni, tada se pravi sesija koja sadrži sve podatke o korisniku.

1.2.6 Dinamički meni

Prilikom učitavanja stranice prikazuje se dinamički meni, koji prikazuje odgovarajuće linkove za odgovarajuće grupe korisnika: admine, neautorizovane korisnike i autorizovane korisnike. Sve u zavisnosti od aktivnosti i stanja sesija.

1.2.7 Korpa

Sistem korpe realizovan je pomoću localStorage-a. Korisnik u korpu može da doda više proizvoda, kao i vište proizvoda istog tipa. Takođe omogućeno mu je da briše proizvode iz korpe.

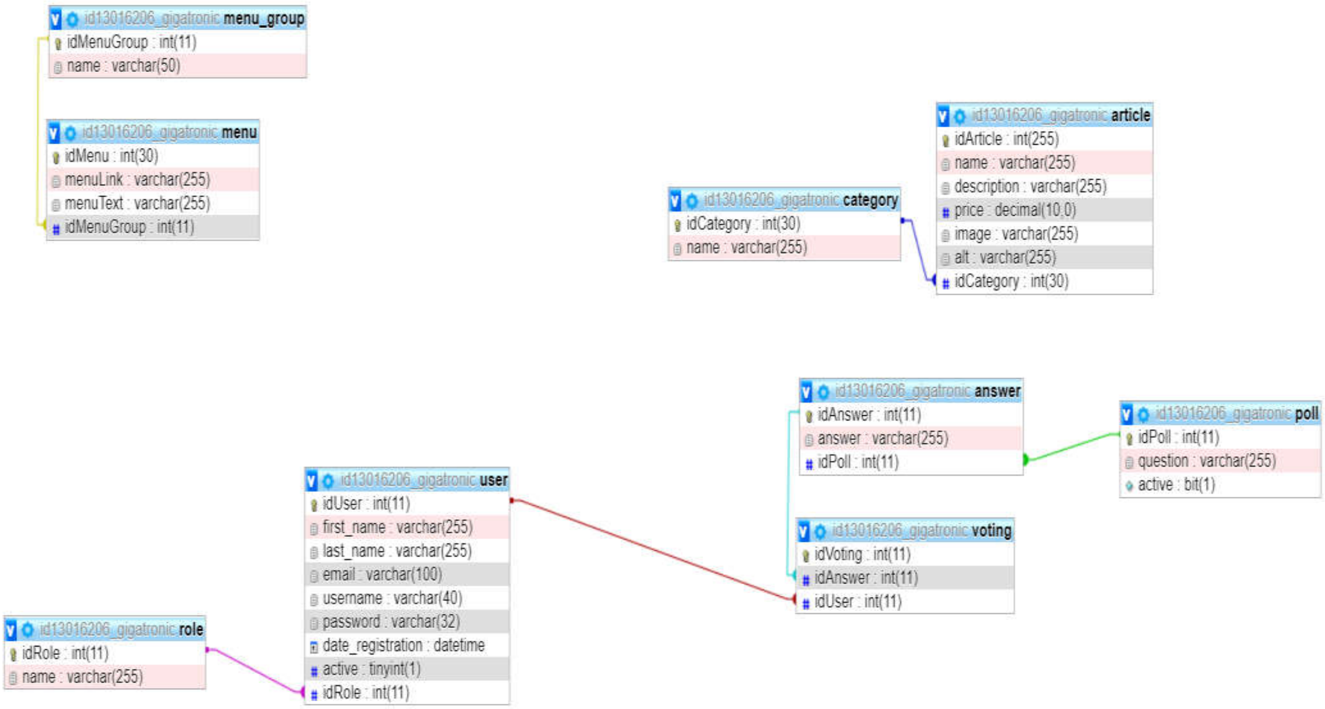
1.2.8 Anketa

Postoji jedna aktivna anketa na sajtu. Na tu anketu mogu da glasaju samo autorizovani korisnici i to samo jednom. Anketa ima jedno pitanje i više ponuđenih odgovora. Kada korisnik glasa prikazuje mu se tablea sa statistikom o broju glasova za sve ponuđene odgovore.

2 Organizacija

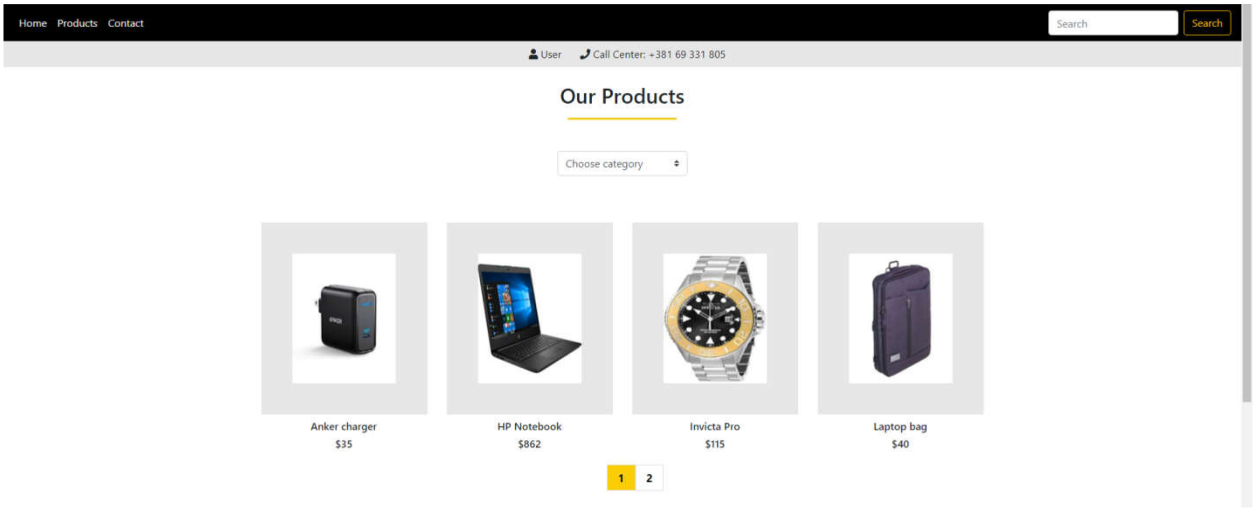
2.1 Organizaciona šema

2.1.1 Šema baze



2.2 Stranice

2.2.1 Products



Our Products

Anker charger
$35

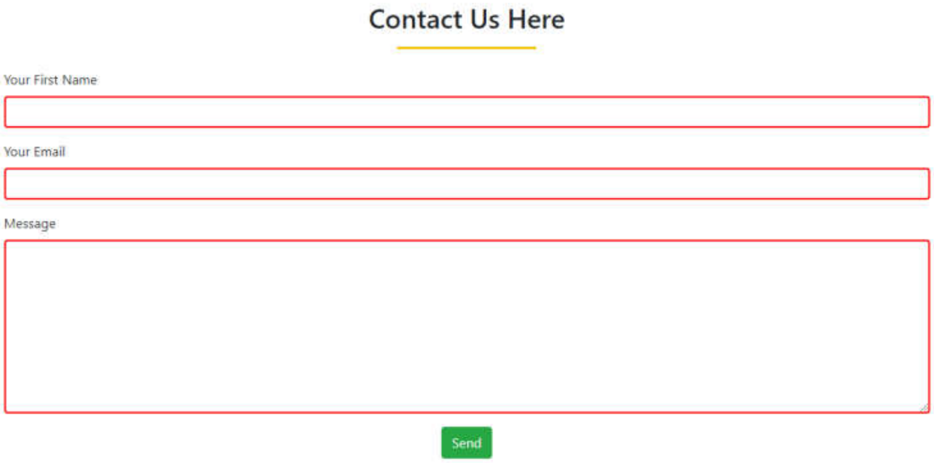
HP Notebook
$862

Invicta Pro
$115

Laptop bag
$40

Na ovoj stranici je prikaz svih proizvoda. Postoji dinamička dropdown lista za filtriranje proizvoda po kategorijama. Takođe je moguće vršiti pretragu po karakterima u nazivu proizvoda. Odnosno kada se klikne na dugme 'Search' u pretrazi, koja se nalazi u meniju, prikazaće se svi proizvodi koji sadrže u svom imenu karaktere koje smo ukucali za pretragu. Urađeno je straničenje, odnosno prikaz 4 proizvoda po strani.

2.2.2 Contact



Contact Us Here

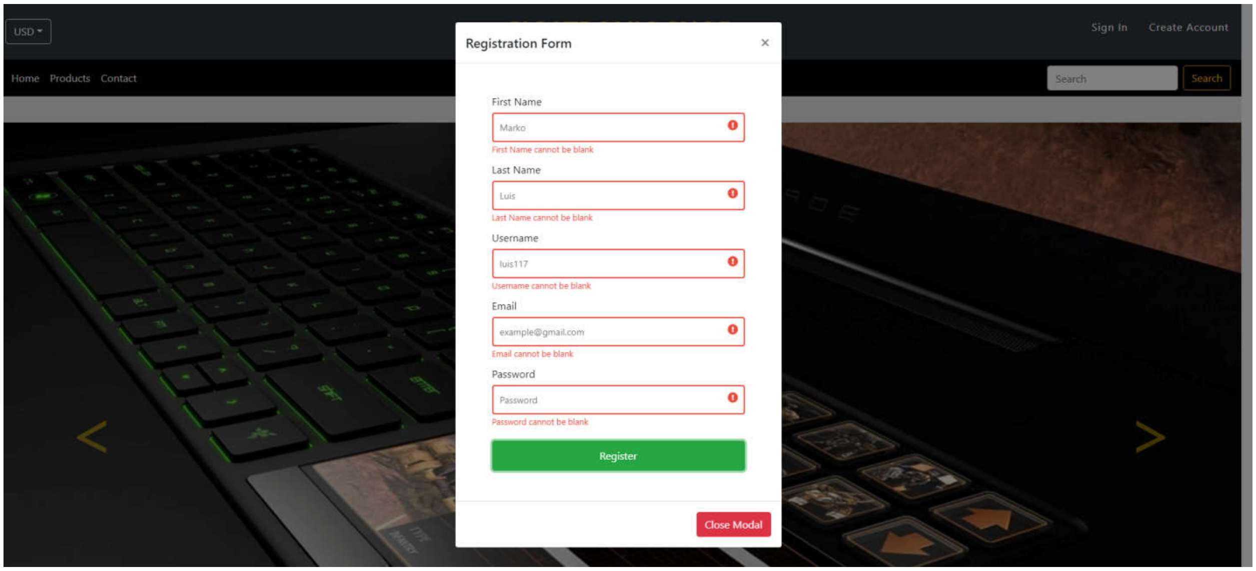
Your First Name

Your Email

Message

Na ovoj stranici korisnici imaju mogućnost slanja poruke administratoru. Provera se vrši klijentskim i serverskim regularnim izrazima. Ako su podaci ispravni, korisnik se redirektuje na stranicu gde mu piše obaveštenje da je uspešno poslat email.

2.2.3 Register



Registration Form

First Name

Marko

First Name cannot be blank

Last Name

Luis

Last Name cannot be blank

Username

luis117

Username cannot be blank

Email

example@gmail.com

Email cannot be blank

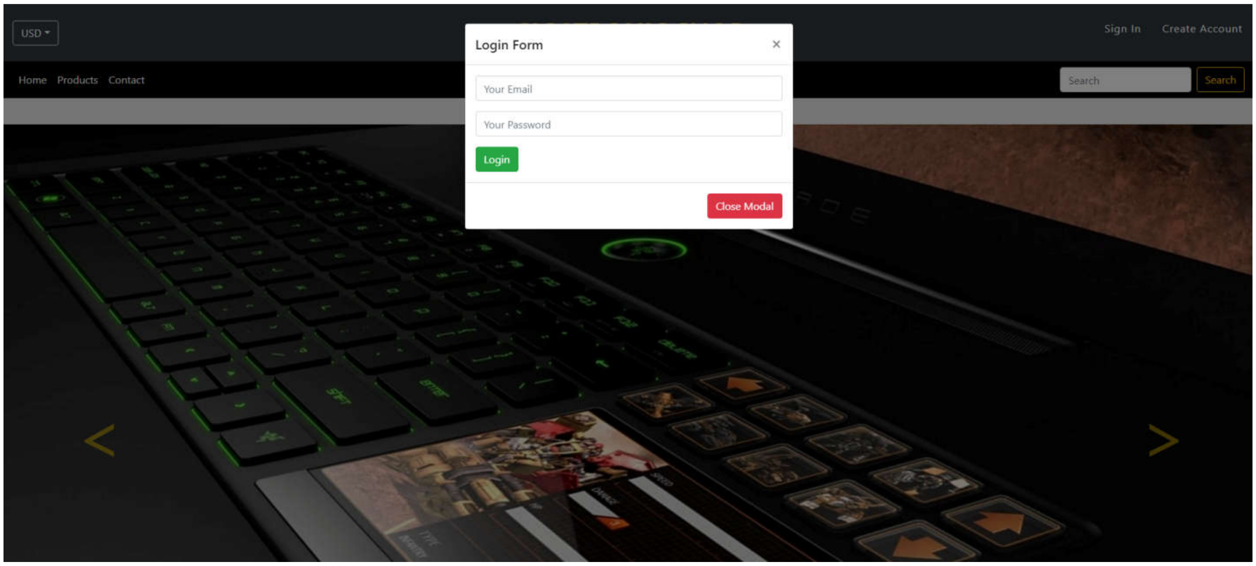
Password

Password

Password cannot be blank

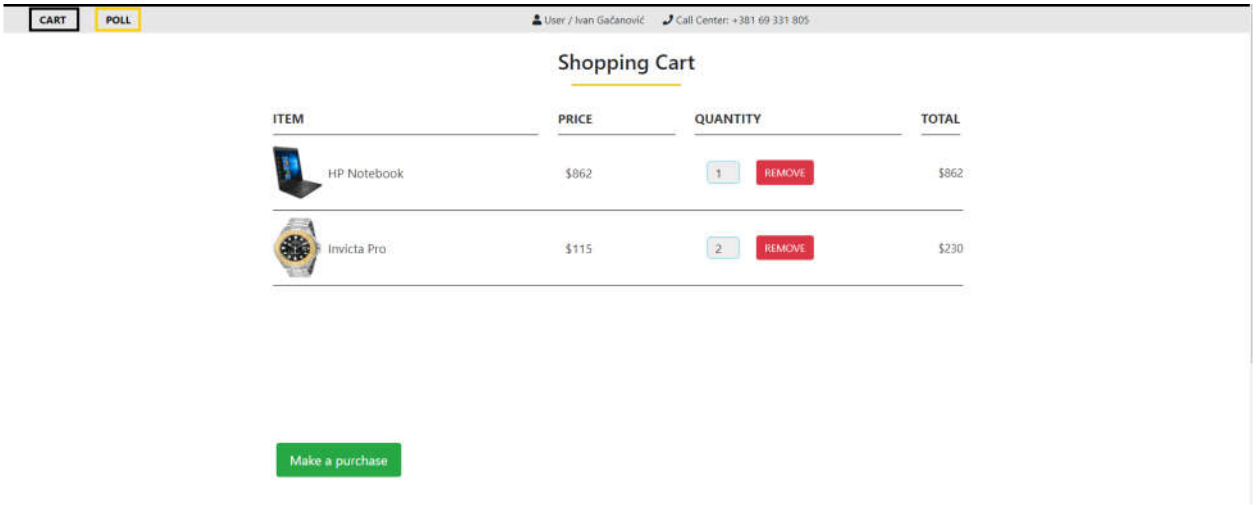
Link za registraciju se nalazi na svakoj strani i pojavljuje se u vidu modala. Provera se vrši klijentskim i serverskim regularnim izrazima. Klikom na dugme korisnik zna da li je ispravno popunio svako polje.

2.2.4 Login



Link za logovanje se takođe nalazi na svakoj strani i pojavljuje se u vidu modala. Prilikom uspešnog logovanja, korisniku se dodeljuje sesija i korisnik se automatski redirektuje na početnu stranicu, dok se admin redirektuje na stranicu sa svim korisnicima. Greške se beleže u sesijama i ispisuju ispod forme.

2.2.5 Cart

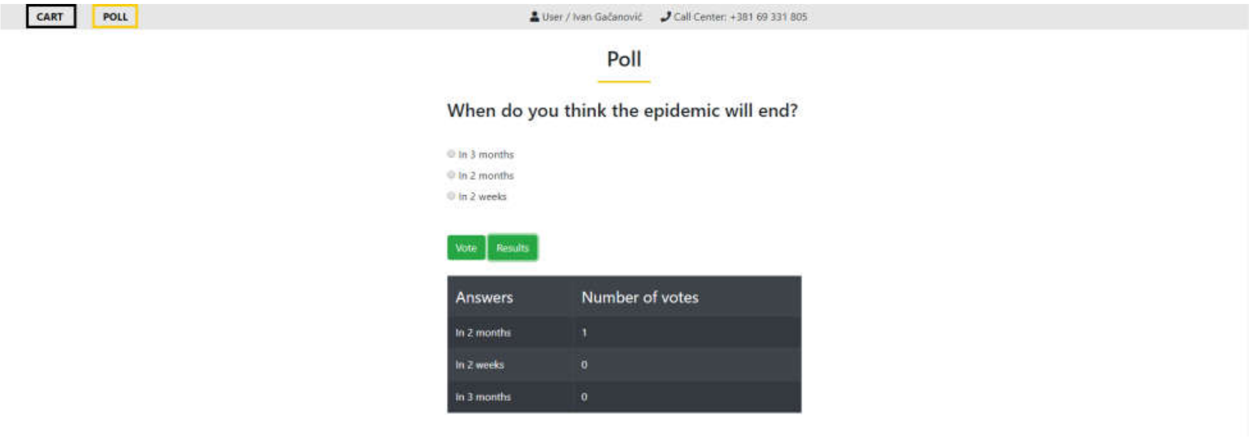


Shopping Cart

ITEM	PRICE	QUANTITY		TOTAL
HP Notebook	$862	1	REMOVE	$862
Invicta Pro	$115	2	REMOVE	$230

Ovoj stranici ima pravo pristupa samo autorizovani korisnik koji nije admin. U korpu se može dodati više proizvoda. Podaci iz korpe se čuvaju u localStorage-u. Korpa se prilikom kupovine prazni i spremna je za novu kupovinu.

2.2.6 Poll



Poll

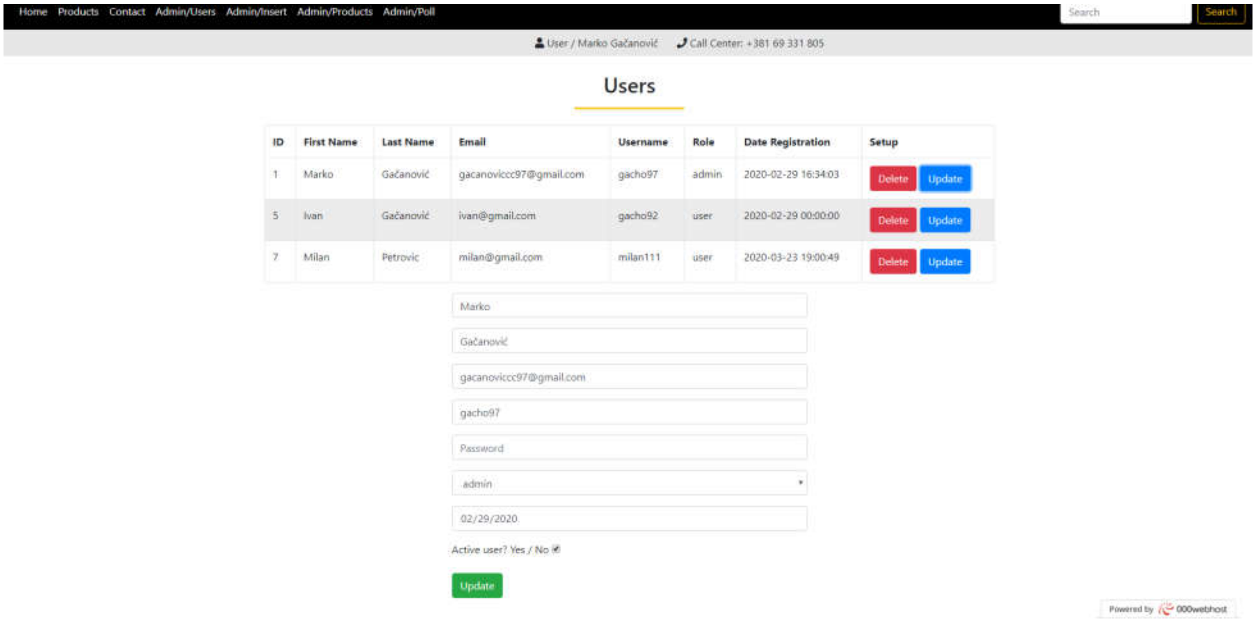
When do you think the epidemic will end?

○ In 3 months
○ In 2 months
○ In 2 weeks

Answers	Number of votes
In 2 months	1
In 2 weeks	0
In 3 months	0

Pravo glasanja imaju samo autorizovani korisnici koji nisu admini. Pravo glasa imaju samo jednom i prilikom glasanja prikazuje im se statistika ukupnog glasanja. Ako je korisnik već jednom glasao, prikazuje mu se poruka da je to već učinio i da ne može opet.

2.2.7 Admin/Users



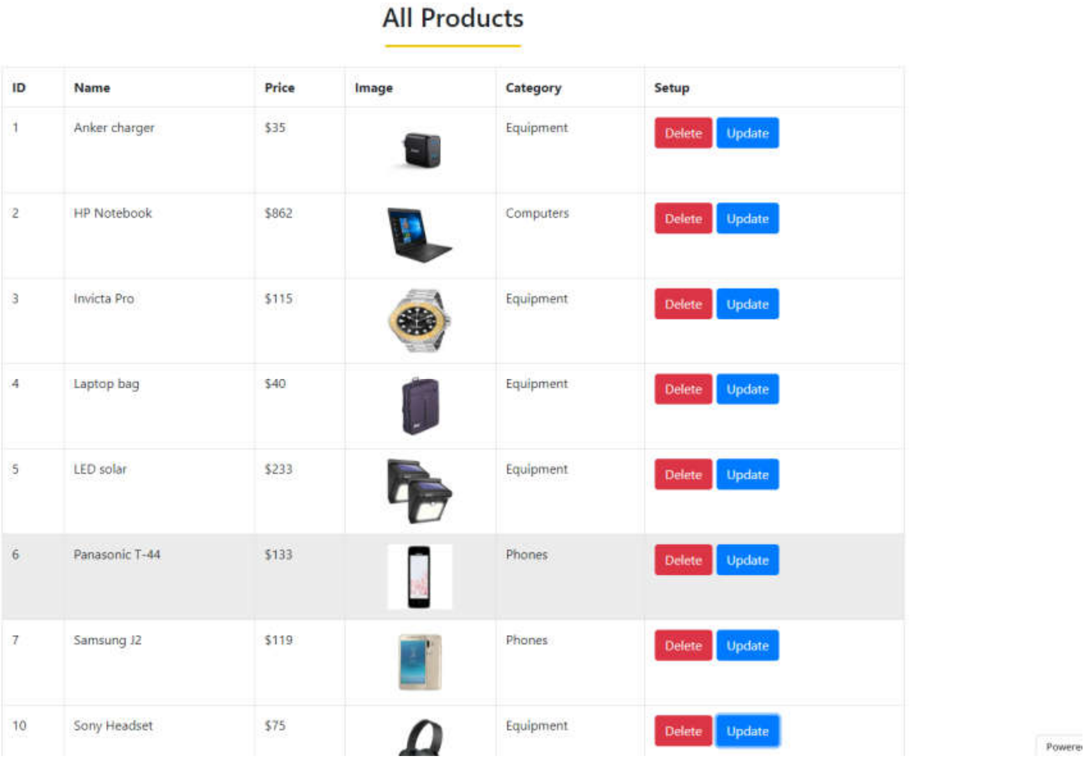
Users

ID	First Name	Last Name	Email	Username	Role	Date Registration	Setup
1	Marko	Gačanović	gacanoviccc97@gmail.com	gacho97	admin	2020-02-29 16:34:03	Delete Update
5	Ivan	Gačanović	ivan@gmail.com	gacho92	user	2020-02-29 00:00:00	Delete Update
7	Milan	Petrovic	milan@gmail.com	milan111	user	2020-03-23 19:00:49	Delete Update

Marko
Gačanović
gacanoviccc97@gmail.com
gacho97
Password
admin
02/29/2020
Active user? Yes / No ☑

Ovoj stranici ima pravo pristupa samo admin. Omogućeno je ažuriranje i brisanje korisnika.

2.2.8 Admin/Products

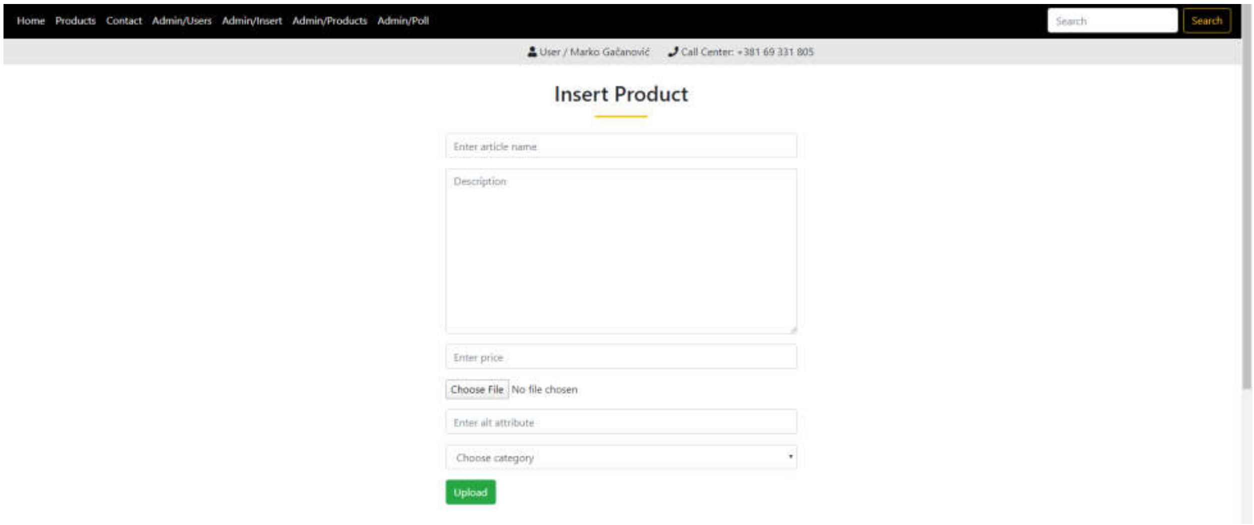


All Products

ID	Name	Price	Image	Category	Setup
1	Anker charger	$35		Equipment	Delete Update
2	HP Notebook	$862		Computers	Delete Update
3	Invicta Pro	$115		Equipment	Delete Update
4	Laptop bag	$40		Equipment	Delete Update
5	LED solar	$233		Equipment	Delete Update
6	Panasonic T-44	$133		Phones	Delete Update
7	Samsung J2	$119		Phones	Delete Update
10	Sony Headset	$75		Equipment	Delete Update

Ovoj stranici ima pravo pristupa samo admin. Omogućeno je ažuriranje I brisanje postojećih proizvoda. Može da menja sledeće podatke: naziv, cenu, opis, sliku I kategoriju.

2.2.9 Admin/Insert



User / Marko Gaćanović   Call Center: +381 69 331 805

Insert Product

Enter article name

Description

Enter price

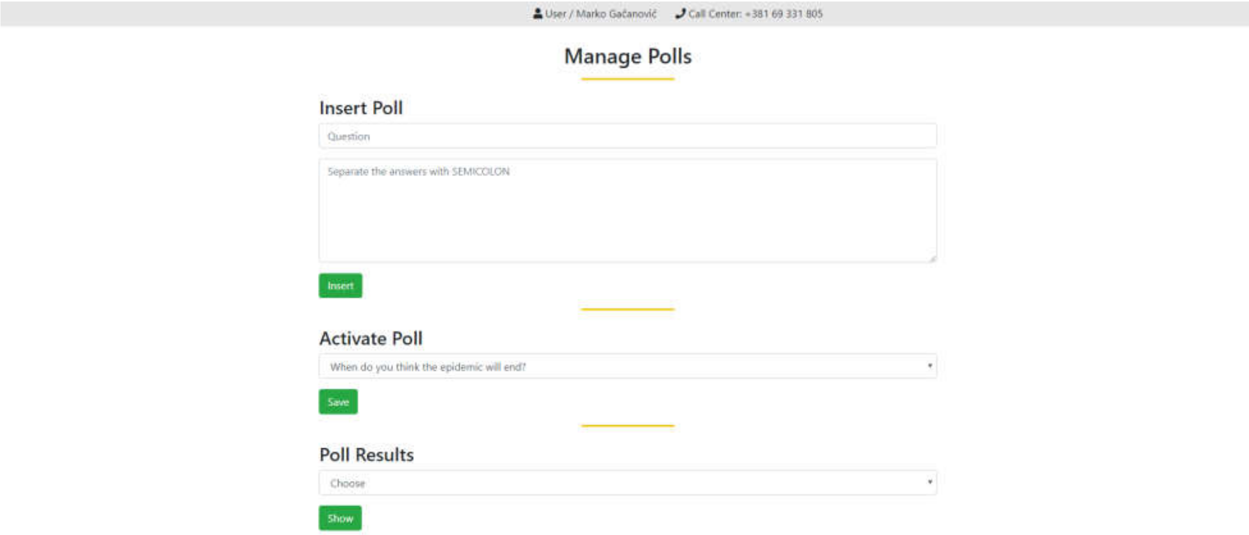
Choose File No file chosen

Enter alt attribute

Choose category

Ovoj stranici ima pravo pristupa samo admin. Admin ima mogućnost dodavanja proizvoda u postojeće kategorije.

2.2.10 Admin/Poll



User / Marko Gačanović   Call Center: +381 69 331 805

Manage Polls

Insert Poll

Question

Separate the answers with SEMICOLON

Insert

Activate Poll

When do you think the epidemic will end?

Save

Poll Results

Choose

Show

Ovoj stranici ima pravo pristupa samo admin. Admin ima mogućnost da dodaje nove ankete, aktivira neke od postojećih kao I da vidi rezultate za te postojeće ankete.

3 Kodovi

3.1 Root

3.1.1 index.php

```
<?php
session_start();
require_once "config/connection.php";

require "models/functions.php";
require "models/users/functions.php";
require "models/products/functions.php";
require "models/polls/functions.php";

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

if (isset($_GET['page'])) {
    $page = isset($_GET['page']) ? $_GET['page'] : null;
    switch ($page) {
        case "home":
            include "views/pages/home.php";
            break;
        case "products":
            include "views/pages/products.php";
            break;
        case "contact":
            include "views/pages/contact.php";
            break;
        case "cart":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'user') {
                include "views/pages/cart.php";
            }
            break;
        case "poll":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'user') {
                include "views/pages/poll.php";
            }
            break;
        case "adminUsers":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'admin') {
                include "views/pages/admin/adminUsers.php";
            }
            break;
        case "adminInsertProduct":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'admin') {
                include "views/pages/admin/adminInsertProdcut.php";
            }
            break;
        case "adminProducts":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'admin') {
                include "views/pages/admin/adminProducts.php";
            }
            break;
        case "adminPoll":
            if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'admin') {
                include "views/pages/admin/adminPoll.php";
            }
            break;
    }
} else {
    include "views/pages/home.php";
}

include "views/fixed/modals.php";
include "views/fixed/footer.php";
```

3.1.2 thanks.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Thanks</title>
</head>
<body>
    <h1>THANK YOU FOR CONTACTING US. <br/> WE WILL ANSWER YOU AS SOON AS POSSIBLE.</h1
>
```

```
</body>
</html>
```

3.2 Views

3.2.1 Fixed

3.2.1.1 head.php

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

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <link rel="shortcut icon" href="assets/img/gigatronic.ico" type="image/x-icon" />
    <!-- Custom Stylesheet -->
    <link rel="stylesheet" type="text/css" href="assets/css/style.css" />
    <!-- Bootstrap CDN -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/cs
s/bootstrap.min.css" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymo
us" />
    <!-- Font Awesome CDN -->
    <script src="https://kit.fontawesome.com/2c9e2408b5.js" crossorigin="anonymous"></
script>
    <!-- jQuery CDN -->
    <script src="https://code.jquery.com/jquery-3.4.1.js" integrity="sha256-
WpOohJOqMqqyKL9FccASB9O0KwACQJpFTUBLTYOVvVU=" crossorigin="anonymous"></script>
    <title>Gigatronic Shop</title>
</head>

<body>
```

3.2.1.2 header.php

```
<!-- Header -->
<header>
    <div class="container-fluid">
        <div class="row bg-dark">
            <div class="col-12 col-sm-12 col-md-4">
                <div class="btn-group">
                    <button class="btn border dropdown-toggle my-2 my-md-4 text-
white" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">USD</button>
                    <div class="dropdown-menu">
                        <a href="#" class="dropdown-item">EU - Euro</a>
                    </div>
                </div>
            </div>
            <div class="col-12 col-md-4 text-center">
                <a href="#">
```

```
                    <h1 class="my-md-3 site-title primary-color text-
uppercase">Gigatronic Shop</h1>
                </a>
            </div>
            <div class="col-12 col-md-4 text-right">
                <p class="my-md-4 header-links">
                    <?php if (!isset($_SESSION['user'])) : ?>
                        <a href="#" class="px-4" data-target="#loginModal" data-
toggle="modal">Sign In</a>
                        <a href="#" class="px-1" data-
target="#registrationModal" data-toggle="modal">Create Account</a>
                    <?php else : ?>
                        <a href="models/logout.php">Logout</a>
                    <?php endif; ?>
                </p>
            </div>
        </div>
    </div>

    <div class="container-fluid p-0">
        <nav class="navbar navbar-expand-lg navbar-light bg-darkest">
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-
label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav">
                    <?php
                    $menuVisitors = getMenuForAllUsers();
                    foreach ($menuVisitors as $mv) :
                    ?>
                        <li class="nav-item active">
                            <a class="nav-link" href="index.php?page=<?= $mv-
>menuLink ?>"><?= $mv->menuText ?></a>
                        </li>
                    <?php endforeach; ?>
                    <?php if (isset($_SESSION['user'])) :
                        if (($_SESSION['user']->roleName === 'admin')) :
                            $menuAdmin = getMenuForAdmin();
                            foreach ($menuAdmin as $ma) :
                    ?>
                                <li class="nav-item active">
                                    <a class="nav-link" href="index.php?page=<?= $ma-
>menuLink ?>"><?= $ma->menuText ?></a>
                                </li>
                            <?php endforeach; ?>
                        <?php endif; ?>
                    <?php endif; ?>
                </ul>
            </div>
            <form class="form-inline my-2 my-lg-0">
                <input class="form-control mr-sm-
2" type="search" placeholder="Search" id="searchInput" aria-label="Search">
```

```
                <button class="btn btn-outline-warning my-2 my-sm-0" type="submit" id="btnSearch">Search</button>
            </form>
        </nav>
    </div>

    <div class="container-fluid gray-container py-2 ">
        <div class="row">
            <div class="col-5">
                <?php
                if (isset($_SESSION['user'])) :
                    if ($_SESSION['user']->roleName === 'user') :
                        $menuUser = getMenuForAuthorizedUsers();
                        foreach ($menuUser as $mu) :
                ?>
                            <a href="index.php?page=<?= $mu->menuLink ?>" class="ml-4">
                                <span class="text-uppercase font-weight-bold"><?= $mu->menuText ?></span>
                            </a>

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

            <div class="col-7">
                <i class="fa fa-user"></i>
                <span class="mr-4">
                    User
                    <?php
                    if (isset($_SESSION['user'])) {
                        echo "/ " . $_SESSION['user']->first_name . " " . $_SESSION['user']->last_name;
                    }
                    ?>
                </span>

                <i class="fa fa-phone"></i>
                <span>Call Center: </span>
                <span>+381 69 331 805</span>
            </div>
        </div>
    </div>
</header>
<!-- End of Header -->
```

3.2.1.3 modals.php

```
<!-- Start of product modal -->
<div id="dataModal" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
```

```
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal">&times;</button>
            </div>
            <div class="modal-body" id="article_detail">

            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-danger" data-dismiss="modal">Close</button>
            </div>
        </div>
    </div>
</div>
<!-- End of product modal -->

<!-- Start of registration modal -->
<div class="modal fade" id="registrationModal" tabindex="-1" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content registration">
            <div class="modal-header">
                <h5 class="modal-title">Registration Form</h5>
                <button type="button" class="close" data-dismiss="modal">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                <form id="formRegister" class="form">
                    <div class="form-group">
                        <label for="firstName">First Name</label>
                        <input type="text" placeholder="Marko" id="firstName" />
                        <i class="fas fa-check-circle"></i>
                        <i class="fas fa-exclamation-circle"></i>
                        <small>Error message</small>
                    </div>
                    <div class="form-group">
                        <label for="lastName">Last Name</label>
                        <input type="text" placeholder="Luis" id="lastName" />
                        <i class="fas fa-check-circle"></i>
                        <i class="fas fa-exclamation-circle"></i>
                        <small>Error message</small>
                    </div>
                    <div class="form-group">
                        <label for="username">Username</label>
                        <input type="text" placeholder="luis117" id="username" />
                        <i class="fas fa-check-circle"></i>
                        <i class="fas fa-exclamation-circle"></i>
                        <small>Error message</small>
                    </div>
                    <div class="form-group">
                        <label for="email">Email</label>
                        <input type="text" placeholder="example@gmail.com" id="email" />

                        <i class="fas fa-check-circle"></i>
                        <i class="fas fa-exclamation-circle"></i>
                        <small>Error message</small>
                    </div>
                    <div class="form-group">
                        <label for="password">Password</label>
                        <input type="password" placeholder="Password" id="password" />
                        <i class="fas fa-check-circle"></i>
                        <i class="fas fa-exclamation-circle"></i>
                        <small>Error message</small>
                    </div>
                    <button class="btn btn-success" id="btnRegister">Register</button>
                </form>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-danger" data-
dismiss="modal">Close Modal</button>
            </div>
        </div>
    </div>
</div>
<!-- End of registration modal -->

<!-- Start of login modal -->
<div class="modal fade" id="loginModal" tabindex="-1" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title">Login Form</h5>
                <button type="button" class="close" data-dismiss="modal">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                <form action="models/login.php" method="POST">
                    <div class="form-group">
                        <input type="text" name="tbEmail" placeholder="Your Email" cla
ss="form-control">
                    </div>
                    <div class="form-group">
                        <input type="password" name="tbPassword" placeholder="Your Pas
sword" class="form-control">
                    </div>

                    <input type="submit" value="Login" name="btnLogin" class="btn btn-
success">
                </form>
            </div>
            <div class="modal-footer">
                <div class="modal-footer-errors">
                    <?php
                    if (isset($_SESSION['errors'])) {
                        $errors = $_SESSION['errors'];
                        foreach ($errors as $one) {
                            echo $one . "<br/>";
                        }
                        unset($_SESSION['errors']);
                    }
                    ?>
                </div>
                <button type="button" class="btn btn-danger" data-
dismiss="modal">Close Modal</button>
            </div>
        </div>
    </div>
</div>
<!-- End of login modal -->
```

3.2.1.4 footer.php

```
<footer class="bg-darkest">
    <div class="container-fluid">
        <div class="container">
            <div class="row">
                <div class="col-4">
                    <ul class="footer-links">
                        <li>
                            <h3>User Account</h3>
                        </li>
                        <?php if (!isset($_SESSION['user'])) : ?>
                            <li>
                                <i class="fa fa-angle-right"></i>
                                <a href="#" data-target="#loginModal" data-
toggle="modal">Already registered? Login here</a>
                            </li>
                            <li>
                                <i class="fa fa-angle-right"></i>
                                <a href="#" data-target="#registrationModal" data-
toggle="modal">Register</a>
                            </li>
                        <?php else : ?>
                            <li>
                                <i class="fa fa-angle-right"></i>
                                <a href="models/logout.php">Logout</a>
                            </li>
                        <?php endif; ?>
                    </ul>
                </div>
                <div class="col-4">
                    <ul class="footer-links social-media">
                        <li>
                            <h3>Social Media</h3>
                        </li>
                        <li>
                            <a href="http://www.facebook.com">
                                <i class="fab fa-facebook fa-2x"></i>
                            </a>
                        </li>
                        <li>
```

```
                            <a href="http://www.twitter.com">
                                <i class="fab fa-twitter fa-2x"></i>
                            </a>
                        </li>
                        <li>
                            <a href="https://www.linkedin.com/in/marko-gacanovic-
4a133016a/">
                                <i class="fab fa-linkedin fa-2x"></i>
                            </a>
                        </li>
                    </ul>
                </div>
                <div class="col-4">
                    <ul class="footer-links">
                        <li>
                            <h3>About Author</h3>
                        </li>
                        <li>
                            <i class="fa fa-angle-right"></i>
                            <a href="https://www.linkedin.com/in/marko-gacanovic-
4a133016a/">
                                Link to bio
                            </a>
                        </li>
                        <li>
                            <i class="fa fa-angle-right"></i>
                            <a href="#">
                                Documentation
                            </a>
                        </li>
                    </ul>
                </div>
            </div>
        </div>
    </div>
</footer>

<script src="assets/js/main.js"></script>
<script src="assets/js/contact.js"></script>
<script src="assets/js/cart.js"></script>
<script src="assets/js/poll.js"></script>
<script src="assets/js/registration.js"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" i
ntegrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymo
us"></script>

</body>

</html>
```

3.2.2 Pages

3.2.2.1 Admin

3.2.2.1.1 AdminInsertProduct.php

```
<div class="container">
    <div class="row">
        <div class="col-6 mx-auto">
            <h2 class="text-center mt-4">Insert Product</h2>
            <hr class="underTitle mb-4" />

            <form action="models/products/insert.php" method="POST" enctype="multipart/form-data">
                <div class="form-group">
                    <input type="text" name="articleName" placeholder="Enter article name" class="form-control" />
                </div>
                <div class="form-group">
                    <textarea name="taDescription" cols="30" rows="10" class="form-control" placeholder="Description"></textarea>
                </div>
                <div class="form-group">
                    <input type="number" name="price" class="form-control" placeholder="Enter price" />
                </div>
                <div class="form-group">
                    <input type="file" name="imageFile" class="form-control-file" />
                </div>
                <div class="form-group">
                    <input type="text" name="alt" placeholder="Enter alt attribute" class="form-control" />
                </div>
                <div class="form-group">
                    <select name="ddlCategory" class="form-control">
                        <option value="0">Choose category</option>
                        <?php
                        $categories = getCategories();
                        foreach ($categories as $cat) :
                        ?>
                            <option value="<?= $cat->idCategory ?>"><?= $cat->name ?></option>
                        <?php endforeach; ?>
                    </select>
                </div>

                <div class="form-group">
                    <input type="submit" value="Upload" name="btnUpload" class="btn btn-success">
                </div>
            </form>
            <?php
            if (isset($_SESSION['msg'])) {
                var_dump($_SESSION['msg']);
                unset($_SESSION['msg']);
            }
            ?>
        </div>
    </div>
</div>
```

3.2.2.1.2 adminPoll.php

```
<div class="row">
    <div class="col-md-6 mx-auto">
        <h2 class="text-center mt-4">Manage Polls</h2>
        <hr class="underTitle mb-4" />
        <form method="POST" action="">
            <h3 class="mt-4">Insert Poll</h3>
            <div class="form-group">
                <input type="text" name="tbPollName" id="tbPollName" placeholder="Question" class="form-control">
            </div>
            <div class="form-group">
                <textarea id="taAnswers" rows="6" class="form-control" placeholder="Separate the answers with SEMICOLON"></textarea>
            </div>
            <input type="button" value="Insert" name="btnAddPoll" id="btnAddPoll" class="btn btn-success">
            <div id="errorsPoll">

            </div>
            <hr class="underTitle mb-4" />

            <h3>Activate Poll</h3>
            <div class="form-group">
                <select name="ddlActivatePoll" id="ddlActivatePoll" class="form-control">

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

            </div>
            <hr class="underTitle mb-4" />

            <h3>Poll Results</h3>
            <div class="form-group">
                <select name="ddlPollResult" id="ddlPollResult" class="form-control">

                </select>
            </div>
            <input type="button" value="Show" name="btnPollResult" id="btnPollResult" class="btn btn-success">
            <div id="pollResultAdmin">

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

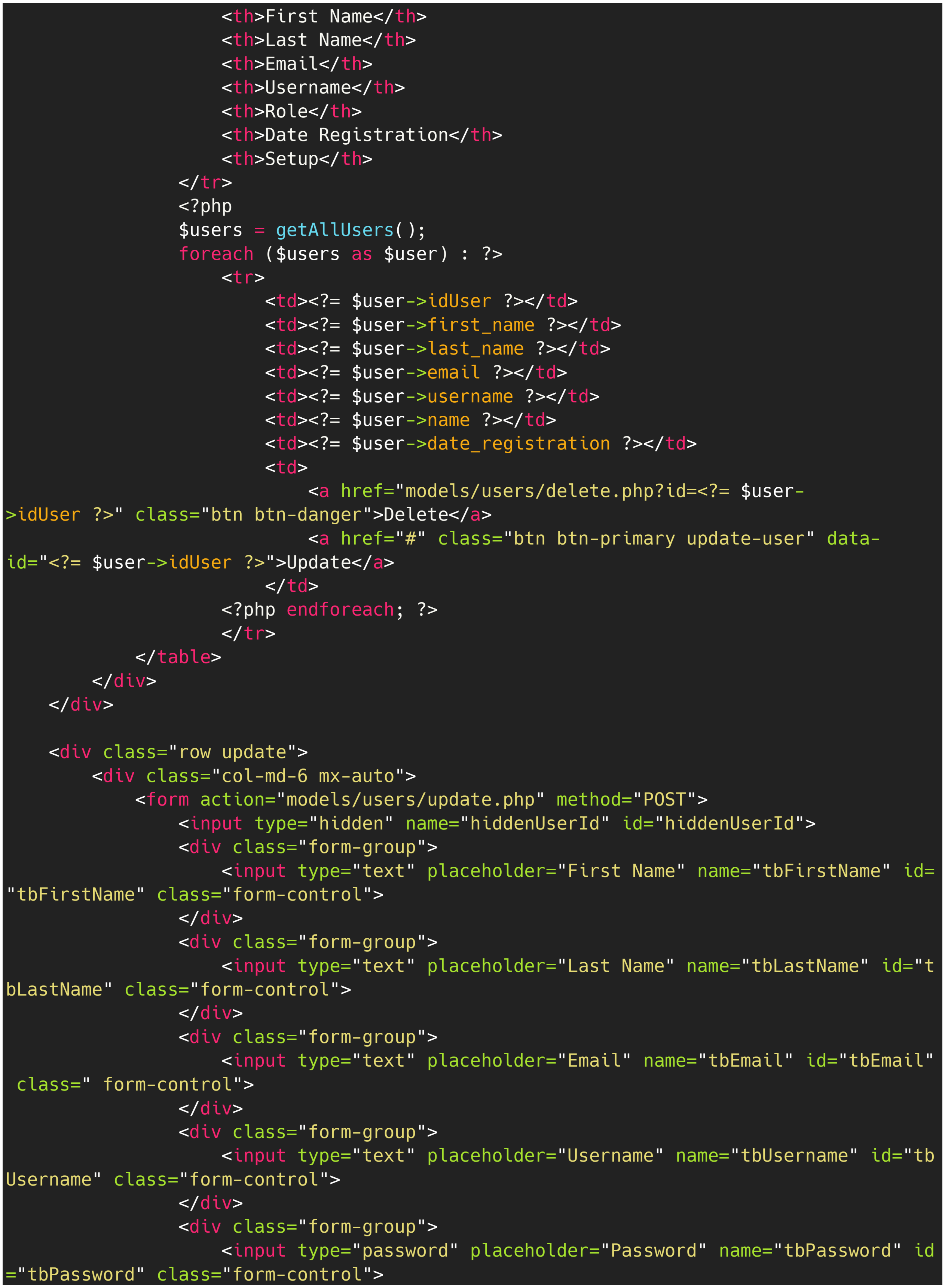
3.2.2.1.3 adminProducts.php

```
<div class="container">
    <div class="row">
        <div class="col-lg-12">
            <h2 class="text-center mt-4">All Products</h2>
            <hr class="underTitle mb-4" />
            <table class="table table-stripped table-bordered table-hover">
                <tr>
                    <th>ID</th>
                    <th>Name</th>
                    <th>Price</th>
                    <th>Image</th>
                    <th>Category</th>
                    <th>Setup</th>
                </tr>
                <?php
                $articles = getAllArticles();
                foreach ($articles as $article) :
                ?>
                    <tr>
                        <td><?= $article->idArticle ?></td>
                        <td><?= $article->name ?></td>
                        <td>$<?= $article->price ?></td>
                        <td align="center"><img src="<?= $article->image ?>" alt="<?= $article->alt ?>" width="80" height="80"></td>
                        <td><?= $article->categoryName ?></td>
                        <td>
                            <a href="models/products/delete.php?id=<?=$article->idArticle?>" class="btn btn-danger">Delete</a>
                            <a href="#" class="btn btn-primary update-product" data-id="<?=$article->idArticle?>">Update</a>
                        </td>
                    </tr>
                <?php endforeach; ?>
            </table>
        </div>
    </div>

    <div class="row update">
        <div class="col-md-6 mx-auto">
            <form action="models/products/update.php" method="POST" enctype="multipart/form-data">
                <input type="hidden" name="hiddenProductID" id="hiddenProductID">
                <div class="form-group">
                    <input type="text" name="productName" id="productName" class="form-control">
                </div>
                <div class="form-group">
```

```
                    <textarea name="taDescription" id="taDescription" cols="30" rows="10" class="form-control"></textarea>
                </div>
                <div class="form-group">
                    <input type="text" name="productPrice" id="productPrice" class="form-control">
                </div>
                <img src="" alt="" style="width:20%;" id="emptyImage" name="emptyImage">
                <div class="form-group">
                    <input type="file" name="productImage" id="productImage">
                </div>

                <div class="form-group">
                    <input type="text" name="productAlt" id="productAlt" class="form-control">
                </div>
                <div class="form-group">
                    <select name="ddlCategory" id="ddlCategory" class="form-control">
                        <option value="0">Choose category</option>
                        <?php
                        $categories = getCategories();
                        foreach ($categories as $cat) : ?>
                            <option value="<?= $cat->idCategory ?>"><?= $cat->name ?></option>
                        <?php endforeach; ?>
                    </select>
                </div>
                <input type="submit" value="Update" class="btn btn-success" name="btnUpdateProduct">
            </form>
        </div>
    </div>

    <div class="update-response">
        <?php
            if (isset($_SESSION['errorUpdate'])) {
                echo $_SESSION['errorUpdate'];
                unset($_SESSION['errorUpdate']);
            }
        ?>
    </div>
</div>
```

3.2.2.1.4 adminUsers.php

```
<div class="container">
    <div class="row">
        <div class="col-lg-12">
            <h2 class="text-center mt-4">Users</h2>
            <hr class="underTitle mb-4" />
            <table class="table table-stripped table-bordered table-hover">
                <tr>
                    <th>ID</th>
```



```
                    <th>First Name</th>
                    <th>Last Name</th>
                    <th>Email</th>
                    <th>Username</th>
                    <th>Role</th>
                    <th>Date Registration</th>
                    <th>Setup</th>
                </tr>
                <?php
                $users = getAllUsers();
                foreach ($users as $user) : ?>
                    <tr>
                        <td><?= $user->idUser ?></td>
                        <td><?= $user->first_name ?></td>
                        <td><?= $user->last_name ?></td>
                        <td><?= $user->email ?></td>
                        <td><?= $user->username ?></td>
                        <td><?= $user->name ?></td>
                        <td><?= $user->date_registration ?></td>
                        <td>
                            <a href="models/users/delete.php?id=<?= $user->idUser ?>" class="btn btn-danger">Delete</a>
                            <a href="#" class="btn btn-primary update-user" data-id="<?= $user->idUser ?>">Update</a>
                        </td>
                    <?php endforeach; ?>
                    </tr>
            </table>
        </div>
    </div>

    <div class="row update">
        <div class="col-md-6 mx-auto">
            <form action="models/users/update.php" method="POST">
                <input type="hidden" name="hiddenUserId" id="hiddenUserId">
                <div class="form-group">
                    <input type="text" placeholder="First Name" name="tbFirstName" id="tbFirstName" class="form-control">
                </div>
                <div class="form-group">
                    <input type="text" placeholder="Last Name" name="tbLastName" id="tbLastName" class="form-control">
                </div>
                <div class="form-group">
                    <input type="text" placeholder="Email" name="tbEmail" id="tbEmail" class=" form-control">
                </div>
                <div class="form-group">
                    <input type="text" placeholder="Username" name="tbUsername" id="tbUsername" class="form-control">
                </div>
                <div class="form-group">
                    <input type="password" placeholder="Password" name="tbPassword" id="tbPassword" class="form-control">
                </div>
                <div class="form-group">
                    <select name="ddlRole" id="ddlRole" class="form-control">
                        <option value="0">choose role</option>
                        <?php
                        $roles = getAllRoles();
                        foreach ($roles as $role) : ?>
                            <option value="<?= $role->idRole ?>"><?= $role-
>name ?></option>
                        <?php endforeach; ?>
                    </select>
                </div>
                <div class="form-group">
                    <input type="date" name="dateRegistration" id="dateRegistration" c
lass="form-control">
                </div>
                <div class="form-group">
                    <label for="chbActive">Active user? Yes / No</label>
                    <input type="checkbox" name="chbActive" id="chbActive" value="1">
                </div>
                <input type="submit" value="Update" class="btn btn-
success" name="btnUpdateUser">
            </form>
        </div>
    </div>

    <div class="updateResponse">
        <?php
        if (isset($_SESSION['message'])) :
            var_dump($_SESSION['message']);
            unset($_SESSION['message']);
        endif;
        ?>
    </div>
</div>
```

3.2.2.2 articleDetails.php

```
<?php
session_start();

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

    $id = $_POST['idArticle'];

    $query = "SELECT * FROM article WHERE idArticle = $id";
    $result = $conn->query($query);

    $output = '';
    $output .= '
        <div class="row">
            <div class="col text-center">
    ';
    if (isset($_SESSION['user']) && $_SESSION['user']->roleName == 'user') {
        foreach ($result as $row) {
            $output .= '
            <div class="shop-item">
                <img src="' . $row->image . '" class="img-fluid mx-
auto" alt="' . $row->alt . '" />
                <h2 class="my-2">' . $row->name . '</h2>
                <h2>$' . $row->price . '</h2>
                <p class="lead text-muted">
                    ' . $row->description . '
                </p>
                <div class="d-flex flex-wrap">
                    <button class="btn btn-success my-2 mx-2 add-to-cart" data-
id="'.$row->idArticle.'">Add to cart</button>
                </div>
            </div>
        ';
        }
    } else {
        foreach ($result as $row) {
            $output .= '
            <img src="' . $row->image . '" class="img-fluid mx-auto" alt="' . $row-
>alt . '" />
            <h2 class="my-2">' . $row->name . '</h2>
            <h2>$' . $row->price . '</h2>
            <p class="lead text-muted">
                ' . $row->description . '
            </p>
            <div class="d-flex flex-wrap align-items-center">
                <p class="text-warning ml-3 mt-2 font-weight-
bold">If you want to buy, you must login first.</p>
            </div>
        ';
        }
    }


    $output .= "</div></div>";
    echo $output;
}
```

3.2.2.3 cart.php

```
<div class="container">
    <div class="row">
        <div class="col-lg-12 cart-center">
            <h2 class="text-center mt-4">Shopping Cart</h2>
            <hr class="underTitle mb-4" />

            <div id="cart" class="container">
            </div>

            <button type='button' class='btn btn-success' id='button-
buy'>Make a purchase</button>
        </div>
    </div>
</div>
```

3.2.2.4 contact.php

```
<div class="container">
    <div class="row">
        <div class="col-xs-8 col-sm-10 col-lg-12 mx-auto">
            <h2 class="text-center mt-4">Contact Us Here</h2>
            <hr class="underTitle mb-4" />

            <form action="models/formToEmail.php" method="POST" onSubmit="return check
FormClientSide();" >
                <div class="form-group">
                    <label for="firstNameContact">Your First Name</label>
                    <input type="text" class="form-
control" name="firstNameContact" id="firstNameContact">
                </div>
                <div class="form-group">
                    <label for="formEmail">Your Email</label>
                    <input type="text" class="form-
control" name="formEmail" id="formEmail">
                </div>
                <div class="form-group">
                    <label for="contentContact">Message</label>
                    <textarea class="form-
control" name="contentContact" id="contentContact" rows="8"></textarea>
                </div>
                <div class="d-flex justify-content-center">
                    <input type="submit" id="btnSendMessage" name="btnSendMessage" cla
ss="btn btn-success" value="Send">
                </div>

                <div id="notification">
                    <?php
                    if (isset($_SESSION['contactErrors'])) {
                        $errors = $_SESSION['contactErrors'];
                        foreach ($errors as $one) {
                            echo $one . "<br/>";
                        }
                        unset($_SESSION['contactErrors']);
                    }
                    ?>
                </div>
            </form>
        </div>
    </div>
</div>
```

3.2.2.5 home.php

```
<main>
    <!-- Start of slider -->
    <div id="carouselFeatured" class="carousel slide" data-ride="carousel">
        <div class="carousel-inner" role="listbox">
            <div class="carousel-item active">
            </div>

            <div class="carousel-item">
                <div class="carousel-caption d-none d-md-block">
                </div>
            </div>

            <div class="carousel-item">
                <div class="carousel-caption d-none d-md-block">
                </div>
            </div>
        </div>
        <a class="carousel-control-prev" href="#carouselFeatured" role="button" data-slide="prev">
            <span><i class="fa fa-angle-left" aria-hidden="true"></i></span>
            <span class="sr-only">Previous</span>
        </a>
        <a class="carousel-control-next" href="#carouselFeatured" role="button" data-slide="next">
            <span><i class="fa fa-angle-right" aria-hidden="true"></i></span>
            <span class="sr-only">Next</span>
        </a>
    </div>
    <!-- End of slider -->

    <!-- Start of details about company -->
    <div id="info">
        <div class="info-box">
            <i class="fas fa-truck fa-2x"></i>
            <div class="info-box-txt">
                <h4>Free Shipping</h4>
                <p>On the territory of Belgrade <br /> for orders over 20$</p>
            </div>
        </div>
        <div class="info-box">
            <i class="far fa-credit-card fa-2x"></i>
            <div class="info-box-txt">
                <h4>More Payment Methods</h4>
                <p>Payment cards, cash on delivery, <br /> through the account</p>
            </div>
        </div>
        <div class="info-box">
            <i class="fas fa-gift fa-2x"></i>
            <div class="info-box-txt">
                <h4>We Know What You Need</h4>
                <p>Over 20.000 items on offer <br /> and win up to 50% discount </p>
            </div>
        </div>
        <div class="info-box">
            <i class="fas fa-phone fa-2x"></i>
            <div class="info-box-txt">
                <h4>Call Our Operator</h4>
                <p>Mobile: 069/312-354-27 <br /> Service: 011/1111-347 </p>
            </div>
        </div>
    </div>
    <!-- End of details about company -->

    <!-- Start of separated products from database -->
    <div id="separatedProducts">
        <div class="container-fluid">
            <h2 class="text-center">Featured Products</h2>
            <hr class="underTitle mb-4" />
            <div class="col-10 col-md-7 col-lg-9 mx-auto">
                <div class="row">
                    <?php
                    $articlesFeatured = getAllArticlesFeatured();
                    foreach ($articlesFeatured as $aF) :
                    ?>
                        <div class="mx-auto col-md-6 col-lg-3 my-3">
                            <div class="featured-container p-5">
                                <img src="<?= $aF->image ?>" alt="<?= $aF-
>alt ?>" class="img-fluid" />

                                <button name="view" value="View" id=<?= $aF-
>idArticle ?> class="view_data">
                                    <i class="fa fa-search"></i>
                                </button>

                            </div>
                            <h6 class="text-center my-2"><?= $aF->name ?></h6>
                            <h6 class="text-center">
                                <span>$<?= $aF->price ?></span>
                            </h6>
                        </div>
                    <?php endforeach; ?>
                </div>
            </div>
        </div>
    </div>
    <!-- End of separated products from database -->

</main>
```

3.2.2.6 poll.php

```
<div class="container">
    <div class="row align-item-center">
```

```
        <div class="col-md-6 mx-auto" id="poll">
            <h2 class="text-center mt-4">Poll</h2>
            <hr class="underTitle mb-4" />
            <?php
            $poll = getActivePoll();
            ?>

            <h3>
                <?php
                if ($poll) {
                    echo $poll->question;
                } else {
                    echo "There is no active poll at the moment.";
                }
                ?>
            </h3>

            <form action="" method="POST" class="form-horizontal" id="formVoting">
                <?php
                $queryAnswers = getActivePollAnswers();
                foreach ($queryAnswers as $answer) :
                ?>
                    <div class="radioButtons">
                        <label>
                            <input type="radio" name="rbPoll" class="rbPoll" value="<?= $answer->idAnswer ?>"> <?= $answer->answer ?>
                        </label>
                    </div>
                <?php endforeach; ?>

                <input type="hidden" id="hiddenUserPollField" value="<?= $_SESSION['user']->idUser ?>">
                <input type="hidden" id="activePoll" value="<?= $poll->idPoll ?>">
            </form>

            <div class="buttons">
                <button class="btn btn-success" id="btnVote" type="button">Vote</button>
                <button class="btn btn-success" id="btnVotingResult">Results</button>
            </div>

            <div id="votingResults">

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

3.2.2.7 *products.php*

```
<div class="container">
    <div class="row">
        <div class="col-xs-8 col-sm-10 col-lg-12 mx-auto">
```

```
            <h2 class="text-center mt-4">Our Products</h2>
            <hr class="underTitle mb-4" />
            <div class="category-list">
                <select name="ddlCategory" id="ddlCategory" class="custom-select my-4">
                    <option value="0">Choose category</option>
                    <?php
                    $categories = getCategories();
                    foreach ($categories as $cat) :
                    ?>
                        <option value="<?= $cat->idCategory ?>"><?= $cat->name ?></option>
                    <?php endforeach; ?>
                </select>
            </div>

            <div id="allArticles" class="filter_data row mt-4">

            </div>

            <div id="pagination">
                <ul id="pagination-list">

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

3.3 Models

3.3.1 Polls

3.3.1.1 activatePoll.php

```
<?php

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

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

    $id = $_POST['id'];

    $code = 404;
    $data = null;

    try {
        $conn->beginTransaction();

        $deactive = $conn->prepare("UPDATE poll SET active = 0 WHERE idPoll <> ?");
        $deactive->execute([$id]);
        $activate = $conn->prepare("UPDATE poll SET active = 1 WHERE idPoll = ?");
        $activate->execute([$id]);

        $conn->commit();

        $code = 201;
        $data = ["message" => "Successfully activated poll."];
    } catch(PDOException $ex) {
        $conn->rollBack();
        $code = 500;
    }
}

echo json_encode($data);
http_response_code($code);
```

3.3.1.2 allPolls.php

```
<?php

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

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

$data = null;
$code = 404;

try {
    $data = getAllPolls();
    $code = 200;
} catch(PDOException $ex) {
    $code = 500;
    echo $ex->getMessage();
}

echo json_encode($data);
http_response_code($code);
```

3.3.1.3 functions.php

```
<?php

function getAllPolls()
{
    return executeQuery("SELECT * FROM poll");
}

function getActivePoll()
{
    global $conn;
    return $conn->query("SELECT * FROM poll WHERE active = 1")->fetch();
}
```

```php
function getActivePollAnswers()
{
    return executeQuery("SELECT * FROM poll p INNER JOIN answer a ON p.idPoll = a.idPoll WHERE p.active = 1");
}

function ifVoted($id)
{
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM poll p INNER JOIN answer a ON p.idPoll = a.idPoll INNER JOIN voting v ON a.idAnswer = v.idAnswer WHERE p.active = 1 AND v.idUser = ?");
    $stmt->execute([$id]);
    return $stmt;
}

function voting($idAnswer, $idUser)
{
    global $conn;
    $vote = $conn->prepare("INSERT INTO voting(idAnswer, idUser) VALUES(?,?)");
    return $vote->execute([$idAnswer, $idUser]);
}

function getResultOfSinglePoll($id) {
    global $conn;
    $stmt = $conn->prepare("SELECT a.answer, COUNT(v.idAnswer) AS num FROM answer a LEFT OUTER JOIN voting v ON a.idAnswer = v.idAnswer WHERE a.idPoll = ? GROUP BY a.answer");
    $stmt->execute([$id]);
    return $stmt->fetchAll();
}
```

3.3.1.4 insertPoll.php

```php
<?php

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

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

    $name = $_POST['name'];
    $answers = $_POST['answers'];

    $regOverall = "/^([\wŠĐŽĆČčćžšđ\d\.\?])+(\s[\wŠĐŽĆČčćžšđ\d\.\?]+)*$/";

    $code = 404;
    $data = null;
    $errors = [];

    if (!preg_match($regOverall, $name)) {
        $errors[] = "Question is not in good format.";
    }

    if (count($answers) < 1) {
        $errors[] = "Answer is not in good format.";
    } else {
        for ($i = 0; $i < count($answers); $i++) {
            if (!preg_match($regOverall, trim($answers[$i]))) {
                $answers[] = ($i + 1) . " answer is not in good format";
            }
        }
    }

    if (count($errors)) {
        $print = "";
        foreach ($errors as $error) {
            $print = $error . "<br/>";
        }
        $code = 422;
        $data = ["message" => $print];
    } else {
        try {
            $conn->beginTransaction();
            $poll = $conn-
>prepare("INSERT INTO poll(idPoll, question, active) VALUES (NULL, :name, 0)");
            $poll->bindParam(":name", $name);
            $poll->execute();

            $id = $conn->lastInsertId();
            $parameters = [];
            $values = [];

            foreach ($answers as $ans) {
                $parameters[] = "(?,?)";
                $values[] = $id;
                $values[] = $ans;
            }

            $query = $conn-
>prepare("INSERT INTO answer (idPoll, answer) VALUES" . implode(",", $parameters));
            $query->execute($values);

            $conn->commit();
            $data = ["message" => "Poll is successfully added."];
            $code = 201;
        } catch (PDOException $ex) {
            $conn->rollBack();
            $code = 500;
        }
    }
}

echo json_encode($data);
http_response_code($code);
```

3.3.1.5 showPoll.php

```php
<?php

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

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

    $id = $_POST['id'];

    $code = 404;
    $data = null;

    try {
        $data = getResultOfSinglePoll($id);
        $code = 200;
    } catch(PDOException $ex) {
        $code = 500;
        echo $ex->getMessage();
    }
}

echo json_encode($data);
http_response_code($code);
```

3.3.1.6 voting.php

```php
<?php

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

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

    $id = $_POST['id'];
    $answer = $_POST['answer'];

    $data = null;
    $code = 404;

    try {
        $ifVoted = ifVoted($id);
        if ($ifVoted->rowCount() == 0) {
            if(voting($answer, $id)) {
                $data = ["message" => "Successfully voted."];
                $code = 201;
            } else {
                $code = 500;
                $data = ["message" => "Error occured while voting."];
            }
        } else {
            $code = 409;
            $data = ["message" => "Already voted."];
        }
    } catch(PDOException $ex) {
        $code = 409;
    }
}

echo json_encode($data);
http_response_code($code);
```

3.3.2 Products

3.3.2.1 ajaxAllArticles.php

```
<?php

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

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

$code = 404;
$data = null;

$articles = getAllArticles();

if ($articles) {
    $data = $articles;
    $code = 200;
} else {
    $code = 500;
}

http_response_code($code);
echo json_encode($data);
```

3.3.2.2 ajaxArticles.php

```
<?php

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

$data = null;
$code = 404;

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

    try {
        $data = getAllArticlesWithPagination();
        $code = 200;
    } catch (PDOException $ex) {
        $code = 500;
        echo $ex->getMessage();
    }

}

echo json_encode($data);
http_response_code($code);
```

3.3.2.3 ajaxGetArticle.php

```
<?php

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

$data = null;
$code = 404;

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

    $id = $_POST['id'];

    try {
        $data = getOneArticle($id);
        $code = 200;
    } catch(PDOException $ex) {
        $code = 500;
        echo $ex->getMessage();
    }
}

echo json_encode($data);
```

3.3.2.4 delete.php

```
<?php

if (isset($_GET['id'])) {
    require_once "../../config/connection.php";
    $id = $_GET['id'];

    $prepare = $conn->prepare("DELETE FROM article WHERE idArticle = :id");
    $prepare->bindParam(":id", $id);
    $prepare->execute();

    header("Location: ../../index.php?page=adminProducts");
}
```

3.3.2.5 filter.php

```php
<?php

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

$data = null;
$code = 404;

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

    $page = ($_POST['idPagination'] - 1) * 4;
    $idCategory = $_POST['idCategory'];

    $query = "SELECT * FROM article";

    if ($idCategory != '0') {
        $query .= " WHERE idCategory=:idCategory";
    }

    $query .= " LIMIT 4 OFFSET $page";
    $stmt = $conn->prepare($query);
    $stmt->bindParam(":idCategory", $idCategory);

    try {
        if ($stmt->execute()) {
            $data = $stmt->fetchAll();
            $code = 200;
        } else {
            $code = 500;
        }
    } catch (PDOException $ex) {
        echo $ex->getMessage();
        $code = 500;
    }
}

echo json_encode($data);
http_response_code($code);
```

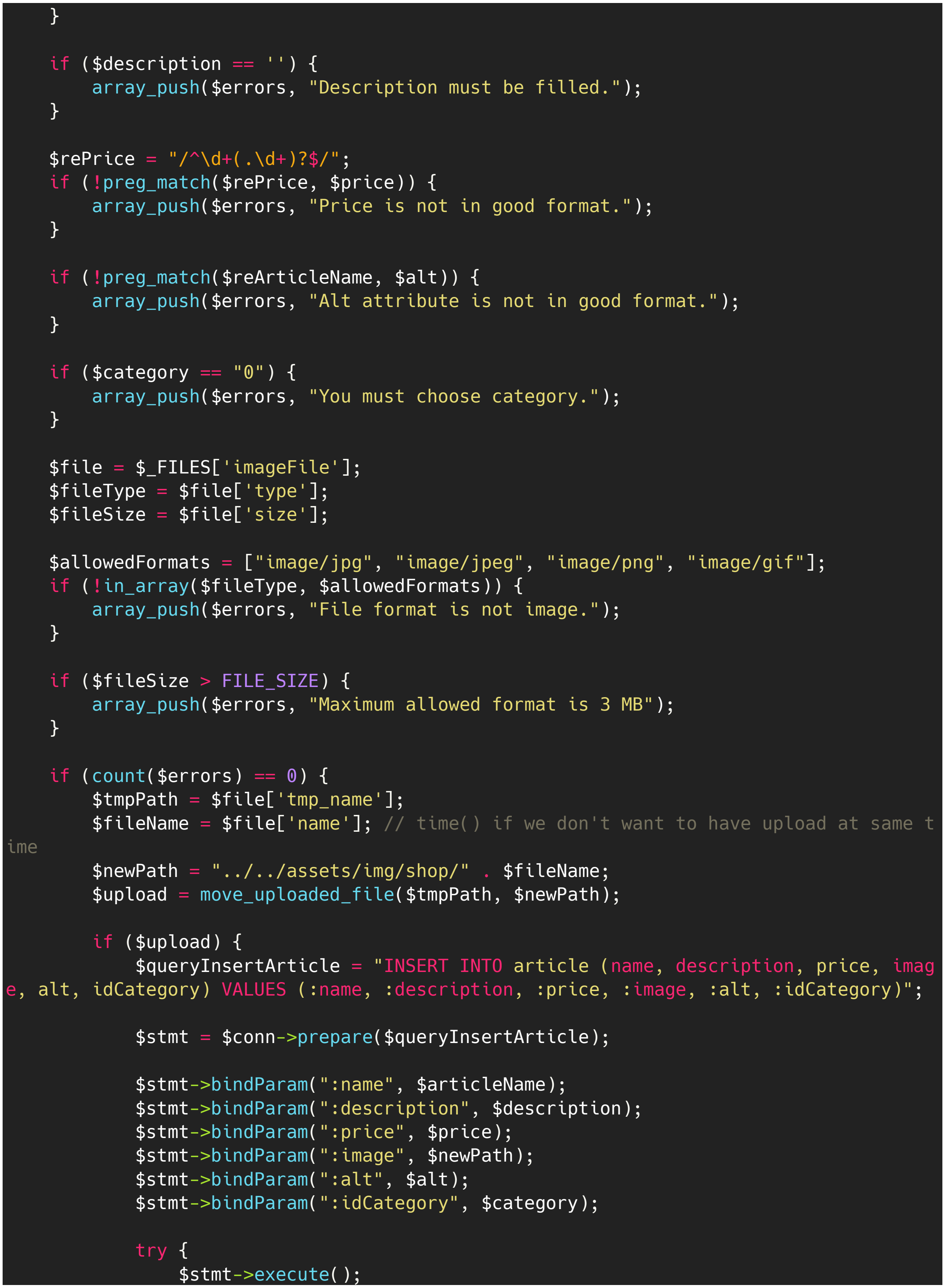
3.3.2.6 functions.php

```php
<?php

function getAllArticlesFeatured() {
    return executeQuery("SELECT * FROM article ORDER BY price DESC LIMIT 0,4");
}

function getAllArticlesWithPagination() {
    $page = ($_POST['id'] - 1) * 4;
    return executeQuery("SELECT * FROM article LIMIT $page, 4");
}
```

```
function getNumberOfArticles() {
    global $conn;
    return $conn->query("SELECT COUNT(*) AS numOfArticles FROM article")->fetch();
}

function getArticlesPerCategory($id) {
    global $conn;
    $stmt = $conn-
>prepare("SELECT COUNT(*) AS numOfArticles FROM article WHERE idCategory = ?");
    $stmt->execute([$id]);
    return $stmt->fetch();
}

function getArticlesBySearchValue($text) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM article WHERE name LIKE LOWER(?)");
    $stmt->execute([$text]);
    return $stmt->fetchAll();
}

function getAllArticles() {
    return executeQuery("SELECT a.*, c.name AS categoryName FROM article a INNER JOIN
category c ON a.idCategory = c.idCategory");
}

function getOneArticle($id) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM article WHERE idArticle = :id");
    $stmt->bindParam(":id", $id);
    $stmt->execute();
    return $stmt->fetch();
}
```

3.3.2.7 insert.php

```
<?php
session_start();
define("FILE_SIZE", 3000000);

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

    $errors = [];

    $articleName = $_POST['articleName'];
    $description = $_POST['taDescription'];
    $price = $_POST['price'];
    $alt = $_POST['alt'];
    $category = $_POST['ddlCategory'];

    $reArticleName = "/^[\w\s\-\_\.]{1,255}$/";
    if (!preg_match($reArticleName, $articleName)) {
        array_push($errors, "Article name is not in good format.");
```



```php
    }

    if ($description == '') {
        array_push($errors, "Description must be filled.");
    }

    $rePrice = "/^\d+(.\d+)?$/";
    if (!preg_match($rePrice, $price)) {
        array_push($errors, "Price is not in good format.");
    }

    if (!preg_match($reArticleName, $alt)) {
        array_push($errors, "Alt attribute is not in good format.");
    }

    if ($category == "0") {
        array_push($errors, "You must choose category.");
    }

    $file = $_FILES['imageFile'];
    $fileType = $file['type'];
    $fileSize = $file['size'];

    $allowedFormats = ["image/jpg", "image/jpeg", "image/png", "image/gif"];
    if (!in_array($fileType, $allowedFormats)) {
        array_push($errors, "File format is not image.");
    }

    if ($fileSize > FILE_SIZE) {
        array_push($errors, "Maximum allowed format is 3 MB");
    }

    if (count($errors) == 0) {
        $tmpPath = $file['tmp_name'];
        $fileName = $file['name']; // time() if we don't want to have upload at same time
        $newPath = "../../assets/img/shop/" . $fileName;
        $upload = move_uploaded_file($tmpPath, $newPath);

        if ($upload) {
            $queryInsertArticle = "INSERT INTO article (name, description, price, image, alt, idCategory) VALUES (:name, :description, :price, :image, :alt, :idCategory)";

            $stmt = $conn->prepare($queryInsertArticle);

            $stmt->bindParam(":name", $articleName);
            $stmt->bindParam(":description", $description);
            $stmt->bindParam(":price", $price);
            $stmt->bindParam(":image", $newPath);
            $stmt->bindParam(":alt", $alt);
            $stmt->bindParam(":idCategory", $category);

            try {
                $stmt->execute();
            } catch (PDOException $ex) {
                echo $ex->getMessage();
            }
            if ($stmt->rowCount()) {
                $_SESSION['msg'] = "Successfull upload.";
                header("Location: ../../index.php?page=adminInsertProduct");
            }
        } else {
            $_SESSION['msg'] = "Upload can't be done. Check database.";
            header("Location: ../../index.php?page=adminInsertProduct");
        }
    } else {
        $_SESSION['msg'] = $errors;
        header("Location: ../../index.php?page=adminInsertProduct");
    }
}
```

3.3.2.8 pagination.php

```
<?php

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

$data = null;
$code = 404;

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

    try {
        $id = $_POST['id'];
        if ($id != 0) {
            $data = getArticlesPerCategory($id);
        } else {
            $data = getNumberOfArticles();
        }
        $code = 200;
    } catch(PDOException $ex) {
        $code = 500;
        echo $ex->getMessage();
    }
}

echo json_encode($data);
http_response_code($code);
```

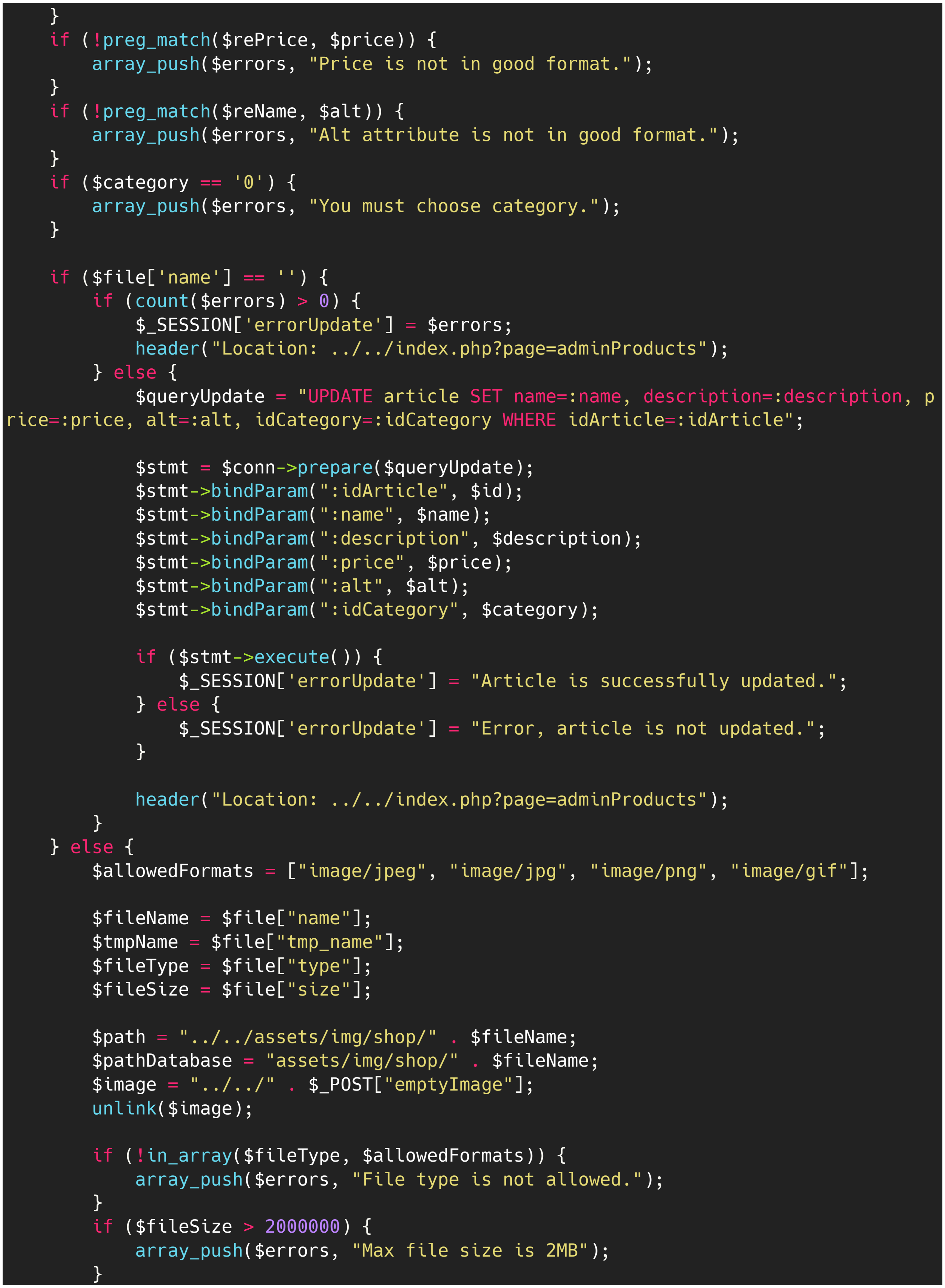
3.3.2.9 search.php

```
<?php

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

```php
$code = 404;
$data = null;

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

    $text = trim($_POST['searchValue']);
    $text = "%$text%";

    try {
        $data = getArticlesBySearchValue($text);
        $code = 200;
    } catch(PDOException $ex) {
        $code = 500;
        echo $ex->getMessage();
    }
}

echo json_encode($data);
http_response_code($code);
```

3.3.2.10 update.php

```php
<?php

session_start();

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

    $id = $_POST['hiddenProductID'];
    $name = $_POST['productName'];
    $description = $_POST['taDescription'];
    $price = $_POST['productPrice'];
    $file = $_FILES['productImage'];
    $alt = $_POST['productAlt'];
    $category = $_POST['ddlCategory'];

    $errors = [];

    $reName = "/^[A-z\d\-\_]+(\s[A-z\d\-\_]+)*$/";
    $reDescription = "/^[\w\d\s\-\.\,]{1,255}$/";
    $rePrice = "/^(\d+)$/";

    if (!preg_match($reName, $name)) {
        array_push($errors, "Name of article is not in good format.");
    }
    if (!preg_match($reDescription, $description)) {
        array_push($errors, "Description only contains letters, numbers and whitespace
");
```



```
    }
    if (!preg_match($rePrice, $price)) {
        array_push($errors, "Price is not in good format.");
    }
    if (!preg_match($reName, $alt)) {
        array_push($errors, "Alt attribute is not in good format.");
    }
    if ($category == '0') {
        array_push($errors, "You must choose category.");
    }

    if ($file['name'] == '') {
        if (count($errors) > 0) {
            $_SESSION['errorUpdate'] = $errors;
            header("Location: ../../index.php?page=adminProducts");
        } else {
            $queryUpdate = "UPDATE article SET name=:name, description=:description, p
rice=:price, alt=:alt, idCategory=:idCategory WHERE idArticle=:idArticle";

            $stmt = $conn->prepare($queryUpdate);
            $stmt->bindParam(":idArticle", $id);
            $stmt->bindParam(":name", $name);
            $stmt->bindParam(":description", $description);
            $stmt->bindParam(":price", $price);
            $stmt->bindParam(":alt", $alt);
            $stmt->bindParam(":idCategory", $category);

            if ($stmt->execute()) {
                $_SESSION['errorUpdate'] = "Article is successfully updated.";
            } else {
                $_SESSION['errorUpdate'] = "Error, article is not updated.";
            }

            header("Location: ../../index.php?page=adminProducts");
        }
    } else {
        $allowedFormats = ["image/jpeg", "image/jpg", "image/png", "image/gif"];

        $fileName = $file["name"];
        $tmpName = $file["tmp_name"];
        $fileType = $file["type"];
        $fileSize = $file["size"];

        $path = "../../assets/img/shop/" . $fileName;
        $pathDatabase = "assets/img/shop/" . $fileName;
        $image = "../../" . $_POST["emptyImage"];
        unlink($image);

        if (!in_array($fileType, $allowedFormats)) {
            array_push($errors, "File type is not allowed.");
        }
        if ($fileSize > 2000000) {
            array_push($errors, "Max file size is 2MB");
        }
        if (count($errors) > 0) {
            $_SESSION['errorUpdate'] = $errors;
            header("Location: ../../index.php?page=adminProducts");
        } elseif(move_uploaded_file($tmpName, $path)) {
            $queryUpdate = "UPDATE article SET name=:name, description=:description, p
rice=:price, image=:image, alt=:alt, idCategory=:idCategory WHERE idArticle=:idArticle
";

            $stmt = $conn->prepare($queryUpdate);
            $stmt->bindParam(":idArticle", $id);
            $stmt->bindParam(":name", $name);
            $stmt->bindParam(":description", $description);
            $stmt->bindParam(":price", $price);
            $stmt->bindParam(":image", $pathDatabase);
            $stmt->bindParam(":alt", $alt);
            $stmt->bindParam(":idCategory", $category);

            if ($stmt->execute()) {
                $_SESSION['errorUpdate'] = "Article is successfully updated.";
            } else {
                $_SESSION['errorUpdate'] = "Error, article is not updated.";
            }

            header("Location: ../../index.php?page=adminProducts");
        }
    }
}
```

3.3.3 Users

3.3.3.1 ajaxGetUser.php

```
<?php

if (isset($_POST['id'])) {
    header("Content-Type: application/json");
    require_once "../../config/connection.php";

    $id = $_POST['id'];

    $code = 404;
    $data = null;

    $queryGetUser = "SELECT * FROM user u INNER JOIN role r ON u.idRole = r.idRole WHE
RE u.idUser = :id";
    $prepare = $conn->prepare($queryGetUser);
    $prepare->bindParam(":id", $id);

    try {
        $prepare->execute();
        $user = $prepare->fetch();
        if ($user) {
            $data = $user;
            $code = 201;
        } else {
            $code = 500;
        }
    } catch(PDOException $ex) {
        $code = 500;
        $data = $ex->getMessage();
    }
}
http_response_code($code);
echo json_encode($data);
```

3.3.3.2 delete.php

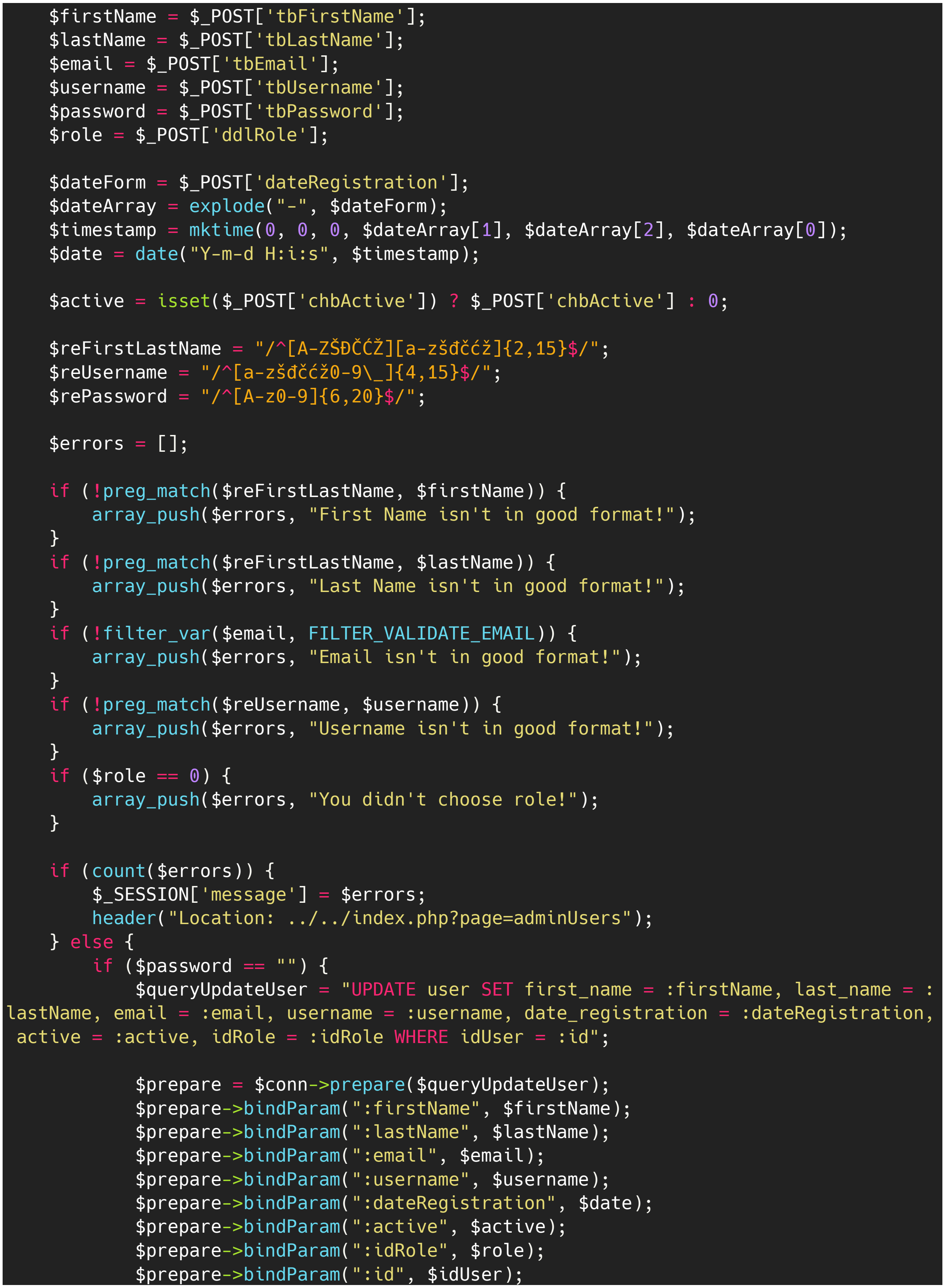
```
<?php

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

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

    $prepare = $conn->prepare("DELETE FROM user WHERE idUser = :id");

    $prepare->bindParam(":id", $id);
    $prepare->execute();

    header("Location: ../../index.php?page=adminUsers");
}
```

3.3.3.3 functions.php

```
<?php

function getAllUsers() {
    return executeQuery("SELECT * FROM user u INNER JOIN role r ON u.idRole = r.idRole");
}

function getAllRoles() {
    return executeQuery("SELECT * FROM role");
}
```

3.3.3.4 update.php

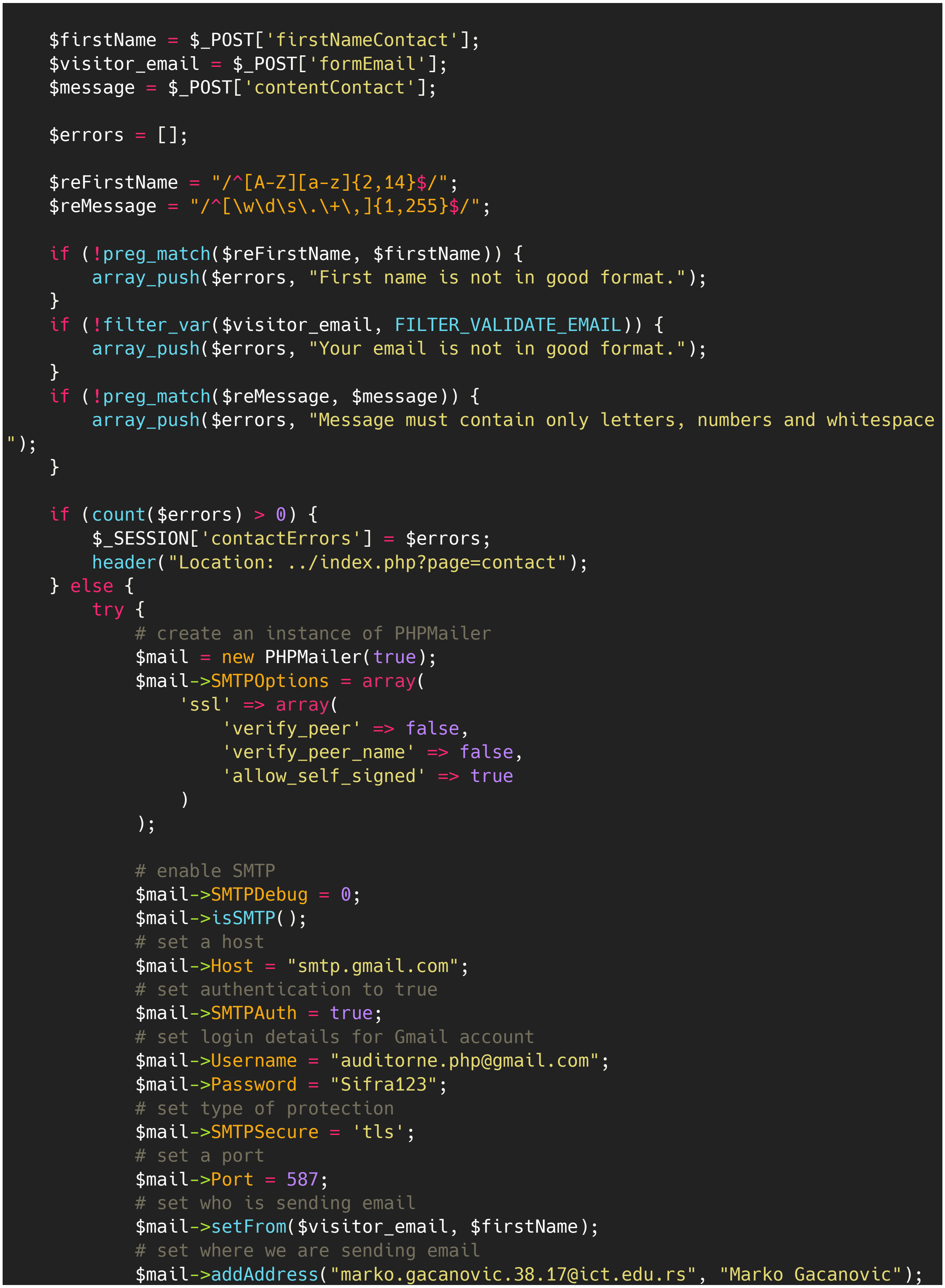
```
<?php
session_start();

if (isset($_POST['btnUpdateUser']) && $_SESSION['user']->roleName == 'admin') {
    require_once "../../config/connection.php";

    $idUser = $_POST['hiddenUserId'];
```



```
    $firstName = $_POST['tbFirstName'];
    $lastName = $_POST['tbLastName'];
    $email = $_POST['tbEmail'];
    $username = $_POST['tbUsername'];
    $password = $_POST['tbPassword'];
    $role = $_POST['ddlRole'];

    $dateForm = $_POST['dateRegistration'];
    $dateArray = explode("-", $dateForm);
    $timestamp = mktime(0, 0, 0, $dateArray[1], $dateArray[2], $dateArray[0]);
    $date = date("Y-m-d H:i:s", $timestamp);

    $active = isset($_POST['chbActive']) ? $_POST['chbActive'] : 0;

    $reFirstLastName = "/^[A-ZŠĐČĆŽ][a-zšđčćž]{2,15}$/";
    $reUsername = "/^[a-zšđčćž0-9\_]{4,15}$/";
    $rePassword = "/^[A-z0-9]{6,20}$/";

    $errors = [];

    if (!preg_match($reFirstLastName, $firstName)) {
        array_push($errors, "First Name isn't in good format!");
    }
    if (!preg_match($reFirstLastName, $lastName)) {
        array_push($errors, "Last Name isn't in good format!");
    }
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        array_push($errors, "Email isn't in good format!");
    }
    if (!preg_match($reUsername, $username)) {
        array_push($errors, "Username isn't in good format!");
    }
    if ($role == 0) {
        array_push($errors, "You didn't choose role!");
    }

    if (count($errors)) {
        $_SESSION['message'] = $errors;
        header("Location: ../../index.php?page=adminUsers");
    } else {
        if ($password == "") {
            $queryUpdateUser = "UPDATE user SET first_name = :firstName, last_name = :
lastName, email = :email, username = :username, date_registration = :dateRegistration,
 active = :active, idRole = :idRole WHERE idUser = :id";

            $prepare = $conn->prepare($queryUpdateUser);
            $prepare->bindParam(":firstName", $firstName);
            $prepare->bindParam(":lastName", $lastName);
            $prepare->bindParam(":email", $email);
            $prepare->bindParam(":username", $username);
            $prepare->bindParam(":dateRegistration", $date);
            $prepare->bindParam(":active", $active);
            $prepare->bindParam(":idRole", $role);
            $prepare->bindParam(":id", $idUser);
            if ($prepare->execute()) {
                $_SESSION['message'] = "User successfully updated.";
                header("Location: ../../index.php?page=adminUsers");
            } else {
                $_SESSION['message'] = "Error, user isn't updated.";
                header("Location: ../../index.php?page=adminUsers&error");
            }
        } else {
            if (!preg_match($rePassword, $password)) {
                $_SESSION['message'] = "You didn't enter correctly all data, and check
 again password, please.";
            } else {
                $password = md5($password);
                $queryUpdateUser = "UPDATE user SET first_name = :firstName, last_name
 = :lastName, email = :email, username = :username, password = :password, date_registr
ation = :dateRegistration, active = :active, idRole = :idRole WHERE idUser = :id";

                $prepare = $conn->prepare($queryUpdateUser);
                $prepare->bindParam(":firstName", $firstName);
                $prepare->bindParam(":lastName", $lastName);
                $prepare->bindParam(":email", $email);
                $prepare->bindParam(":username", $username);
                $prepare->bindParam(":password", $password);
                $prepare->bindParam(":dateRegistration", $date);
                $prepare->bindParam(":active", $active);
                $prepare->bindParam(":idRole", $role);
                $prepare->bindParam(":id", $idUser);

                if ($prepare->execute()) {
                    $_SESSION['message'] = "User successfully updated.";
                    header("Location: ../../index.php?page=adminUsers");
                } else {
                    $_SESSION['message'] = "Error, user isn't updated.";
                    header("Location: ../../index.php?page=adminUsers&error");
                }
            }
        }
    }
}
```

3.3.4 formToEmail.php

```
<?php

use PHPMailer\PHPMailer\PHPMailer;
session_start();

if (isset($_POST['btnSendMessage'])) {
    require "../assets/vendor/PHPMailer/src/PHPMailer.php";
    require "../assets/vendor/PHPMailer/src/SMTP.php";
    require "../assets/vendor/PHPMailer/src/Exception.php";
```



```php
    $firstName = $_POST['firstNameContact'];
    $visitor_email = $_POST['formEmail'];
    $message = $_POST['contentContact'];

    $errors = [];

    $reFirstName = "/^[A-Z][a-z]{2,14}$/";
    $reMessage = "/^[\w\d\s\.\+\,]{1,255}$/";

    if (!preg_match($reFirstName, $firstName)) {
        array_push($errors, "First name is not in good format.");
    }
    if (!filter_var($visitor_email, FILTER_VALIDATE_EMAIL)) {
        array_push($errors, "Your email is not in good format.");
    }
    if (!preg_match($reMessage, $message)) {
        array_push($errors, "Message must contain only letters, numbers and whitespace
");
    }

    if (count($errors) > 0) {
        $_SESSION['contactErrors'] = $errors;
        header("Location: ../index.php?page=contact");
    } else {
        try {
            # create an instance of PHPMailer
            $mail = new PHPMailer(true);
            $mail->SMTPOptions = array(
                'ssl' => array(
                    'verify_peer' => false,
                    'verify_peer_name' => false,
                    'allow_self_signed' => true
                )
            );

            # enable SMTP
            $mail->SMTPDebug = 0;
            $mail->isSMTP();
            # set a host
            $mail->Host = "smtp.gmail.com";
            # set authentication to true
            $mail->SMTPAuth = true;
            # set login details for Gmail account
            $mail->Username = "auditorne.php@gmail.com";
            $mail->Password = "Sifra123";
            # set type of protection
            $mail->SMTPSecure = 'tls';
            # set a port
            $mail->Port = 587;
            # set who is sending email
            $mail->setFrom($visitor_email, $firstName);
            # set where we are sending email
            $mail->addAddress("marko.gacanovic.38.17@ict.edu.rs", "Marko Gacanovic");
            $mail->isHTML(true);
            # set subject
            $mail->Subject = "New Form submission.";
            # set body
            $mail-
>Body = "You have received a new message from the user $firstName.\n" . "Here is the m
essage:\n $message " . $to = "marko.gacanovic.38.17@ict.edu.rs";
            # send an email
            $mail->send();
            header("Location: ../thanks.html");
        } catch (Exception $ex) {
            echo $ex->getMessage();
        }
    }
} else {
    echo "Yoou need to submit the form!";
}
```

3.3.5 functions.php

```
<?php

function getMenuForAllUsers()
{
    return executeQuery("SELECT * FROM menu WHERE idMenuGroup = 1");
}

function getMenuForAuthorizedUsers()
{
    return executeQuery("SELECT * FROM menu WHERE idMenuGroup = 2");
}

function getMenuForAdmin()
{
    return executeQuery("SELECT * FROM menu WHERE idMenuGroup = 3");
}

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

3.3.6 login.php

```
<?php

session_start();

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

```php
    $email = $_POST['tbEmail'];
    $password = $_POST['tbPassword'];

    $errors = [];

    $rePassword = "/^[A-z0-9]{6,20}$/";

    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $errors[] = "Email is not good!";
    }

    if (!preg_match($rePassword, $password)) {
        $errors[] = "Password is not good!";
    }

    if (count($errors) > 0) {
        $_SESSION['errors'] = $errors;
        header("Location: ../index.php");
    } else {
        $queryLoginUser = "SELECT u.idUser, u.first_name, u.last_name, u.email, u.username, r.name AS roleName FROM user u INNER JOIN role r ON u.idRole = r.idRole WHERE email = :email AND password = :password";

        $stmt = $conn->prepare($queryLoginUser);

        $password = md5($password);

        $stmt->bindParam(":email", $email);
        $stmt->bindParam(":password", $password);

        $stmt->execute();
        $user = $stmt->fetch();

        if ($user) {
            $_SESSION['user'] = $user;
            if(isset($_SESSION['user']) && $_SESSION['user']->roleName=='admin'):
                header("Location: ../index.php?page=adminUsers");
            else:
                header("Location: ../index.php?page=home");
            endif;
        } else {
            $_SESSION['errors'] = "Wrong email or password!";
            header("Location: ../index.php");
        }
    }
}
```

3.3.7 logout.php

```php
<?php

session_start();
unset($_SESSION['user']);
```

```
header("Location: ../index.php");
```

3.3.8 register.php

```
<?php

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

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

$code = 404;
$data = null;

if (isset($_POST['send'])) {
    $firstName = $_POST['firstName'];
    $lastName = $_POST['lastName'];
    $email = $_POST['email'];
    $username = $_POST['username'];
    $password = md5($_POST['password']);

    $errors = [];

    $reFirstLastName = "/^[A-ZŠĐČĆŽ][a-zšđčćž]{2,15}$/";
    $reUsername = "/^[a-zšđčćž0-9\_]{4,15}$/";
    $rePassword = "/^[A-z0-9]{6,20}$/";

    if (!preg_match($reFirstLastName, $firstName)) {
        array_push($errors, "Fist Name field isn't proprely filled!");
    } elseif(!$firstName) {
        array_push($errors, "First Name is required!");
    }
    if (!preg_match($reFirstLastName, $lastName)) {
        array_push($errors, "Last Name field isn't proprely filled!");
    } elseif (!$lastName) {
        array_push($errors, "Last Name is required!");
    }
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        array_push($errors, "Email isn't in good format!");
    }
    if (!preg_match($reUsername, $username)) {
        array_push($errors, "Username field isn't proprely filled!");
    } elseif (!$username) {
        array_push($errors, "Username is required!");
    }
    if (!preg_match($rePassword, $_POST['password'])) {
        array_push($errors, "Password field isn't proprely filled!");
    } elseif (!$_POST['password']) {
        array_push($errors, "Password is required!");
    }

    if (count($errors)) {
        $code = 422;
        $data = $errors;
    } else {
        $queryRegisterUser = "INSERT INTO user (first_name, last_name, email, username, password, active, idRole) VALUES (:firstName, :lastName, :email, :username, :password, 1, 2)";

        $stmt = $conn->prepare($queryRegisterUser);
        $stmt->bindParam(":firstName", $firstName);
        $stmt->bindParam(":lastName", $lastName);
        $stmt->bindParam(":email", $email);
        $stmt->bindParam(":username", $username);
        $stmt->bindParam(":password", $password);

        try {
            $code = $stmt->execute() ? 201 : 500;
        } catch(PDOException $ex) {
            echo "Registration failed: " . $ex->getMessage();
        }
    }
}

http_response_code($code);
echo json_encode($data);
```

3.4 Config

3.4.1 connection.php

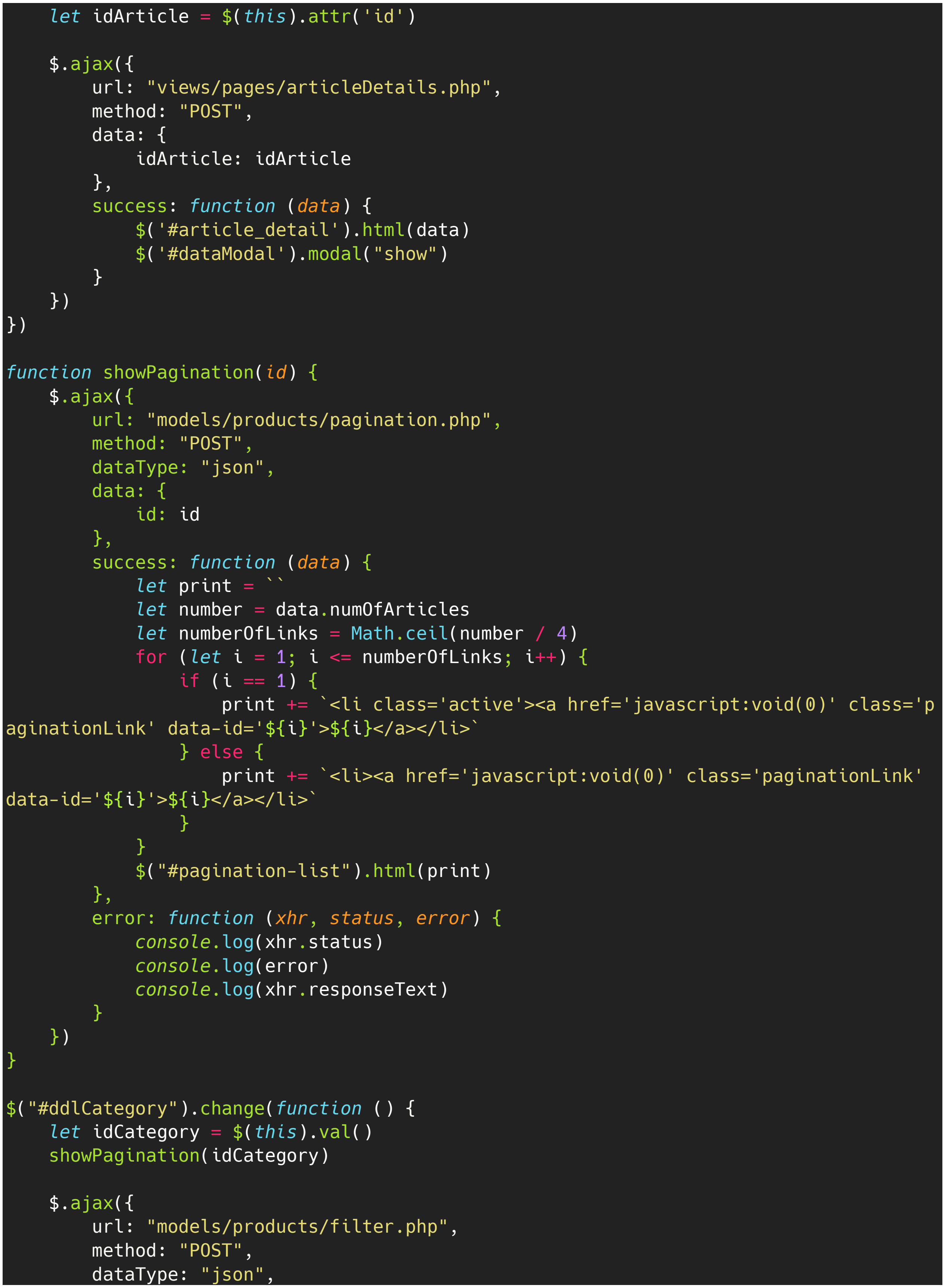
```
<?php

$serverName = "localhost";
$dbName = "gigatronic_shop";
$username = "root";
$password = "";

try {
    $conn = new PDO("mysql:host=$serverName;dbname=$dbName;charset=utf8", $username, $password);
    $conn->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $ex) {
    echo "Error with database: " . $ex->getMessage();
}

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

3.5 Assets

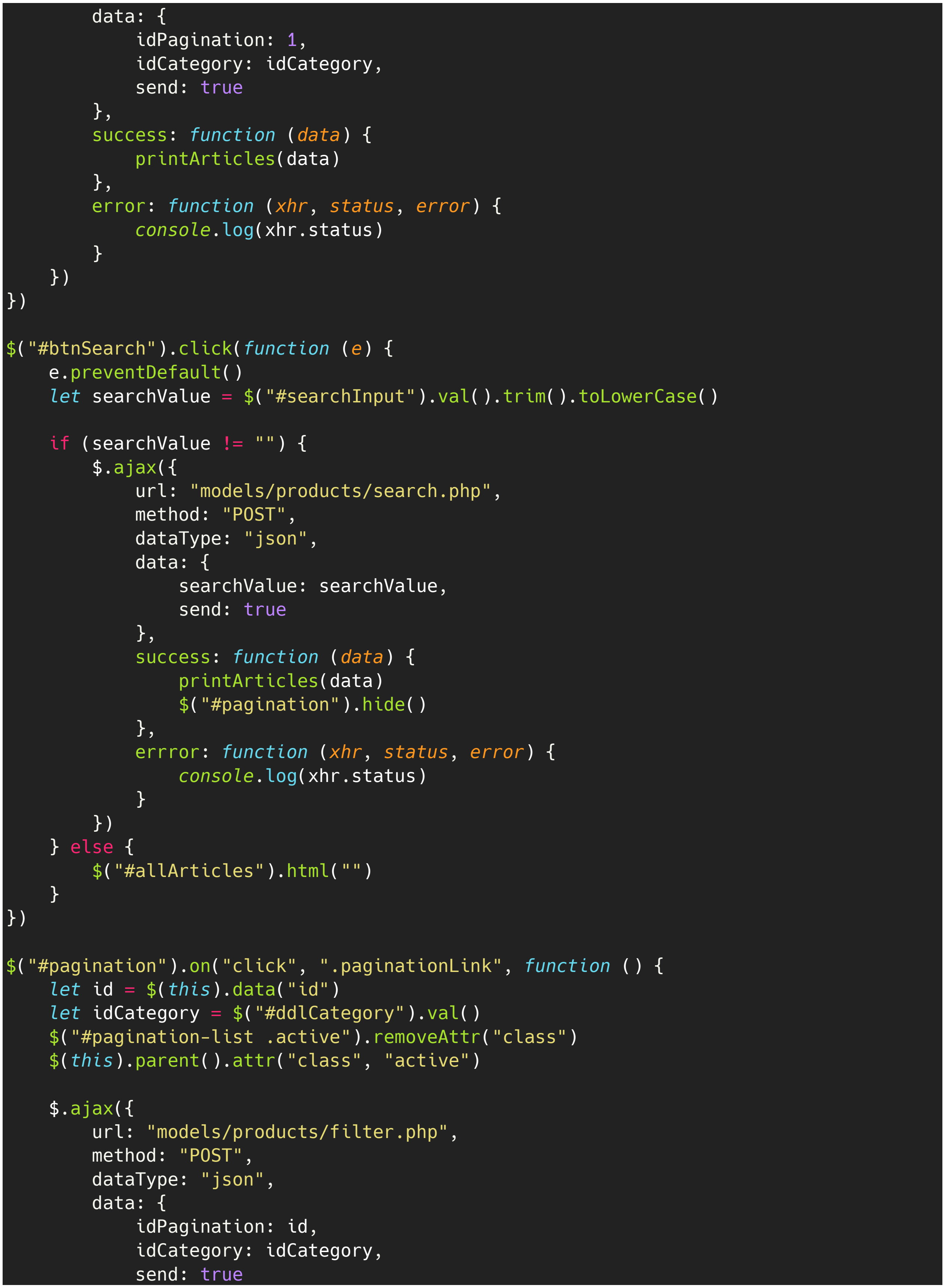
3.5.1 js

3.5.1.1 main.js

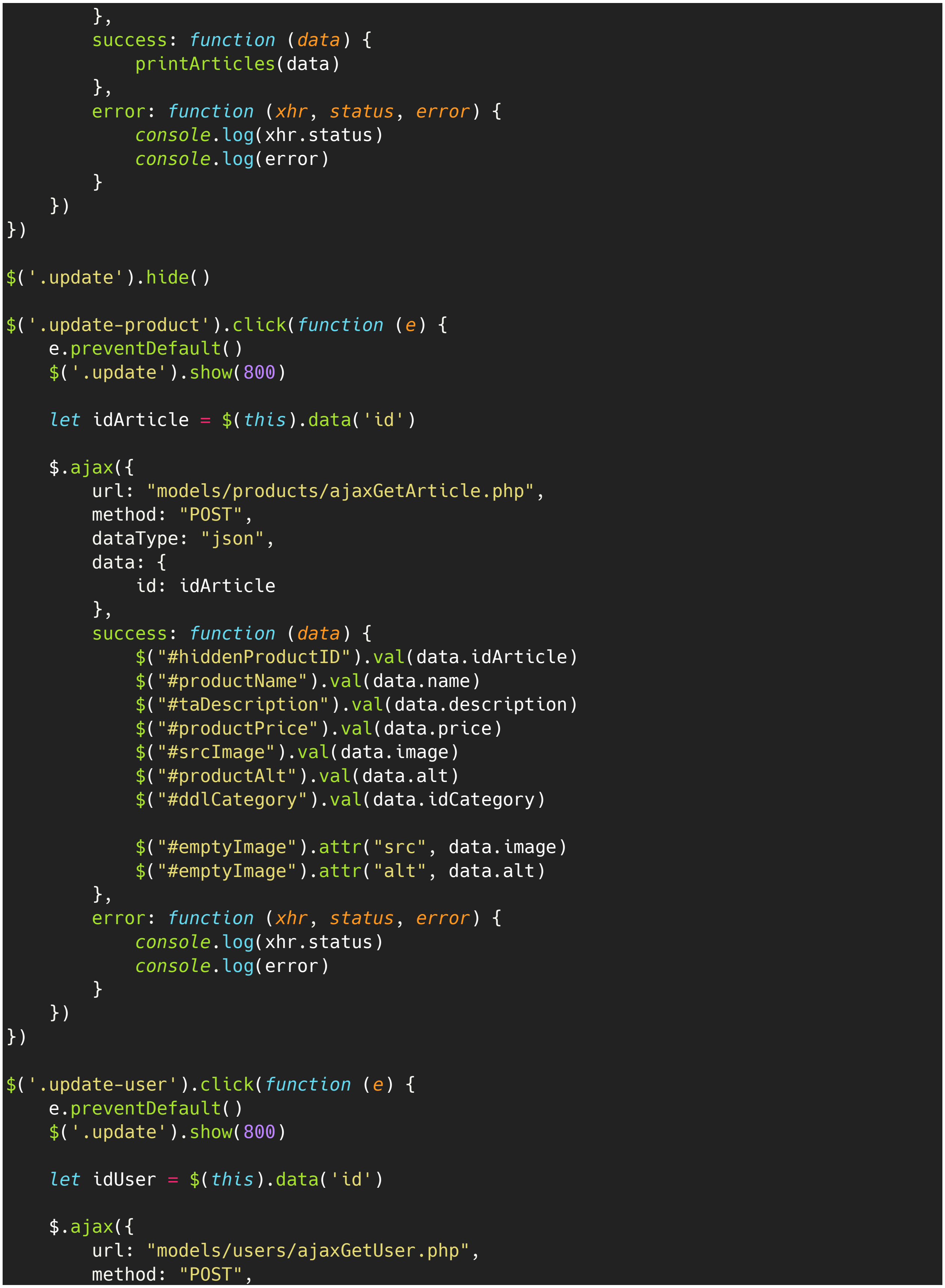
```
const BASE_URL = "http://localhost/Gigatronic_Shop/"

$(document).ready(function () {
    populateAllArticles()
    showPagination(0)
})

function populateAllArticles() {
    $.ajax({
        url: "models/products/ajaxArticles.php",
        method: "POST",
        dataType: "json",
        data: {
            id: 1
        },
        success: function (data) {
            printArticles(data)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
        }
    })
}

function printArticles(articles) {
    let print = ""
    for (let article of articles) {
        print += `
        <div class="col-md-3 col-lg-3 my-4">
            <div class="featured-container p-5">
                <img src="${article.image}" alt="${article.alt}" class="img-fluid" />
                <button name="view" value="View" class="view_data" id="${article.idArticle}">
                    <i class="fa fa-search"></i>
                </button>
            </div>
            <h6 class="text-center my-2">${article.name}</h6>
            <h6 class="text-center">
                <span>$${article.price}</span>
            </h6>
        </div>
        `
    }
    $("#allArticles").html(print)
}

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



```
    let idArticle = $(this).attr('id')

    $.ajax({
        url: "views/pages/articleDetails.php",
        method: "POST",
        data: {
            idArticle: idArticle
        },
        success: function (data) {
            $('#article_detail').html(data)
            $('#dataModal').modal("show")
        }
    })
})

function showPagination(id) {
    $.ajax({
        url: "models/products/pagination.php",
        method: "POST",
        dataType: "json",
        data: {
            id: id
        },
        success: function (data) {
            let print = ``
            let number = data.numOfArticles
            let numberOfLinks = Math.ceil(number / 4)
            for (let i = 1; i <= numberOfLinks; i++) {
                if (i == 1) {
                    print += `<li class='active'><a href='javascript:void(0)' class='paginationLink' data-id='${i}'>${i}</a></li>`
                } else {
                    print += `<li><a href='javascript:void(0)' class='paginationLink' data-id='${i}'>${i}</a></li>`
                }
            }
            $("#pagination-list").html(print)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
            console.log(error)
            console.log(xhr.responseText)
        }
    })
}

$("#ddlCategory").change(function () {
    let idCategory = $(this).val()
    showPagination(idCategory)

    $.ajax({
        url: "models/products/filter.php",
        method: "POST",
        dataType: "json",
```



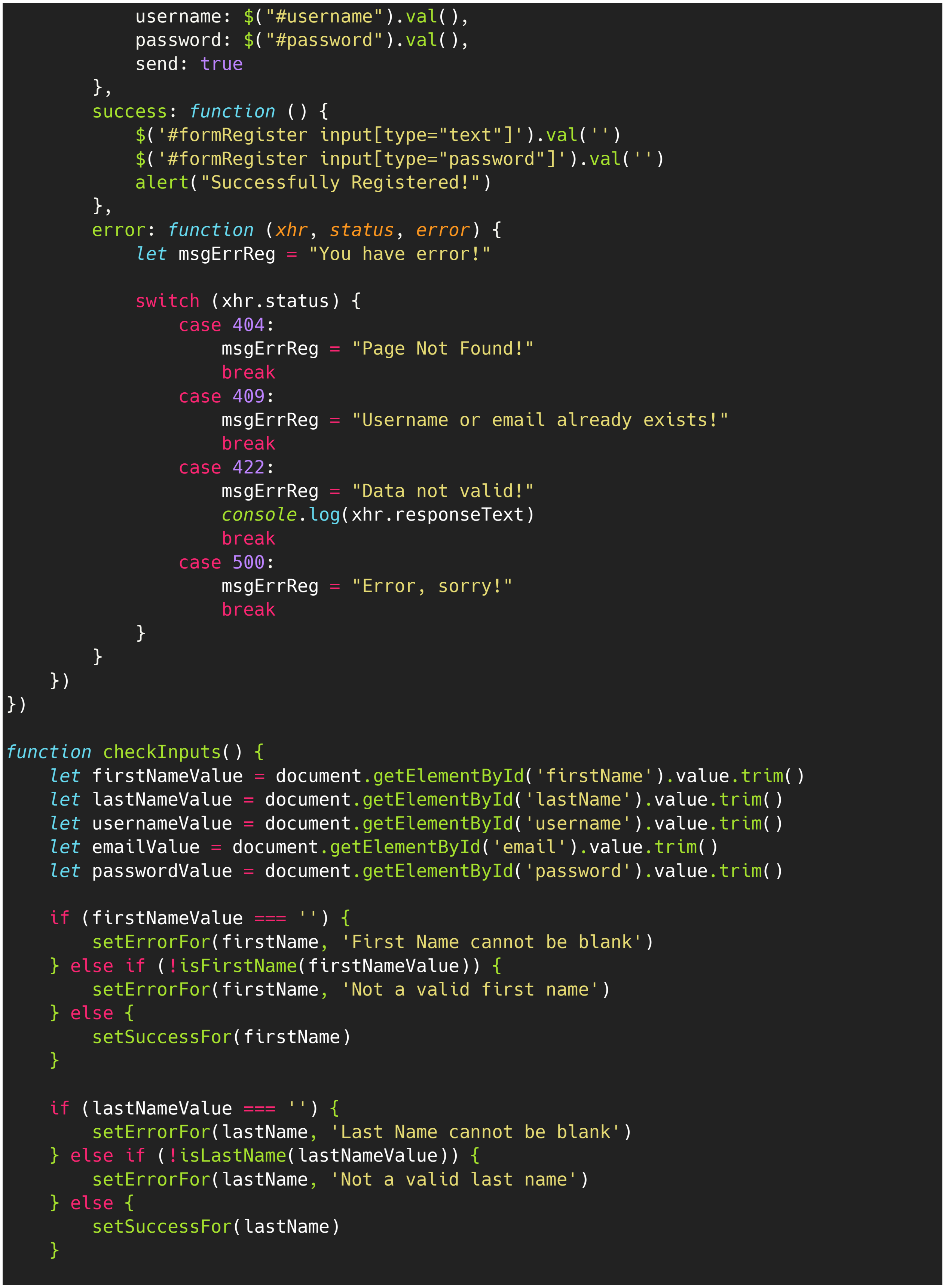
```
        data: {
            idPagination: 1,
            idCategory: idCategory,
            send: true
        },
        success: function (data) {
            printArticles(data)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
        }
    })
})

$("#btnSearch").click(function (e) {
    e.preventDefault()
    let searchValue = $("#searchInput").val().trim().toLowerCase()

    if (searchValue != "") {
        $.ajax({
            url: "models/products/search.php",
            method: "POST",
            dataType: "json",
            data: {
                searchValue: searchValue,
                send: true
            },
            success: function (data) {
                printArticles(data)
                $("#pagination").hide()
            },
            errror: function (xhr, status, error) {
                console.log(xhr.status)
            }
        })
    } else {
        $("#allArticles").html("")
    }
})

$("#pagination").on("click", ".paginationLink", function () {
    let id = $(this).data("id")
    let idCategory = $("#ddlCategory").val()
    $("#pagination-list .active").removeAttr("class")
    $(this).parent().attr("class", "active")

    $.ajax({
        url: "models/products/filter.php",
        method: "POST",
        dataType: "json",
        data: {
            idPagination: id,
            idCategory: idCategory,
            send: true
```



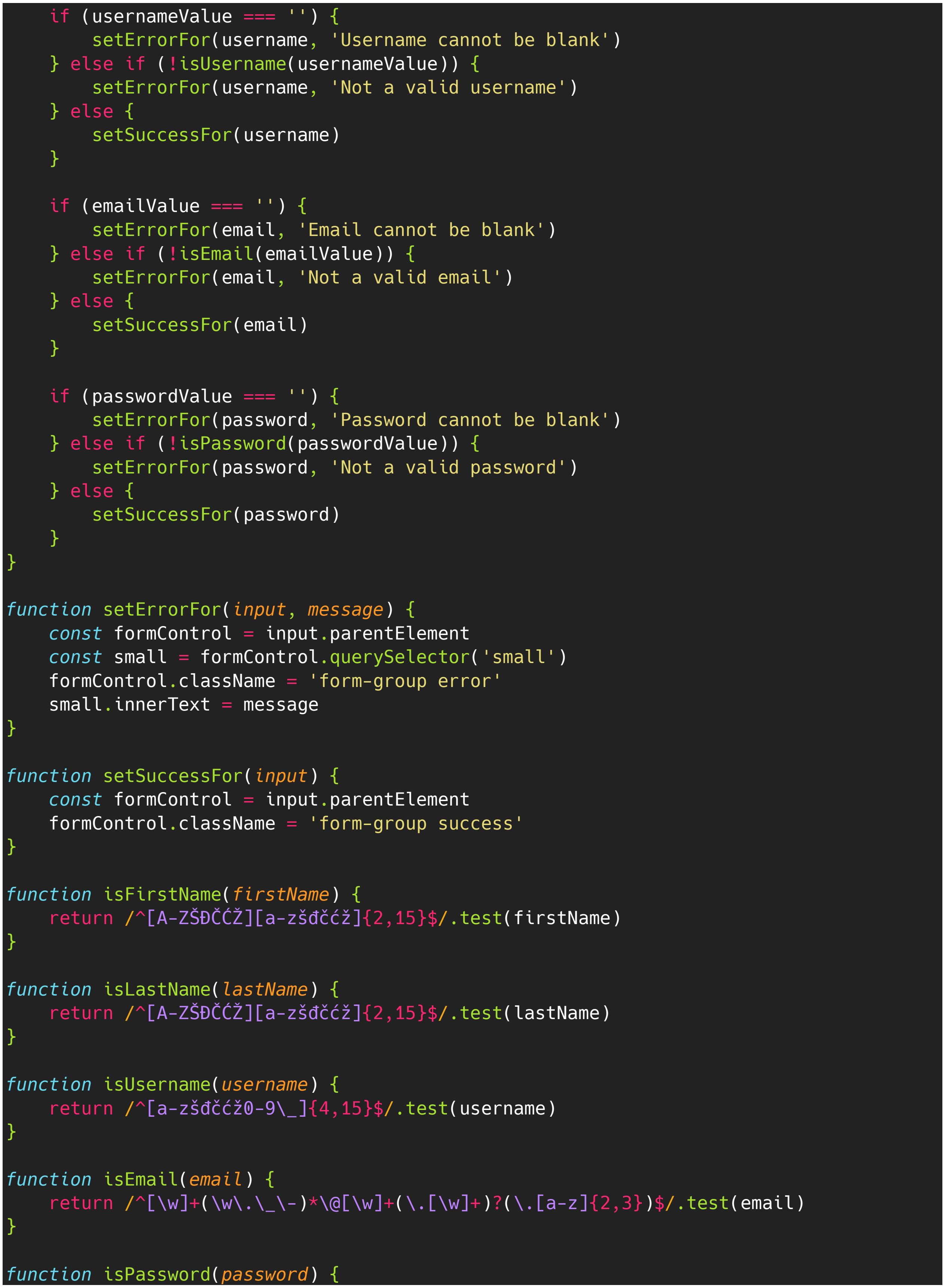
```
        },
        success: function (data) {
            printArticles(data)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
            console.log(error)
        }
    })
})

$('.update').hide()

$('.update-product').click(function (e) {
    e.preventDefault()
    $('.update').show(800)

    let idArticle = $(this).data('id')

    $.ajax({
        url: "models/products/ajaxGetArticle.php",
        method: "POST",
        dataType: "json",
        data: {
            id: idArticle
        },
        success: function (data) {
            $("#hiddenProductID").val(data.idArticle)
            $("#productName").val(data.name)
            $("#taDescription").val(data.description)
            $("#productPrice").val(data.price)
            $("#srcImage").val(data.image)
            $("#productAlt").val(data.alt)
            $("#ddlCategory").val(data.idCategory)

            $("#emptyImage").attr("src", data.image)
            $("#emptyImage").attr("alt", data.alt)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
            console.log(error)
        }
    })
})

$('.update-user').click(function (e) {
    e.preventDefault()
    $('.update').show(800)

    let idUser = $(this).data('id')

    $.ajax({
        url: "models/users/ajaxGetUser.php",
        method: "POST",
        dataType: "json",
        data: {
            id: idUser
        },
        success: function (data) {
            $("#hiddenUserId").val(data.idUser)
            $("#tbFirstName").val(data.first_name)
            $("#tbLastName").val(data.last_name)
            $("#tbEmail").val(data.email)
            $("#tbUsername").val(data.username)
            $("#ddlRole").val(data.idRole)
            let DatetimeArray = data.date_registration.split(" ")
            $("#dateRegistration").val(DatetimeArray[0])
            $("input[name='chbActive']").removeAttr('checked')
            if (data.active == 1) {
                $("input[name='chbActive']").prop('checked', true)
                $("input[name='chbActive']").val(data.active)
            }
        },
        error: function (xhr, statusTxt, error) {
            let status = xhr.status

            switch (status) {
                case 500:
                    console.log("Error on server")
                    break;
                case 404:
                    console.log("Page not found")
                    break;
                default:
                    console.log("Error: " + status + " - " + statusTxt)
            }
        }
    })
})
```

3.5.1.2 registration.js

```
const form = document.getElementById('formRegister')

form.addEventListener('submit', e => {
    e.preventDefault()
    checkInputs()
})

$("#btnRegister").click(function () {
    $.ajax({
        url: BASE_URL + "models/register.php",
        method: "POST",
        dataType: "json",
        data: {
            firstName: $("#firstName").val(),
            lastName: $("#lastName").val(),
            email: $("#email").val(),
```



```
            username: $("#username").val(),
            password: $("#password").val(),
            send: true
        },
        success: function () {
            $('#formRegister input[type="text"]').val('')
            $('#formRegister input[type="password"]').val('')
            alert("Successfully Registered!")
        },
        error: function (xhr, status, error) {
            let msgErrReg = "You have error!"

            switch (xhr.status) {
                case 404:
                    msgErrReg = "Page Not Found!"
                    break
                case 409:
                    msgErrReg = "Username or email already exists!"
                    break
                case 422:
                    msgErrReg = "Data not valid!"
                    console.log(xhr.responseText)
                    break
                case 500:
                    msgErrReg = "Error, sorry!"
                    break
            }
        }
    })
})

function checkInputs() {
    let firstNameValue = document.getElementById('firstName').value.trim()
    let lastNameValue = document.getElementById('lastName').value.trim()
    let usernameValue = document.getElementById('username').value.trim()
    let emailValue = document.getElementById('email').value.trim()
    let passwordValue = document.getElementById('password').value.trim()

    if (firstNameValue === '') {
        setErrorFor(firstName, 'First Name cannot be blank')
    } else if (!isFirstName(firstNameValue)) {
        setErrorFor(firstName, 'Not a valid first name')
    } else {
        setSuccessFor(firstName)
    }

    if (lastNameValue === '') {
        setErrorFor(lastName, 'Last Name cannot be blank')
    } else if (!isLastName(lastNameValue)) {
        setErrorFor(lastName, 'Not a valid last name')
    } else {
        setSuccessFor(lastName)
    }
```



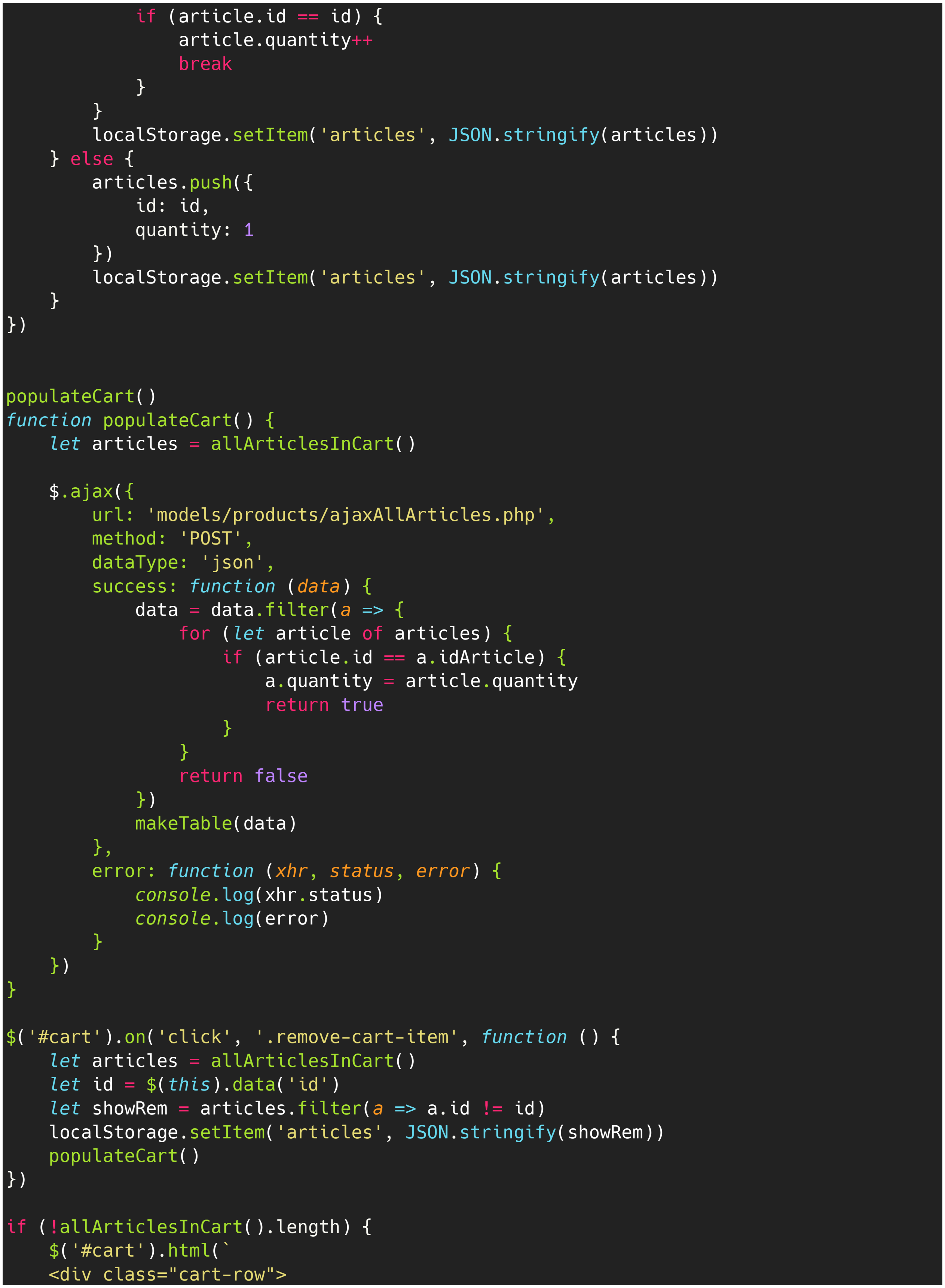
```
    if (usernameValue === '') {
        setErrorFor(username, 'Username cannot be blank')
    } else if (!isUsername(usernameValue)) {
        setErrorFor(username, 'Not a valid username')
    } else {
        setSuccessFor(username)
    }

    if (emailValue === '') {
        setErrorFor(email, 'Email cannot be blank')
    } else if (!isEmail(emailValue)) {
        setErrorFor(email, 'Not a valid email')
    } else {
        setSuccessFor(email)
    }

    if (passwordValue === '') {
        setErrorFor(password, 'Password cannot be blank')
    } else if (!isPassword(passwordValue)) {
        setErrorFor(password, 'Not a valid password')
    } else {
        setSuccessFor(password)
    }
}

function setErrorFor(input, message) {
    const formControl = input.parentElement
    const small = formControl.querySelector('small')
    formControl.className = 'form-group error'
    small.innerText = message
}

function setSuccessFor(input) {
    const formControl = input.parentElement
    formControl.className = 'form-group success'
}

function isFirstName(firstName) {
    return /^[A-ZŠĐČĆŽ][a-zšđčćž]{2,15}$/.test(firstName)
}

function isLastName(lastName) {
    return /^[A-ZŠĐČĆŽ][a-zšđčćž]{2,15}$/.test(lastName)
}

function isUsername(username) {
    return /^[a-zšđčćž0-9\_]{4,15}$/.test(username)
}

function isEmail(email) {
    return /^[\w]+(\w\.\_\-)*\@[\w]+(\.[\w]+)?(\.[a-z]{2,3})$/.test(email)
}

function isPassword(password) {
    return /^[A-z0-9]{6,20}$/.test(password)
}
```

3.5.1.3 contact.js

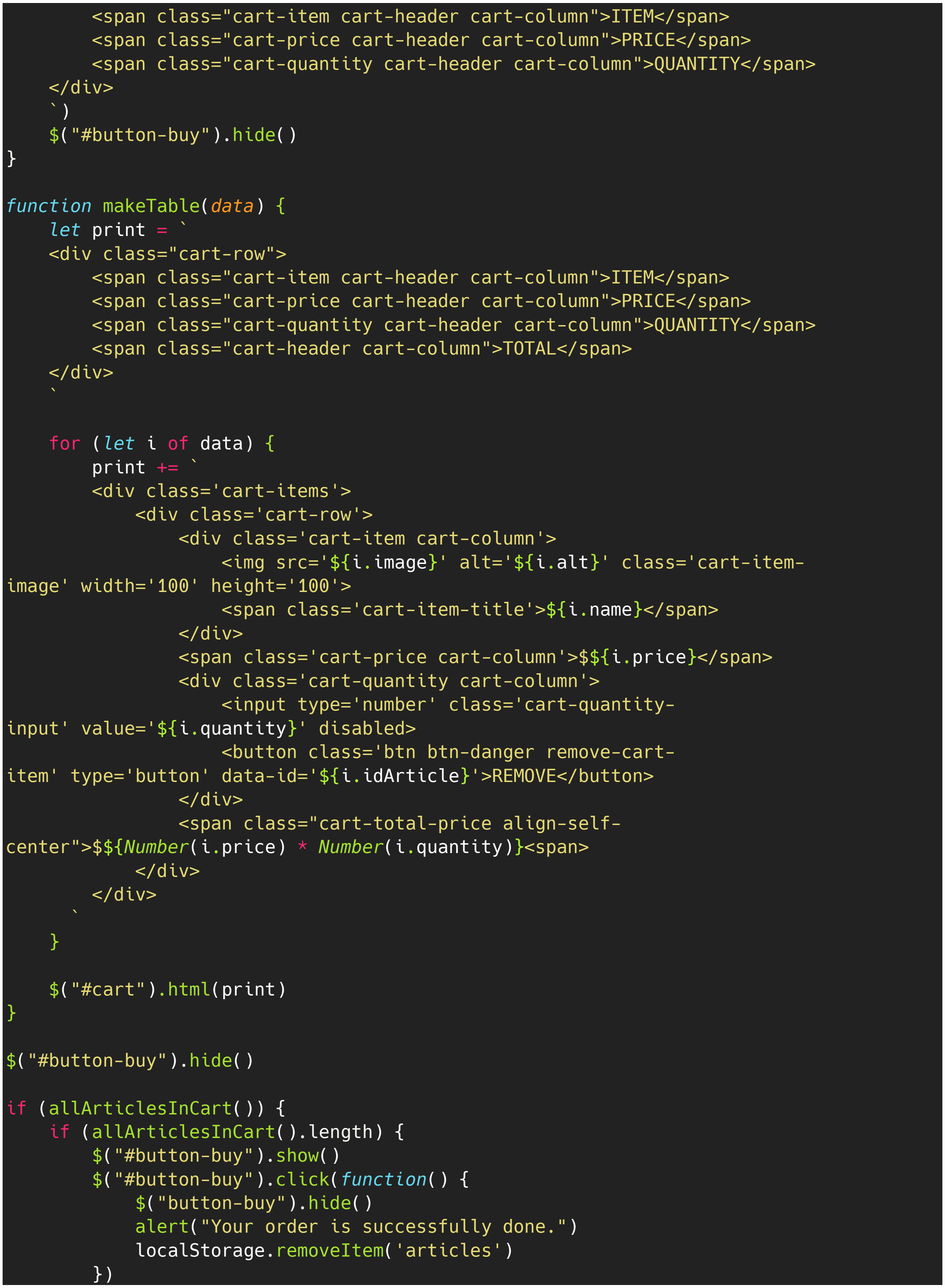
```
$(document).ready(function() {
    $("#firstNameContact").val("")
    $("#formEmail").val("")
    $("#contentContact").val("")
    checkFormClientSide()
})

function checkFormClientSide() {
    $("#firstNameContact").blur(checkFirstName)
    $("#firstNameContact").addClass("cssBorder")
    $("#formEmail").blur(checkEmail)
    $("#formEmail").addClass("cssBorder")
    $("#contentContact").blur(checkMessage)
    $("#contentContact").addClass("cssBorder")

    $("#btnSendMessage").click(checkForm)

    let reFirstName = /^[A-Z][a-z]{2,14}$/
    let reEmail = /^[\w]+[\.\w\d]*\@[\w]+([\.][\w]+)+$/
    let reMessage = /^[\w\d\s\.\,\+]+$/

    function checkFirstName() {
        let firstName = $("#firstNameContact").val()
        if (!reFirstName.test(firstName)) {
            $("#firstNameContact").addClass("notCorrect")
        } else {
            $("#firstNameContact").removeClass("notCorrect")
        }
    }
    function checkEmail() {
        let email = $("#formEmail").val()
        if (!reEmail.test(email)) {
            $("#formEmail").addClass("notCorrect")
        } else {
            $("#formEmail").removeClass("notCorrect")
        }
    }
    function checkMessage() {
        let message = $("#contentContact").val()
        if (!reMessage.test(message)) {
            $("#contentContact").addClass("notCorrect")
        } else {
            $("#contentContact").removeClass("notCorrect")
        }
    }

    function checkForm() {
        var firstName = $("#firstNameContact").val()
        var email = $("#formEmail").val()
        var message = $("#contentContact").val()

        var arrayMistakes = []

        if ($('.notCorrect').length || (firstName == "") || (email == "") || (message
== "")) {
            if (!reFirstName.test(firstName))
                arrayMistakes.push("Minimum 3 and maximum 15 characters for first name
.")
            if (!reEmail.test(email))
                arrayMistakes.push("Email is not in good format.")
            if (!reMessage.test(message))
                arrayMistakes.push("Message must contain only uppercase, lowercase, nu
mbers and .,+")

            checkEmpty(firstName, email, message)
        } else {
            $("#notification").html("")
        }
    }

    function checkEmpty(firstName, email, message) {
        if (firstName == "") {
            $("#firstNameContact").addClass("notCorrect")
        }
        if (email == "") {
            $("#formEmail").addClass("notCorrect")
        }
        if (message == "") {
            $("#contentContact").addClass("notCorrect")
        }
    }
}
```

3.5.1.4 cart.js

```
function allArticlesInCart() {
    return JSON.parse(localStorage.getItem('articles'))
}

if (!allArticlesInCart()) {
    localStorage.setItem('articles', JSON.stringify([]))
    $("button-buy").hide()
}

$(document).on('click', '.add-to-cart', function () {
    alert("You successfully added item to cart.")
    let id = $(this).data('id')
    let articles = allArticlesInCart()

    if (articles.filter(a => a.id == id).length) {
        for (let article of articles) {
```



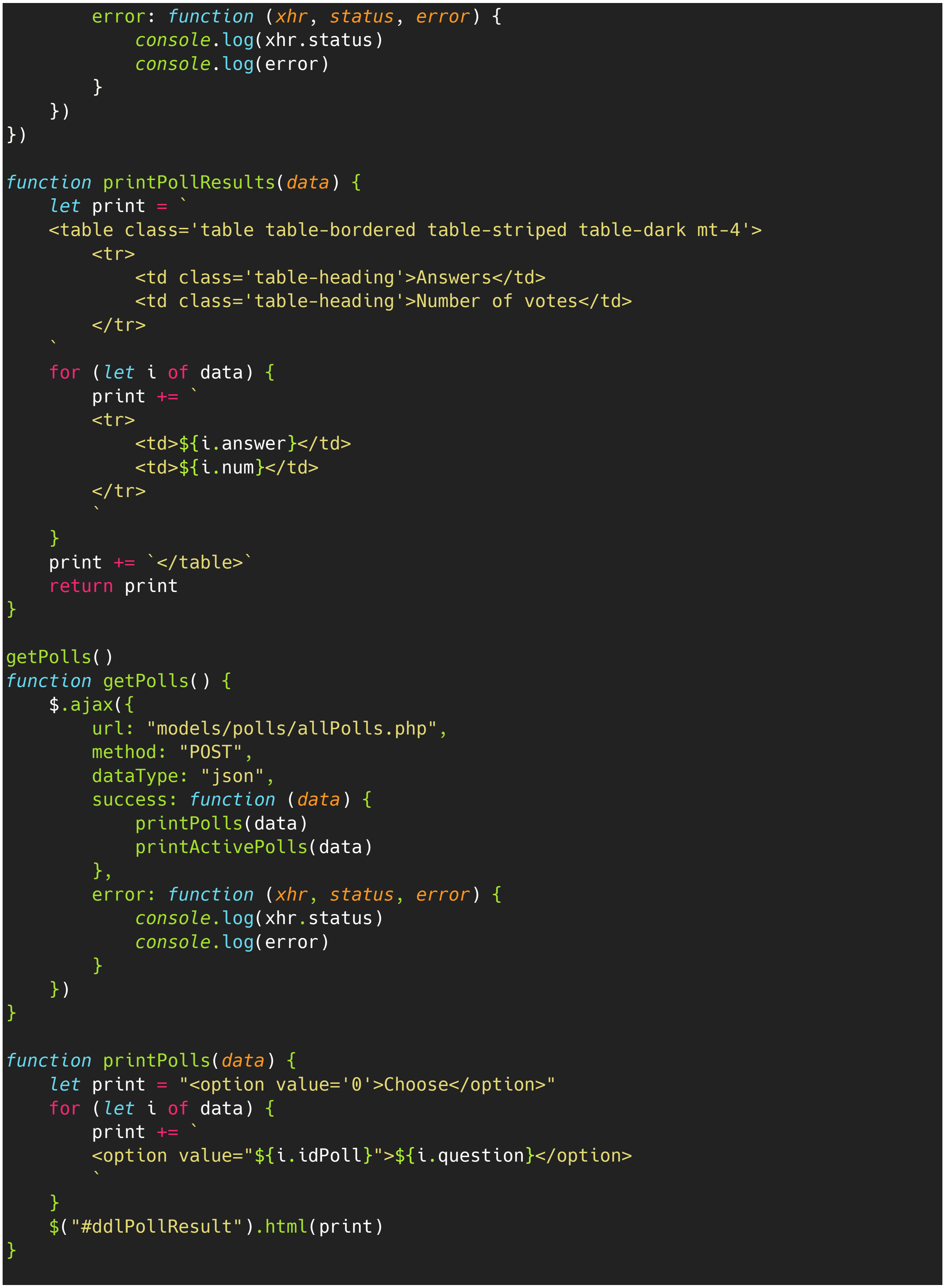
```
            if (article.id == id) {
                article.quantity++
                break
            }
        }
        localStorage.setItem('articles', JSON.stringify(articles))
    } else {
        articles.push({
            id: id,
            quantity: 1
        })
        localStorage.setItem('articles', JSON.stringify(articles))
    }
})


populateCart()
function populateCart() {
    let articles = allArticlesInCart()

    $.ajax({
        url: 'models/products/ajaxAllArticles.php',
        method: 'POST',
        dataType: 'json',
        success: function (data) {
            data = data.filter(a => {
                for (let article of articles) {
                    if (article.id == a.idArticle) {
                        a.quantity = article.quantity
                        return true
                    }
                }
                return false
            })
            makeTable(data)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
            console.log(error)
        }
    })
}

$('#cart').on('click', '.remove-cart-item', function () {
    let articles = allArticlesInCart()
    let id = $(this).data('id')
    let showRem = articles.filter(a => a.id != id)
    localStorage.setItem('articles', JSON.stringify(showRem))
    populateCart()
})

if (!allArticlesInCart().length) {
    $('#cart').html(`
    <div class="cart-row">
```



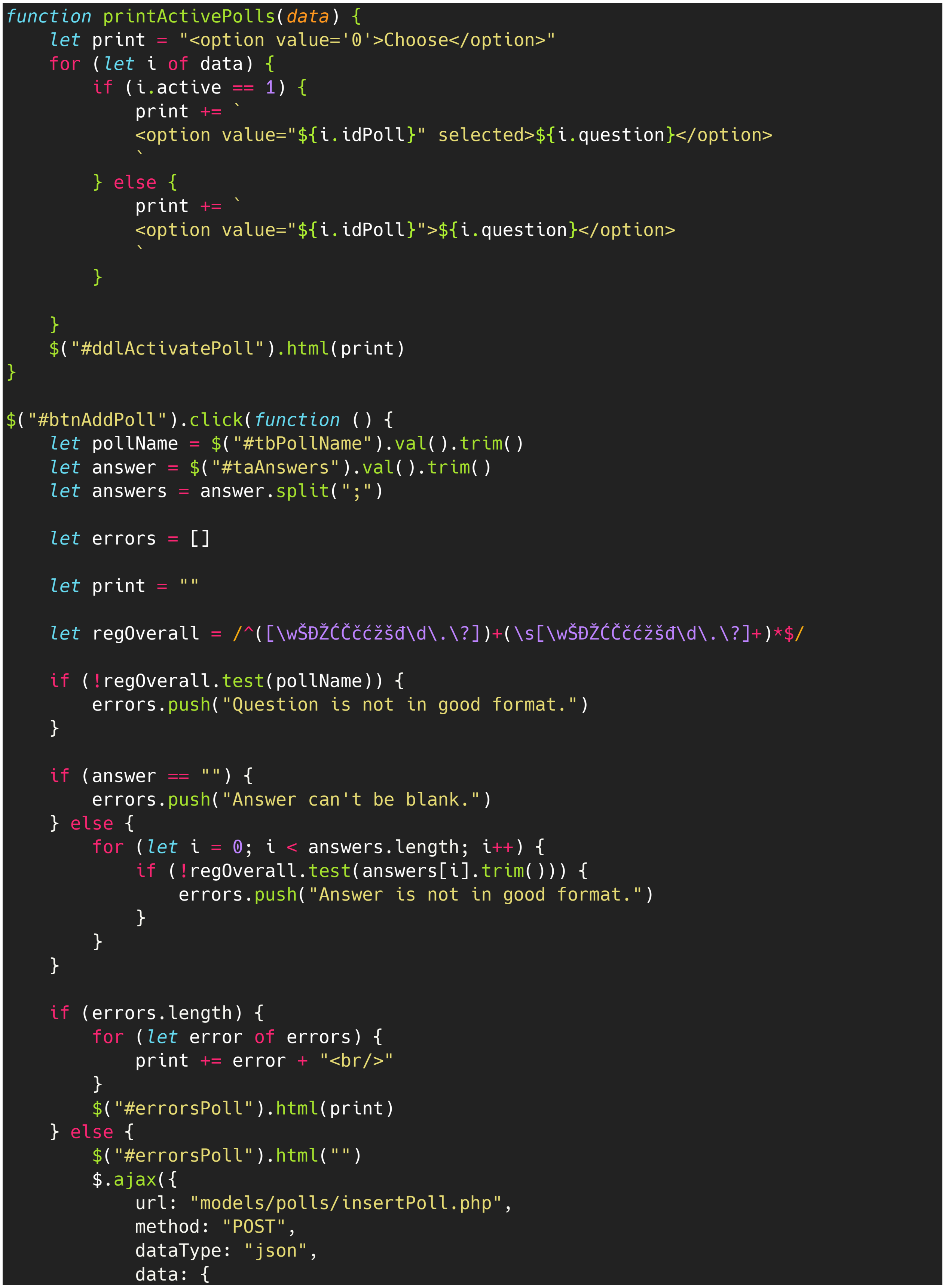
```
        <span class="cart-item cart-header cart-column">ITEM</span>
        <span class="cart-price cart-header cart-column">PRICE</span>
        <span class="cart-quantity cart-header cart-column">QUANTITY</span>
    </div>
    `)
    $("#button-buy").hide()
}

function makeTable(data) {
    let print = `
    <div class="cart-row">
        <span class="cart-item cart-header cart-column">ITEM</span>
        <span class="cart-price cart-header cart-column">PRICE</span>
        <span class="cart-quantity cart-header cart-column">QUANTITY</span>
        <span class="cart-header cart-column">TOTAL</span>
    </div>
    `

    for (let i of data) {
        print += `
        <div class='cart-items'>
            <div class='cart-row'>
                <div class='cart-item cart-column'>
                    <img src='${i.image}' alt='${i.alt}' class='cart-item-image' width='100' height='100'>
                    <span class='cart-item-title'>${i.name}</span>
                </div>
                <span class='cart-price cart-column'>$${i.price}</span>
                <div class='cart-quantity cart-column'>
                    <input type='number' class='cart-quantity-input' value='${i.quantity}' disabled>
                    <button class='btn btn-danger remove-cart-item' type='button' data-id='${i.idArticle}'>REMOVE</button>
                </div>
                <span class="cart-total-price align-self-center">$${Number(i.price) * Number(i.quantity)}<span>
            </div>
        </div>
      `
    }

    $("#cart").html(print)
}

$("#button-buy").hide()

if (allArticlesInCart()) {
    if (allArticlesInCart().length) {
        $("#button-buy").show()
        $("#button-buy").click(function() {
            $("button-buy").hide()
            alert("Your order is successfully done.")
            localStorage.removeItem('articles')
        })
    }
}
```

3.5.1.5 poll.js

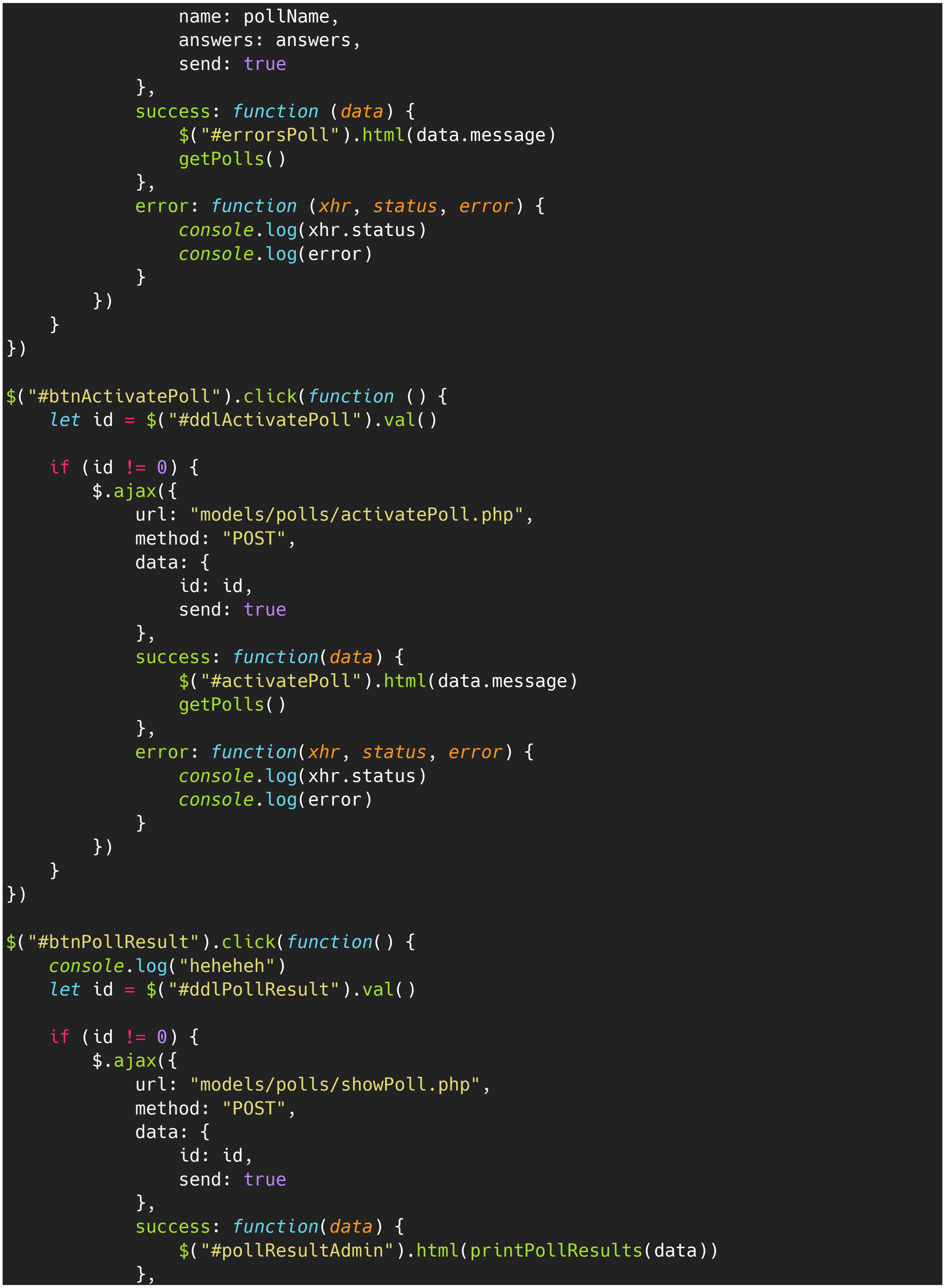
```
$("#btnVote").click(function () {
    let id = $("#hiddenUserPollField").val()
    let answer = $("input[name='rbPoll']:checked").val()

    if (!answer) {
        $("#votingResults").html("You must choose answer.")
    } else {
        $.ajax({
            url: "models/polls/voting.php",
            method: "POST",
            dataType: "json",
            data: {
                id: id,
                answer: answer,
                send: true
            },
            success: function (data) {
                $("#votingResults").html(data.message)
            },
            error: function (xhr, status, error) {
                switch (xhr.status) {
                    case 409:
                        $("#votingResults").html("You aleready voted.")
                        break
                    default:
                        console.log(xhr.status)
                        console.log(error)
                        break
                }
            }
        })
    }
})

$("#btnVotingResult").click(function () {
    let id = $("#activePoll").val()

    $.ajax({
        url: "models/polls/showPoll.php",
        method: "POST",
        dataType: "json",
        data: {
            id: id,
            send: true
        },
        success: function (data) {
            console.log(data)
            $("#votingResults").html(printPollResults(data))
        },
```



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

function printPollResults(data) {
    let print = `
    <table class='table table-bordered table-striped table-dark mt-4'>
        <tr>
            <td class='table-heading'>Answers</td>
            <td class='table-heading'>Number of votes</td>
        </tr>
    `
    for (let i of data) {
        print += `
        <tr>
            <td>${i.answer}</td>
            <td>${i.num}</td>
        </tr>
        `
    }
    print += `</table>`
    return print
}

getPolls()
function getPolls() {
    $.ajax({
        url: "models/polls/allPolls.php",
        method: "POST",
        dataType: "json",
        success: function (data) {
            printPolls(data)
            printActivePolls(data)
        },
        error: function (xhr, status, error) {
            console.log(xhr.status)
            console.log(error)
        }
    })
}

function printPolls(data) {
    let print = "<option value='0'>Choose</option>"
    for (let i of data) {
        print += `
        <option value="${i.idPoll}">${i.question}</option>
        `
    }
    $("#ddlPollResult").html(print)
}
```



```
function printActivePolls(data) {
    let print = "<option value='0'>Choose</option>"
    for (let i of data) {
        if (i.active == 1) {
            print += `
            <option value="${i.idPoll}" selected>${i.question}</option>
            `
        } else {
            print += `
            <option value="${i.idPoll}">${i.question}</option>
            `
        }

    }
    $("#ddlActivatePoll").html(print)
}

$("#btnAddPoll").click(function () {
    let pollName = $("#tbPollName").val().trim()
    let answer = $("#taAnswers").val().trim()
    let answers = answer.split(";")

    let errors = []

    let print = ""

    let regOverall = /^([\wŠĐŽĆČčćžšđ\d\.\?])+(\s[\wŠĐŽĆČčćžšđ\d\.\?]+)*$/

    if (!regOverall.test(pollName)) {
        errors.push("Question is not in good format.")
    }

    if (answer == "") {
        errors.push("Answer can't be blank.")
    } else {
        for (let i = 0; i < answers.length; i++) {
            if (!regOverall.test(answers[i].trim())) {
                errors.push("Answer is not in good format.")
            }
        }
    }

    if (errors.length) {
        for (let error of errors) {
            print += error + "<br/>"
        }
        $("#errorsPoll").html(print)
    } else {
        $("#errorsPoll").html("")
        $.ajax({
            url: "models/polls/insertPoll.php",
            method: "POST",
            dataType: "json",
            data: {
```

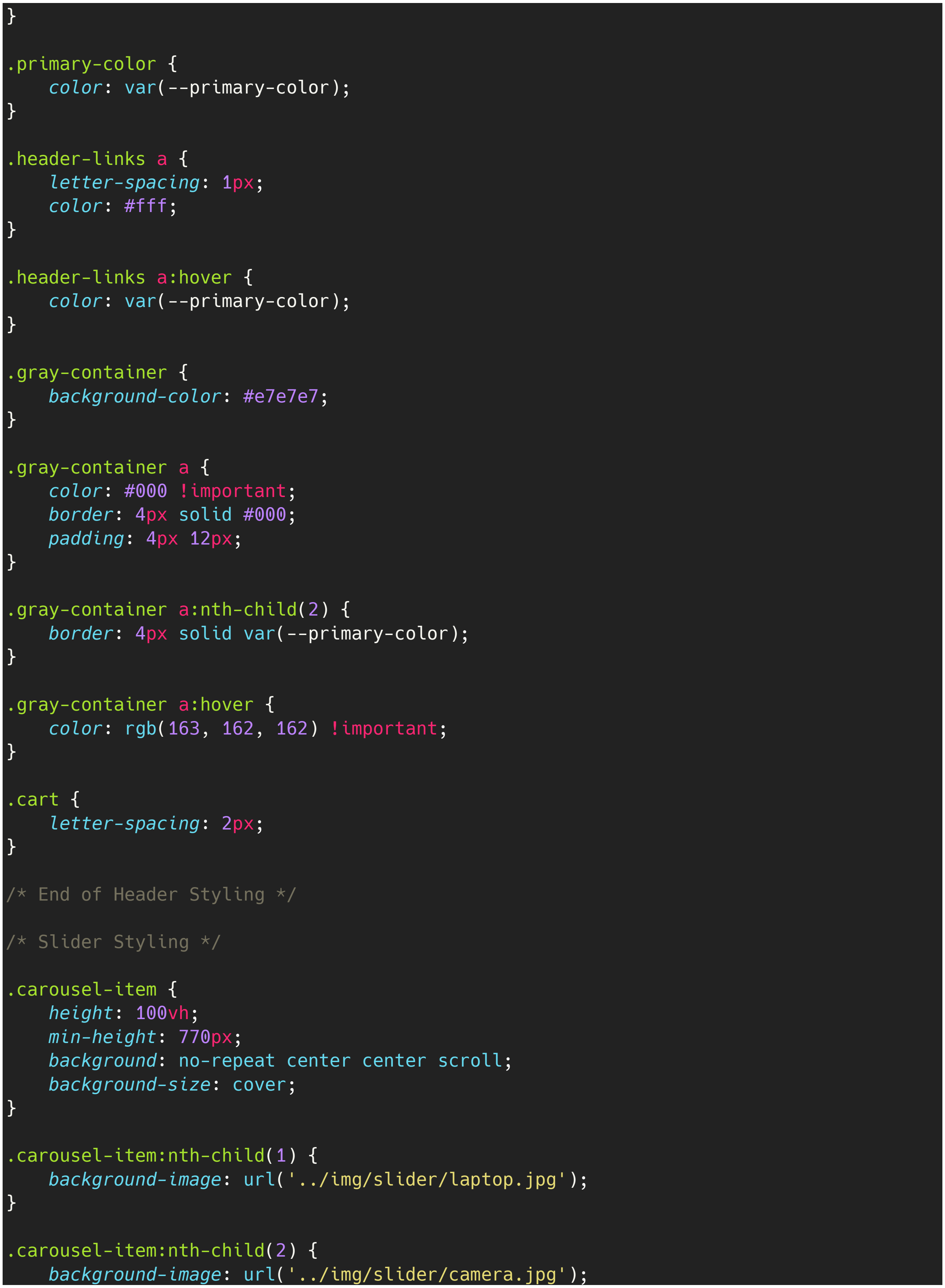


```
                name: pollName,
                answers: answers,
                send: true
            },
            success: function (data) {
                $("#errorsPoll").html(data.message)
                getPolls()
            },
            error: function (xhr, status, error) {
                console.log(xhr.status)
                console.log(error)
            }
        })
    }
})

$("#btnActivatePoll").click(function () {
    let id = $("#ddlActivatePoll").val()

    if (id != 0) {
        $.ajax({
            url: "models/polls/activatePoll.php",
            method: "POST",
            data: {
                id: id,
                send: true
            },
            success: function(data) {
                $("#activatePoll").html(data.message)
                getPolls()
            },
            error: function(xhr, status, error) {
                console.log(xhr.status)
                console.log(error)
            }
        })
    }
})

$("#btnPollResult").click(function() {
    console.log("heheheh")
    let id = $("#ddlPollResult").val()

    if (id != 0) {
        $.ajax({
            url: "models/polls/showPoll.php",
            method: "POST",
            data: {
                id: id,
                send: true
            },
            success: function(data) {
                $("#pollResultAdmin").html(printPollResults(data))
            },
            error: function(xhr, status, error) {
                console.log(xhr.status)
                console.log(error)
            }
        })
    }
})
```

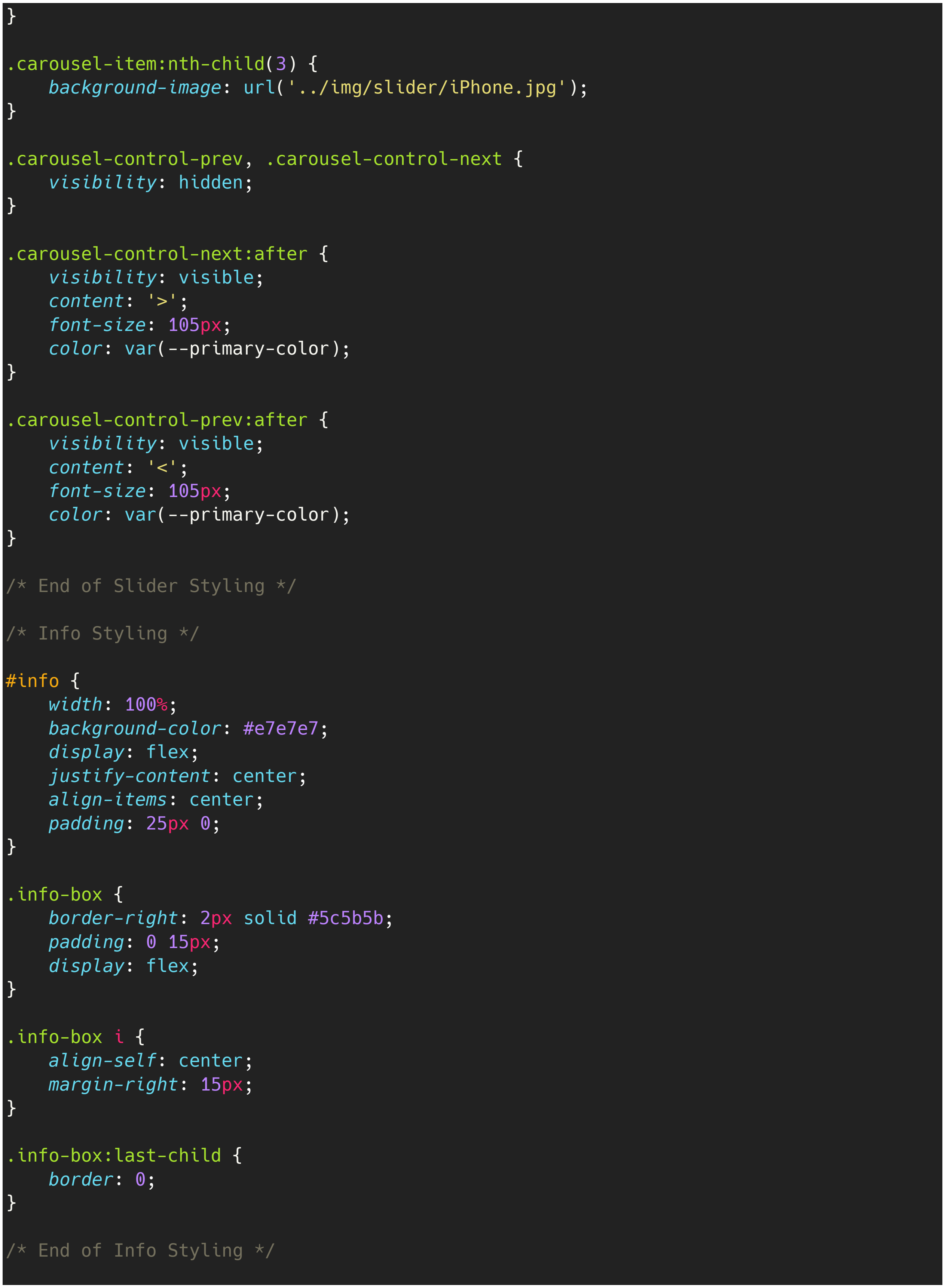
3.5.2 css

3.5.2.1 style.css

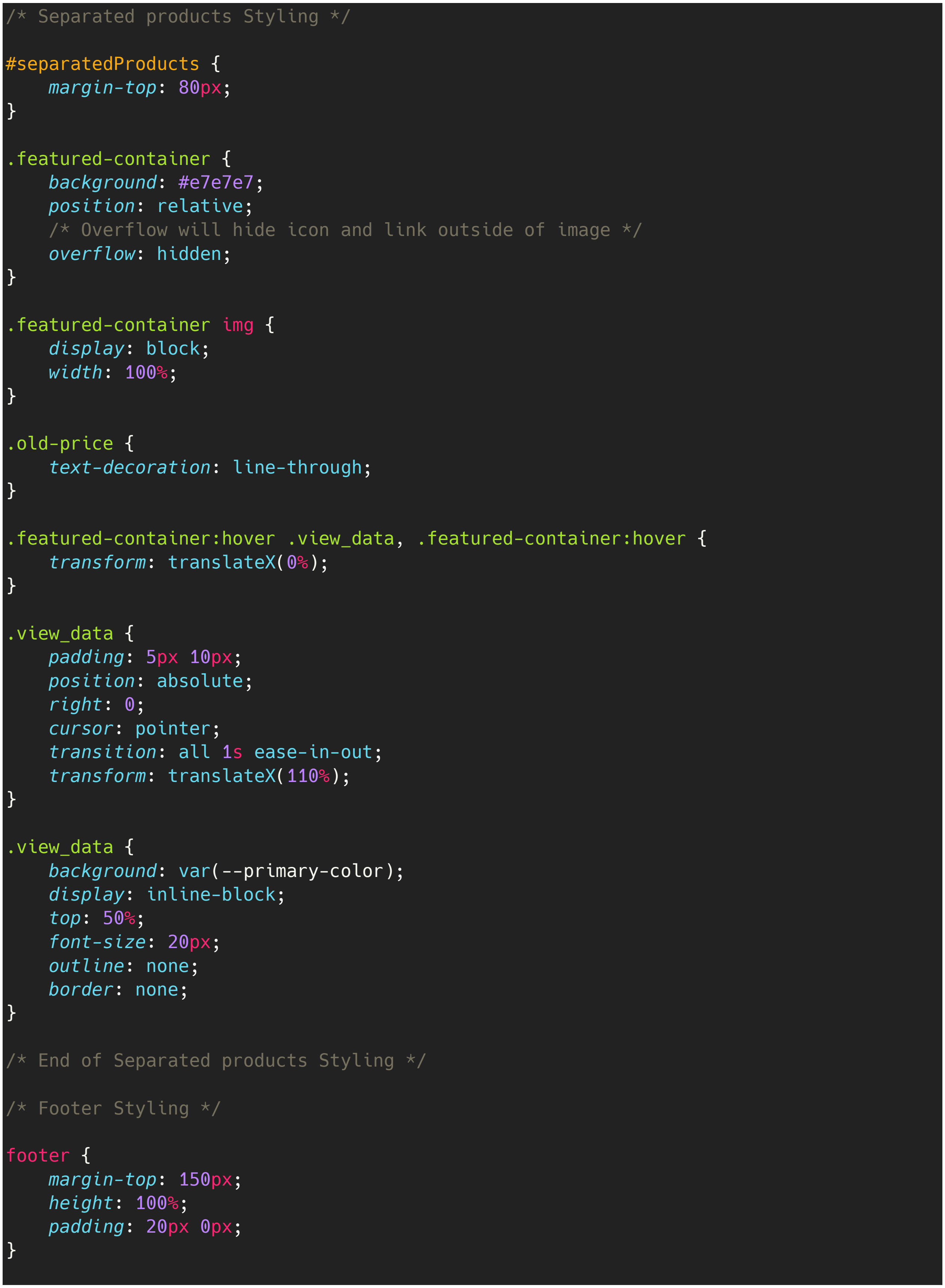
```
/* Import font and basic settings */

@import url('https://fonts.googleapis.com/css?family=Lato&display=swap');
:root {
    --primary-color: #f9ce09;
}

body, html {
    font-family: "Lato", sans-serif;
}

a {
    text-decoration: none !important;
}

ul {
    list-style: none;
}

.underTitle {
    width: 15%;
    height: 2px;
    background-color: var(--primary-color);
}

/* Header Styling */

.navbar-toggler {
    background-color: #fff !important;
}

.bg-darkest {
    background-color: #000;
}

.bg-darkest a {
    color: #fff !important;
}

.bg-darkest a:hover {
    color: rgb(163, 162, 162) !important;
```



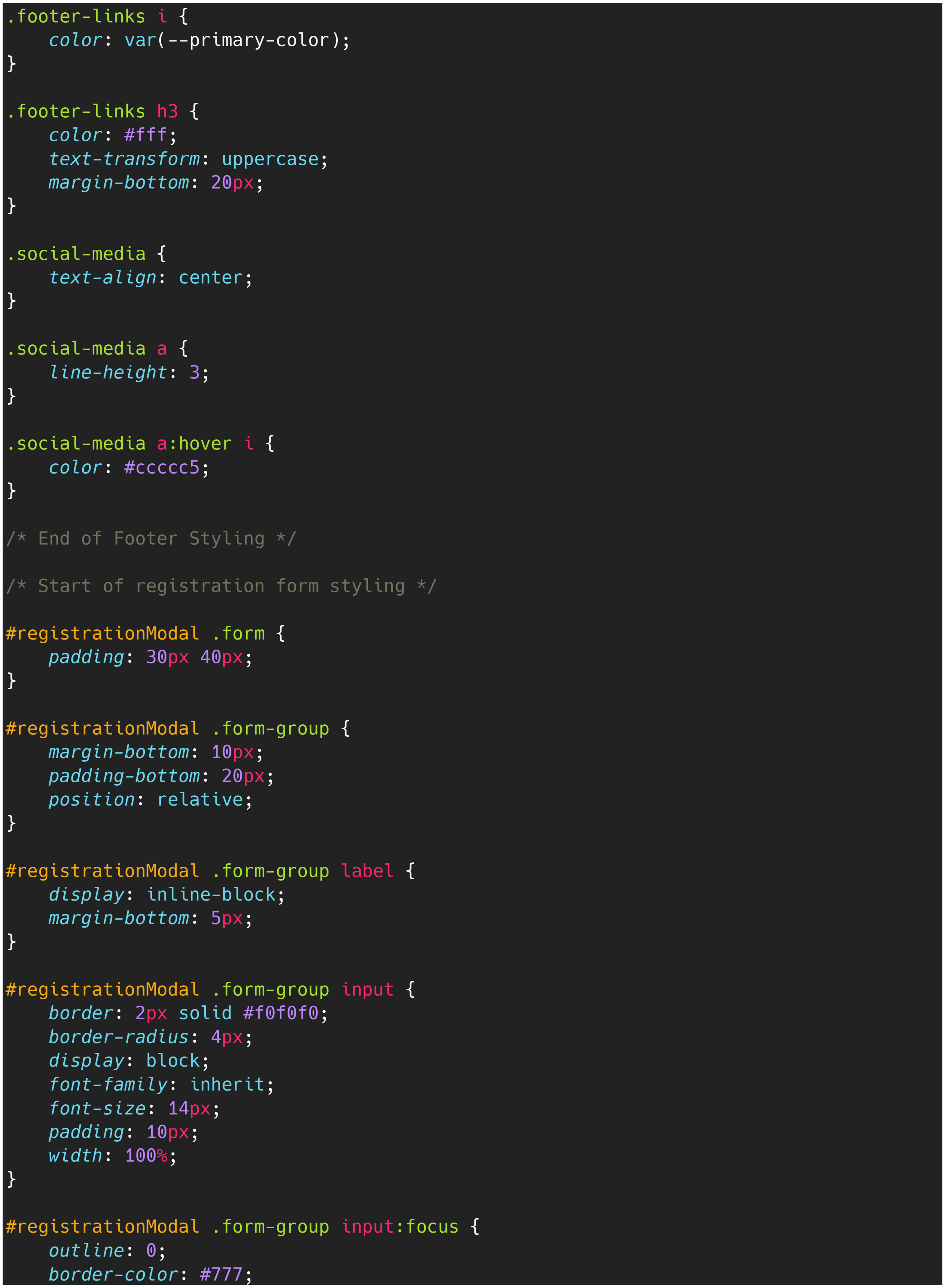
```css
}

.primary-color {
    color: var(--primary-color);
}

.header-links a {
    letter-spacing: 1px;
    color: #fff;
}

.header-links a:hover {
    color: var(--primary-color);
}

.gray-container {
    background-color: #e7e7e7;
}

.gray-container a {
    color: #000 !important;
    border: 4px solid #000;
    padding: 4px 12px;
}

.gray-container a:nth-child(2) {
    border: 4px solid var(--primary-color);
}

.gray-container a:hover {
    color: rgb(163, 162, 162) !important;
}

.cart {
    letter-spacing: 2px;
}

/* End of Header Styling */

/* Slider Styling */

.carousel-item {
    height: 100vh;
    min-height: 770px;
    background: no-repeat center center scroll;
    background-size: cover;
}

.carousel-item:nth-child(1) {
    background-image: url('../img/slider/laptop.jpg');
}

.carousel-item:nth-child(2) {
    background-image: url('../img/slider/camera.jpg');
```



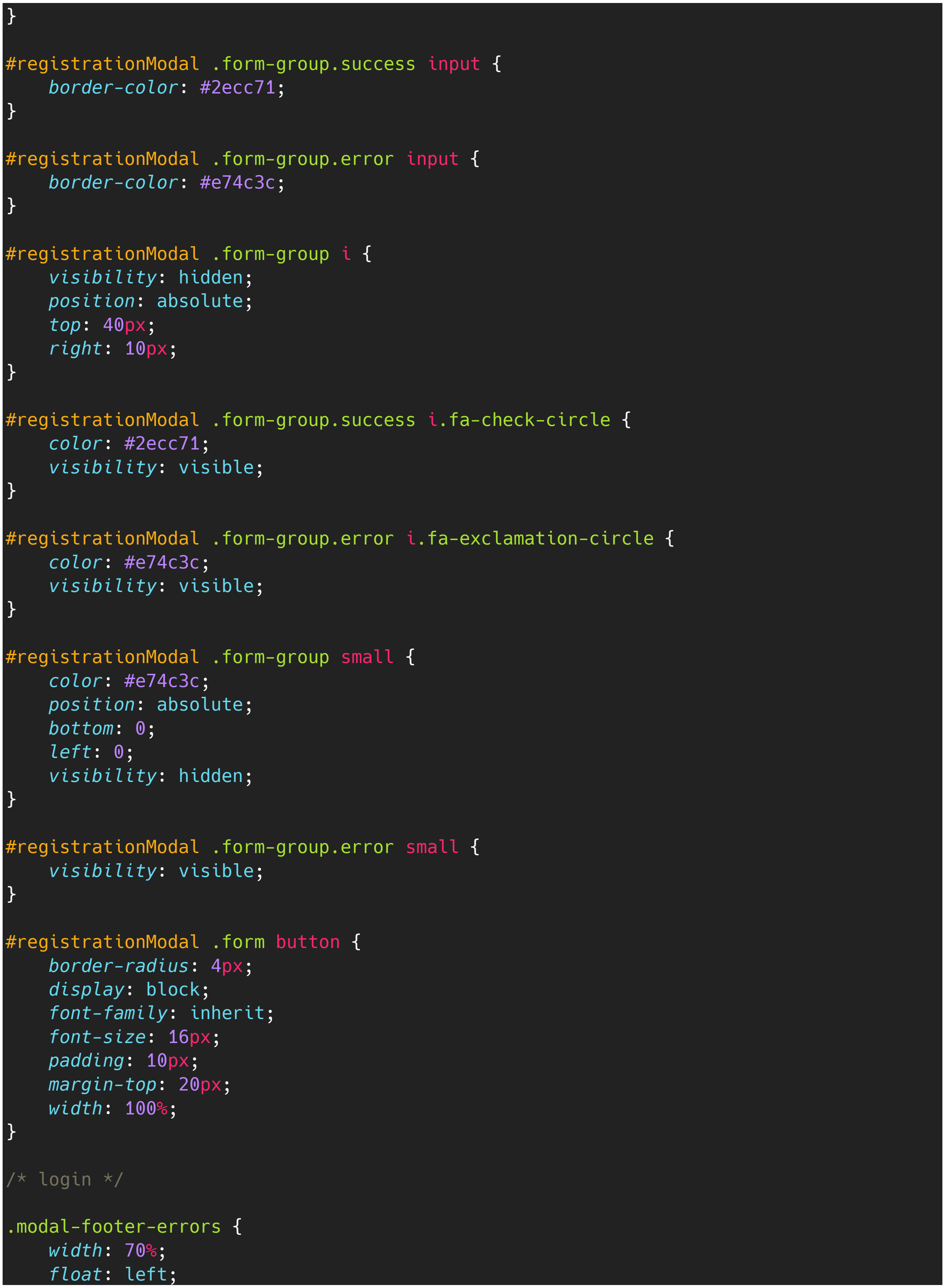
```css
}

.carousel-item:nth-child(3) {
    background-image: url('../img/slider/iPhone.jpg');
}

.carousel-control-prev, .carousel-control-next {
    visibility: hidden;
}

.carousel-control-next:after {
    visibility: visible;
    content: '>';
    font-size: 105px;
    color: var(--primary-color);
}

.carousel-control-prev:after {
    visibility: visible;
    content: '<';
    font-size: 105px;
    color: var(--primary-color);
}

/* End of Slider Styling */

/* Info Styling */

#info {
    width: 100%;
    background-color: #e7e7e7;
    display: flex;
    justify-content: center;
    align-items: center;
    padding: 25px 0;
}

.info-box {
    border-right: 2px solid #5c5b5b;
    padding: 0 15px;
    display: flex;
}

.info-box i {
    align-self: center;
    margin-right: 15px;
}

.info-box:last-child {
    border: 0;
}

/* End of Info Styling */
```



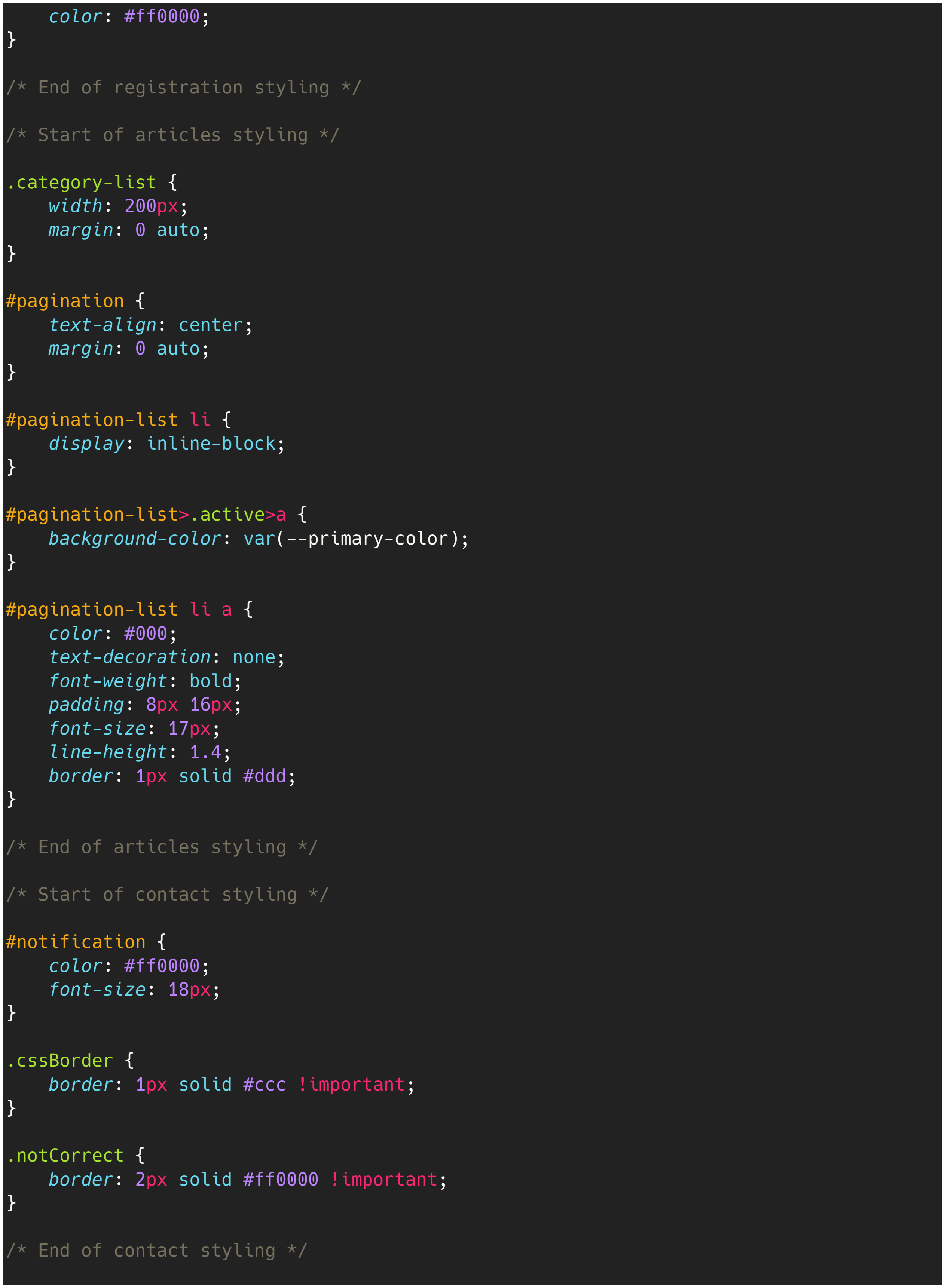
```css
/* Separated products Styling */

#separatedProducts {
    margin-top: 80px;
}

.featured-container {
    background: #e7e7e7;
    position: relative;
    /* Overflow will hide icon and link outside of image */
    overflow: hidden;
}

.featured-container img {
    display: block;
    width: 100%;
}

.old-price {
    text-decoration: line-through;
}

.featured-container:hover .view_data, .featured-container:hover {
    transform: translateX(0%);
}

.view_data {
    padding: 5px 10px;
    position: absolute;
    right: 0;
    cursor: pointer;
    transition: all 1s ease-in-out;
    transform: translateX(110%);
}

.view_data {
    background: var(--primary-color);
    display: inline-block;
    top: 50%;
    font-size: 20px;
    outline: none;
    border: none;
}

/* End of Separated products Styling */

/* Footer Styling */

footer {
    margin-top: 150px;
    height: 100%;
    padding: 20px 0px;
}
```



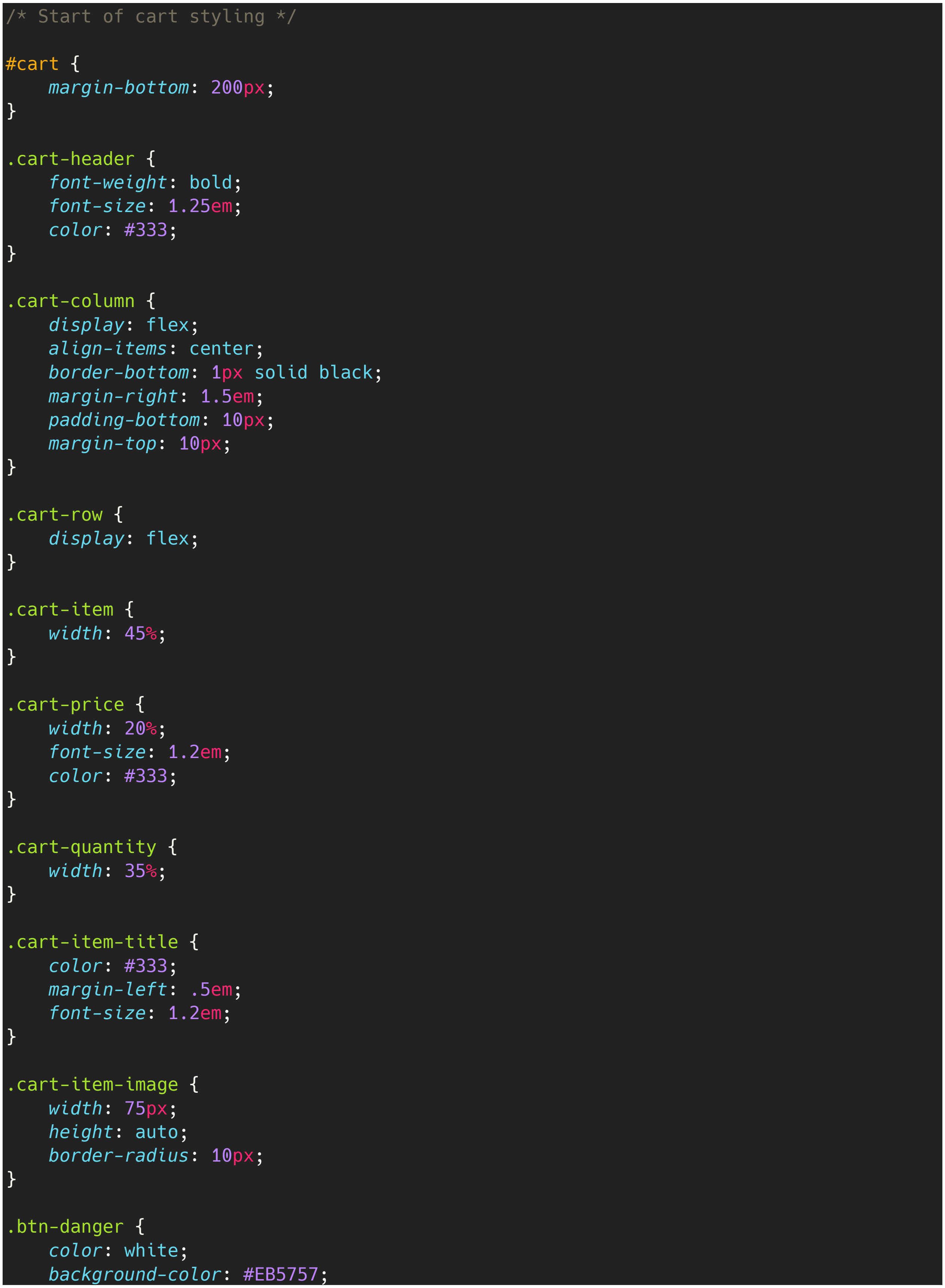
```css
.footer-links i {
    color: var(--primary-color);
}

.footer-links h3 {
    color: #fff;
    text-transform: uppercase;
    margin-bottom: 20px;
}

.social-media {
    text-align: center;
}

.social-media a {
    line-height: 3;
}

.social-media a:hover i {
    color: #ccccc5;
}

/* End of Footer Styling */

/* Start of registration form styling */

#registrationModal .form {
    padding: 30px 40px;
}

#registrationModal .form-group {
    margin-bottom: 10px;
    padding-bottom: 20px;
    position: relative;
}

#registrationModal .form-group label {
    display: inline-block;
    margin-bottom: 5px;
}

#registrationModal .form-group input {
    border: 2px solid #f0f0f0;
    border-radius: 4px;
    display: block;
    font-family: inherit;
    font-size: 14px;
    padding: 10px;
    width: 100%;
}

#registrationModal .form-group input:focus {
    outline: 0;
    border-color: #777;
```



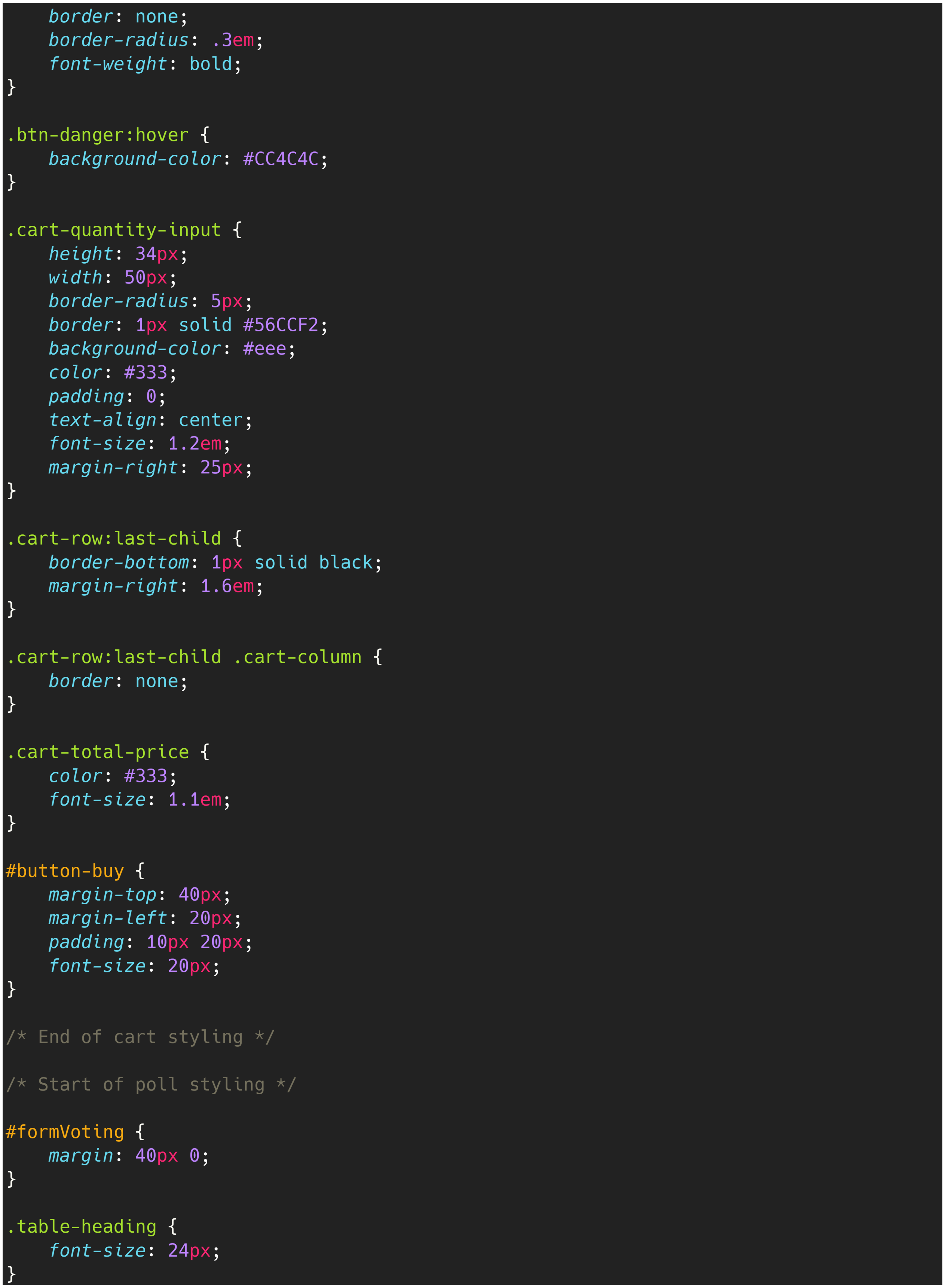
```
}

#registrationModal .form-group.success input {
    border-color: #2ecc71;
}

#registrationModal .form-group.error input {
    border-color: #e74c3c;
}

#registrationModal .form-group i {
    visibility: hidden;
    position: absolute;
    top: 40px;
    right: 10px;
}

#registrationModal .form-group.success i.fa-check-circle {
    color: #2ecc71;
    visibility: visible;
}

#registrationModal .form-group.error i.fa-exclamation-circle {
    color: #e74c3c;
    visibility: visible;
}

#registrationModal .form-group small {
    color: #e74c3c;
    position: absolute;
    bottom: 0;
    left: 0;
    visibility: hidden;
}

#registrationModal .form-group.error small {
    visibility: visible;
}

#registrationModal .form button {
    border-radius: 4px;
    display: block;
    font-family: inherit;
    font-size: 16px;
    padding: 10px;
    margin-top: 20px;
    width: 100%;
}

/* login */

.modal-footer-errors {
    width: 70%;
    float: left;
```



```css
    color: #ff0000;
}

/* End of registration styling */

/* Start of articles styling */

.category-list {
    width: 200px;
    margin: 0 auto;
}

#pagination {
    text-align: center;
    margin: 0 auto;
}

#pagination-list li {
    display: inline-block;
}

#pagination-list>.active>a {
    background-color: var(--primary-color);
}

#pagination-list li a {
    color: #000;
    text-decoration: none;
    font-weight: bold;
    padding: 8px 16px;
    font-size: 17px;
    line-height: 1.4;
    border: 1px solid #ddd;
}

/* End of articles styling */

/* Start of contact styling */

#notification {
    color: #ff0000;
    font-size: 18px;
}

.cssBorder {
    border: 1px solid #ccc !important;
}

.notCorrect {
    border: 2px solid #ff0000 !important;
}

/* End of contact styling */
```



```css
/* Start of cart styling */

#cart {
    margin-bottom: 200px;
}

.cart-header {
    font-weight: bold;
    font-size: 1.25em;
    color: #333;
}

.cart-column {
    display: flex;
    align-items: center;
    border-bottom: 1px solid black;
    margin-right: 1.5em;
    padding-bottom: 10px;
    margin-top: 10px;
}

.cart-row {
    display: flex;
}

.cart-item {
    width: 45%;
}

.cart-price {
    width: 20%;
    font-size: 1.2em;
    color: #333;
}

.cart-quantity {
    width: 35%;
}

.cart-item-title {
    color: #333;
    margin-left: .5em;
    font-size: 1.2em;
}

.cart-item-image {
    width: 75px;
    height: auto;
    border-radius: 10px;
}

.btn-danger {
    color: white;
    background-color: #EB5757;
```



```
    border: none;
    border-radius: .3em;
    font-weight: bold;
}

.btn-danger:hover {
    background-color: #CC4C4C;
}

.cart-quantity-input {
    height: 34px;
    width: 50px;
    border-radius: 5px;
    border: 1px solid #56CCF2;
    background-color: #eee;
    color: #333;
    padding: 0;
    text-align: center;
    font-size: 1.2em;
    margin-right: 25px;
}

.cart-row:last-child {
    border-bottom: 1px solid black;
    margin-right: 1.6em;
}

.cart-row:last-child .cart-column {
    border: none;
}

.cart-total-price {
    color: #333;
    font-size: 1.1em;
}

#button-buy {
    margin-top: 40px;
    margin-left: 20px;
    padding: 10px 20px;
    font-size: 20px;
}

/* End of cart styling */

/* Start of poll styling */

#formVoting {
    margin: 40px 0;
}

.table-heading {
    font-size: 24px;
}
```

```
/* End of poll styling */
```