

# DOKUMENTACIJA

<http://mobishoptpo.epizy.com/>

Praktikum iz Web Programiranja PHP

Bojan Maksimović 92/19

Smer: Internet tehnologije

Modul: Web programiranje

Predmet: Praktikum iz Web Programiranja PHP

# SADRŽAJ

## Contents

1.	Uvod .....	5
1.1	Korišćeni programski jezici .....	5
1.2	Opis funkcionalnosti .....	5
1.3	Template .....	5
2.	Organizacija .....	6
2.1	Dizajn baze podataka .....	6
2.2	Mapa sajta .....	7
2.3	Struktura projekta.....	8
	root.....	8
	assets .....	8
	config .....	8
	data.....	8
	models.....	9
	models/admin.....	9
	views .....	10
	views/pages/admin.....	10
2.4	Slike stranica i opis funkcionalnosti .....	11
	Na svim stranicama.....	11
	views/pages/home.php .....	12
	views/pages/phones.php.....	13
	views/pages/details.php .....	14
	views/pages/contact.php .....	15
	views/pages/author.php.....	16
	views/pages/cart.php .....	17
	views/pages/login.php.....	18
	views/pages/register.php .....	19
	views/pages/admin.php .....	20
3.	Kodovi .....	25
3.1	PHP .....	25
	index.php.....	25
	views/fixed/header.php .....	26
	views/fixed/head.php .....	28
	views/fixed/footer.php .....	29
	views/fixed/scripts.php .....	30
	views/pages/home.php .....	31
	views/pages/phones.php.....	33
	views/pages/details.php .....	35
	views/pages/contact.php .....	37
	views/pages/author.php.....	39
	views/pages/cart.php .....	40

views/pages/login.php.....	42
views/pages/register.php .....	43
views/pages/admin.php .....	45
views/pages/admin/page-stats.php .....	47
views/pages/admin/logins.php .....	49
views/pages/admin/users.php.....	51
views/pages/admin/phones.php.....	53
views/pages/admin/brands.php .....	55
views/pages/admin/operating-systems.php .....	56
views/pages/admin/orders.php .....	57
views/pages/admin/messages.php .....	59
views/pages/admin/users/add-user-form.php .....	60
views/pages/admin/users/edit-user-form.php.....	62
views/pages/admin/phones/add-phone-form.php.....	64
views/pages/admin/phones/edit-phone-form.php .....	66
views/pages/admin/os/add-os-form.php.....	68
views/pages/admin/os/edit-os-form.php .....	69
views/pages/admin/brands/add-brand-form.php .....	70
views/pages/admin/brands/edit-brand-form.php.....	71
views/pages/admin/orders/edit-order-form.php .....	72
models/nav/functions.php .....	73
models/slider/functions.php .....	74
models/footer/functions.php .....	75
models/users/functions.php.....	76
models/users/get-current-user.php .....	78
models/users/login.php .....	79
models/users/register.php.....	81
models/users/logout.php .....	83
models/phones/functions.php.....	84
models/phones/get-phones.php .....	85
models/contact/functions.php .....	87
models/contact/send-message.php .....	88
models/author/functions.php.....	90
models/author/export-to-word.php.....	91
models/cart/functions.php .....	92
models/cart/add-to-cart.php .....	95
models/cart/change-quantity.php .....	96
models/cart/create-order.php.....	97
models/cart/get-cart-quantity.php .....	99
models/cart/get-cart.php.....	100
models/cart/remove-item.php .....	101
models/forms/functions.php.....	102
models/admin/functions.php .....	103
models/admin/delete.php .....	110
models/admin/users/add-user.php.....	112

models/admin/users/edit-user.php.....	114
models/admin/users/export-to-excel.php .....	117
models/admin/phones/add-phone.php.....	119
models/admin/phones/edit-phone.php .....	122
models/admin/os/add-os.php.....	126
models/admin/os/edit-os.php .....	127
models/admin/orders/edit-order.php .....	129
models/admin/brands/add-brand.php .....	131
models/admin/brands/edit-brand.php.....	132
config/.env .....	134
config/config.php .....	135
config/connection.php.....	136
3.2 CSS .....	138
style.css .....	138
3.3 JavaScript/jQuery.....	146
main.js.....	146
form-validation.js.....	148
phones.js .....	149
contact.js .....	153
cart.js.....	154
login.js .....	156
register.js.....	157
admin.js.....	158

# 1. Uvod

## 1.1 Korišćeni programski jezici

Za izradu sajta korišćeni su HTML, SASS/CSS, JavaScript/jQuery, PHP i XML(sitemap). Radno okruženje Visual Studio Code.

## 1.2 Opis funkcionalnosti

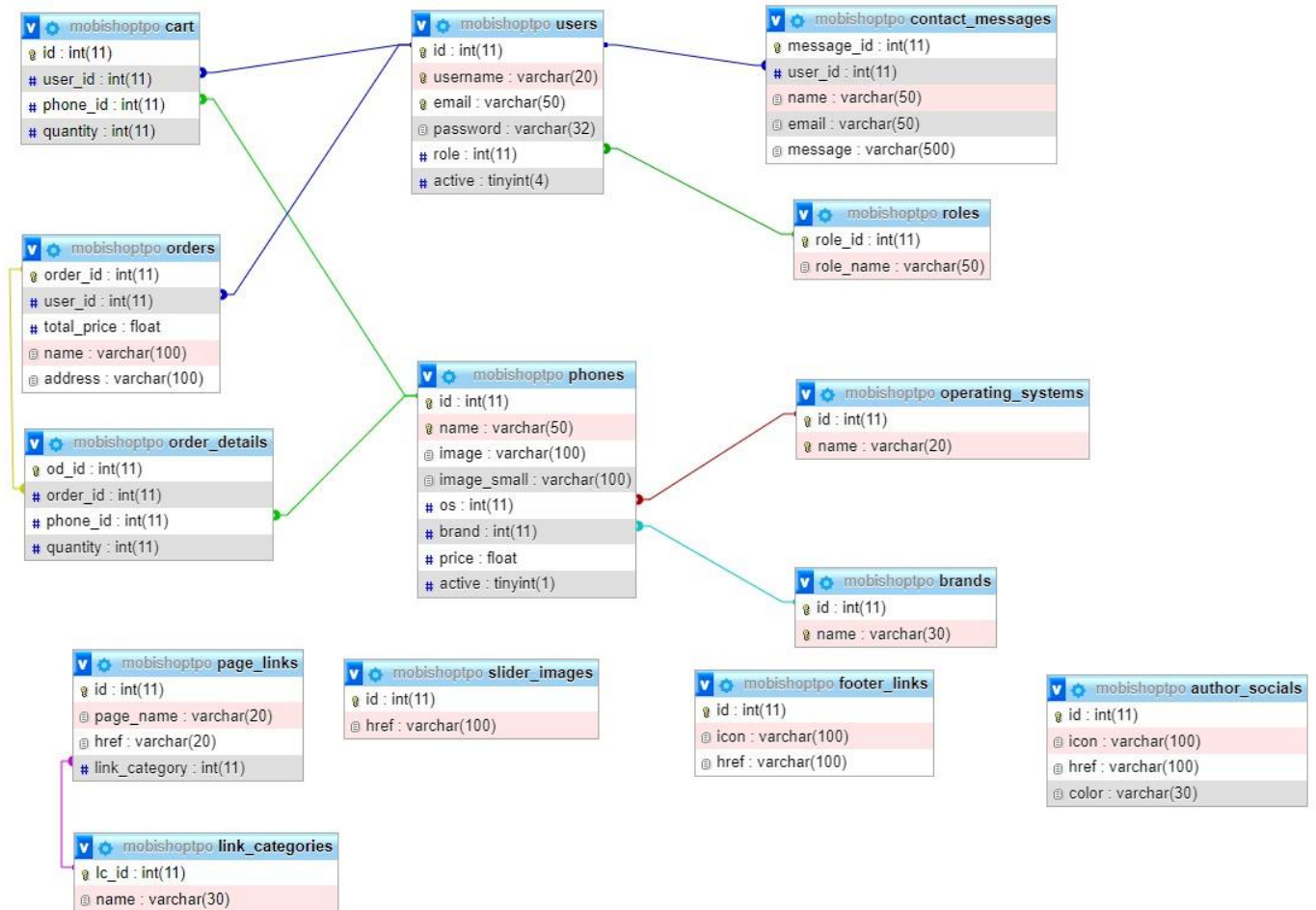
- Dinamičko ispisivanje menija iz baze podataka
- Dinamičko ispisivanje telefona iz baze podataka
- Pretraga I filtriranje telefona sa paginacijom
- Kreiranje naloga I logovanje
- Zaključavanje naloga i slanje mejla sa obaveštenjem
- Kontaktiranje administratora preko kontakt forme
- Sve forme su proverene I na klijentu I na serveru
- Dodavanje proizvoda u korpu
- Kreiranje porudžbine
- CRUD operacije nad korisnicima, telefonima, brendovima, operativnim sistemima, porudžbina I kontakt porukama
- Resize uploadovane slike pri dodavanju ili menjanju telefona
- Beleženje pristupa stranicama u log.txt fajl
- Statistika pristupa stranicama (u %) i prikaz koliko je bilo pristupa svakoj stranici u poslednjih 24h
- Beleženje grešaka u error\_log.txt
- Beleženje pokušaja logovanja u login\_log.txt
- Prikaz broja korisnika koji su se ulogovali tekućeg dana
- Export informacija o autoru sa autor stranice u Word fajl
- Export tabele sa korisnicima iz admin panela u Excel fajl

## 1.3 Template

Za izradu sajta nije korišćen template.

## 2. Organizacija

### 2.1 Dizajn baze podataka

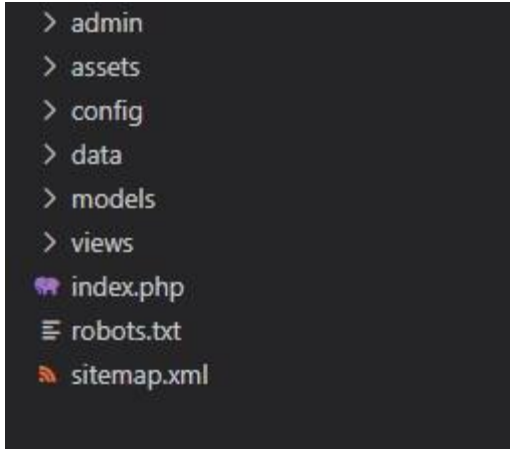


## 2.2 Mapa sajta

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=phones</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.7</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=contact</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.6</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=author</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.6</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=cart</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.5</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=login</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.4</priority>
  </url>
  <url>
    <loc>http://mobishoptpo.epizy.com/index.php?p=register</loc>
    <lastmod>2021-06-09</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.4</priority>
  </url>
</urlset>
```

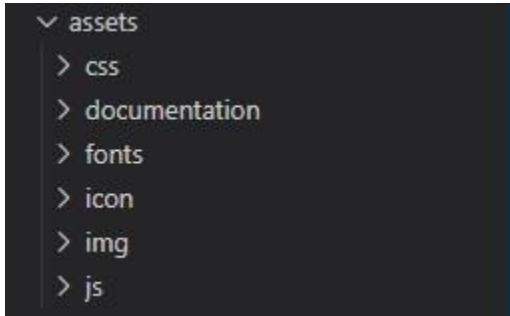
## 2.3 Struktura projektu

### root



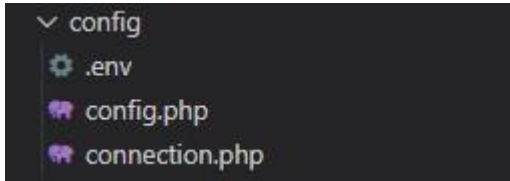
```
> admin
> assets
> config
> data
> models
> views
🐘 index.php
📄 robots.txt
📄 sitemap.xml
```

### assets



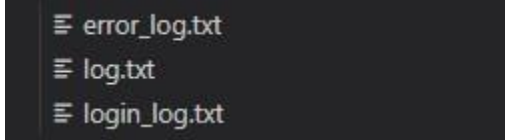
```
▼ assets
  > css
  > documentation
  > fonts
  > icon
  > img
  > js
```

### config



```
▼ config
  ⚙ .env
  🐘 config.php
  🐘 connection.php
```

### data



```
📄 error_log.txt
📄 log.txt
📄 login_log.txt
```



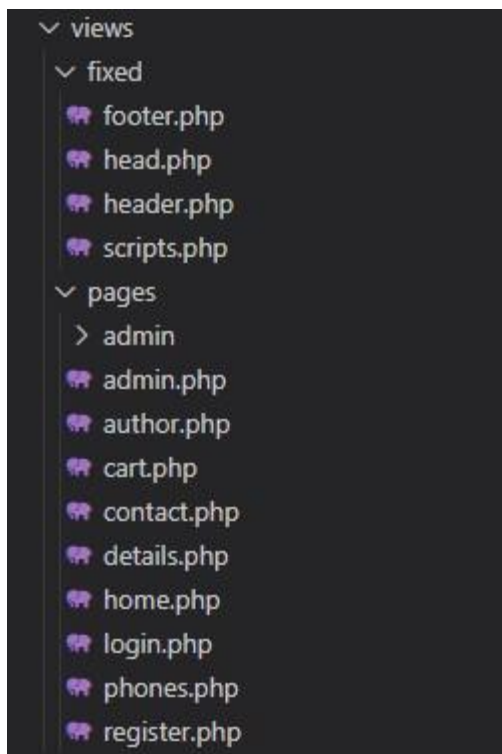
## models

- ▼ models
  - > admin
  - > author
  - > cart
  - > contact
  - > footer
  - > forms
  - > nav
  - > phones
  - > slider
  - > users

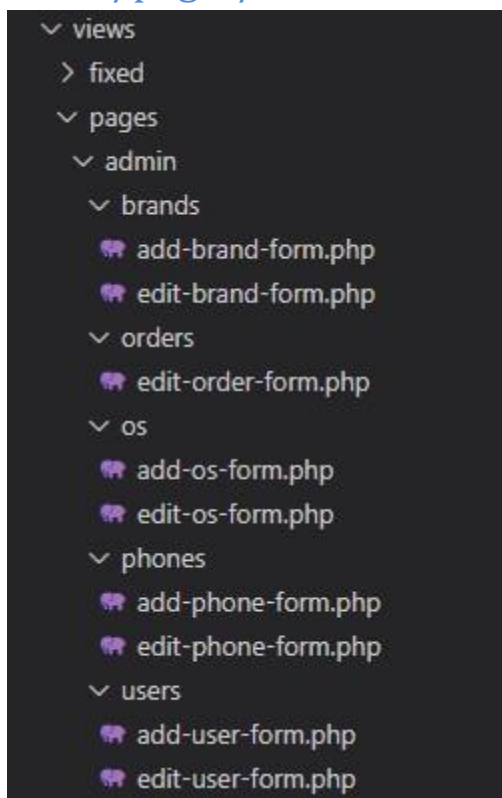
## models/admin

- ▼ models
  - ▼ admin
    - ▼ brands
      - 🐘 add-brand.php
      - 🐘 edit-brand.php
    - ▼ orders
      - 🐘 edit-order.php
    - ▼ os
      - 🐘 add-os.php
      - 🐘 edit-os.php
    - ▼ phones
      - 🐘 add-phone.php
      - 🐘 edit-phone.php
    - ▼ users
      - 🐘 add-user.php
      - 🐘 edit-user.php
      - 🐘 export-to-excel.php
    - 🐘 delete.php
    - 🐘 functions.php

## views

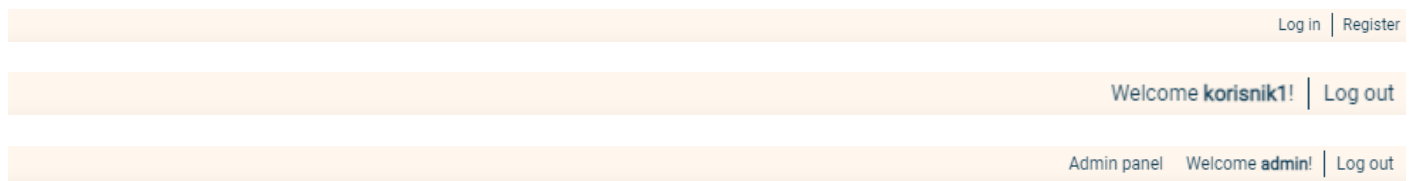


## views/pages/admin



## 2.4 Slike stranica i opis funkcionalnosti

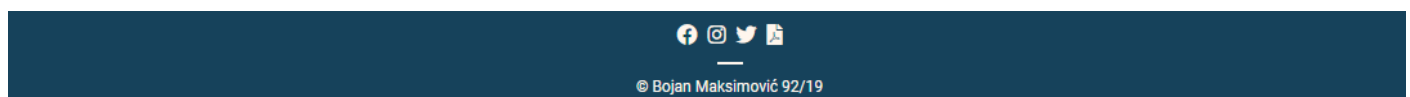
### Na svim stranicama



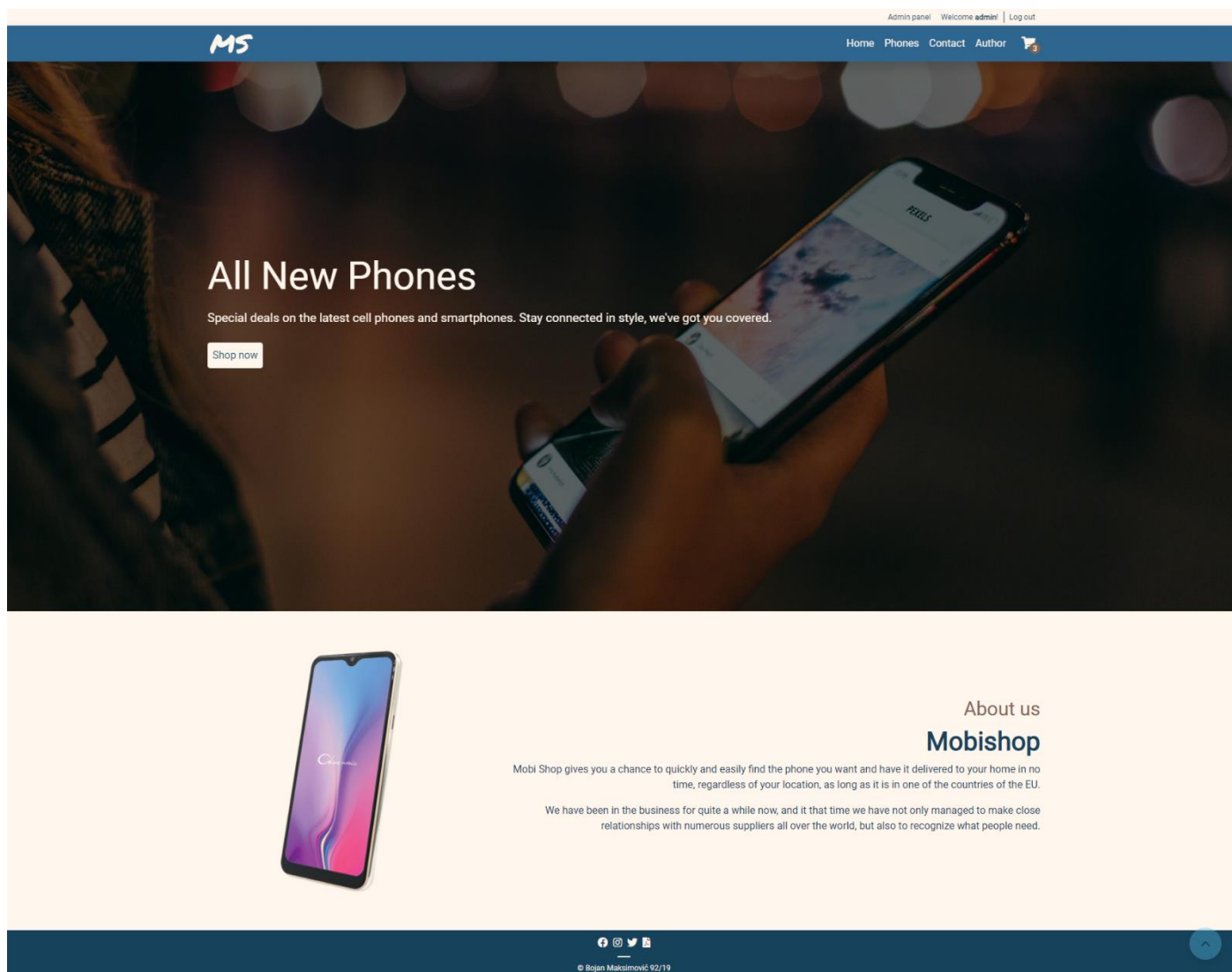
Na vrhu svake stranice nalazi se traka za logovanje/registraciju. Ako korisnik nije ulogovan, na traci se nalaze linkovi za logovanje i registraciju. U koliko je korisnik ulogovan, na traci se ispisuje Poruka 'Welcome [username]!' i link koji omogućava korisniku da se izloguje. Dodatno, ako trenutni korisnik ima privilegije administratora, na traci će se pojaviti link za pristup Admin panelu.



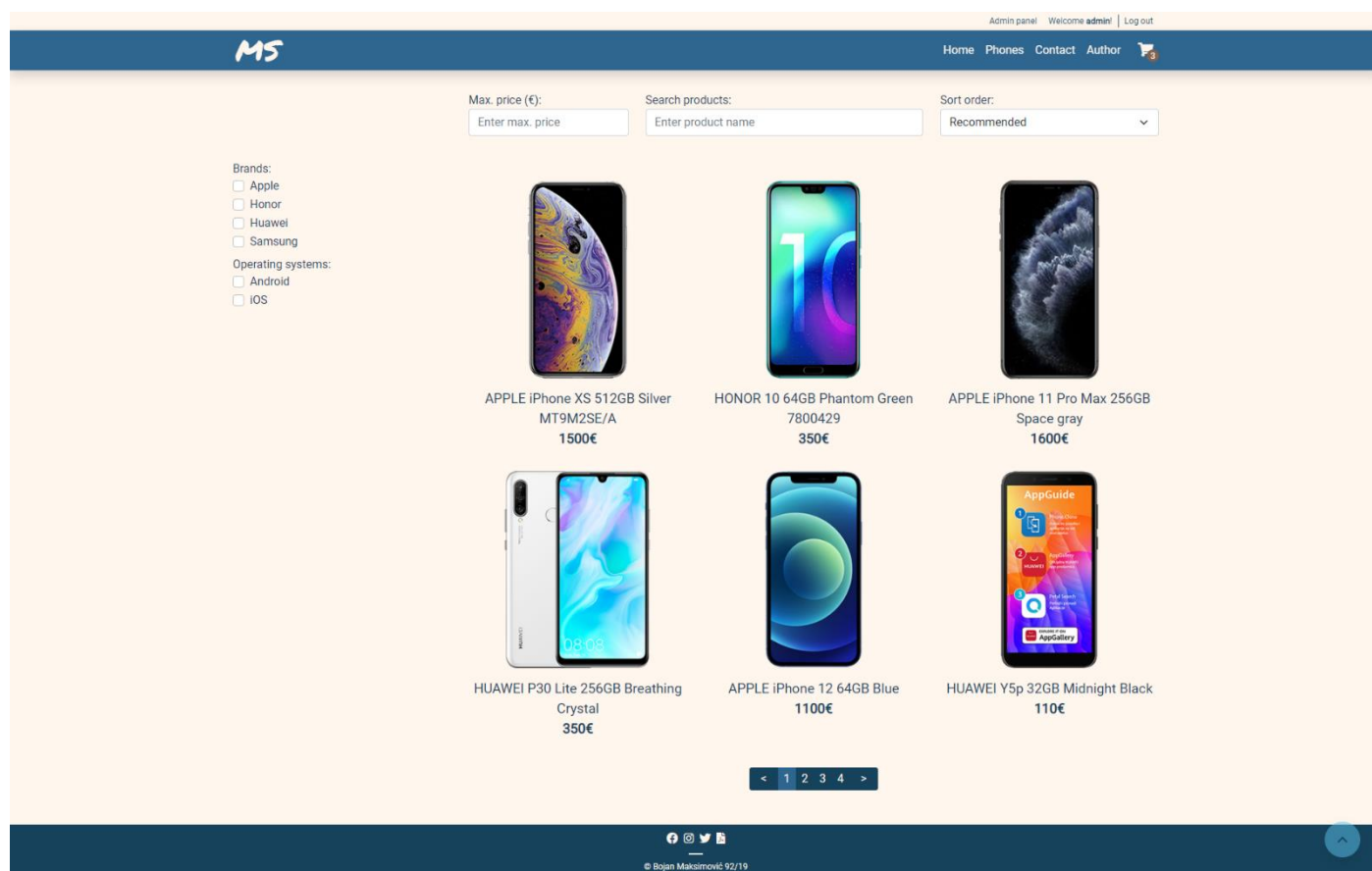
Ispod trake za logovanje nalazi se navigacioni meni (meni na velikim i meni na malim rezolucijama) dinamički ispisan iz tabele u bazi podataka. U njemu se takođe nalazi i link za pristup korpi (stranica cart.php) u čijem uglu piše koliko se proizvoda nalazi u korpi trenutno ulogovanog korisnika.



Na dnu svake stranice nalazi se footer u kom se nalaze linkovi do socijalnih mreža i dokumentacije koji su takođe učitani iz tabele u bazi podataka.

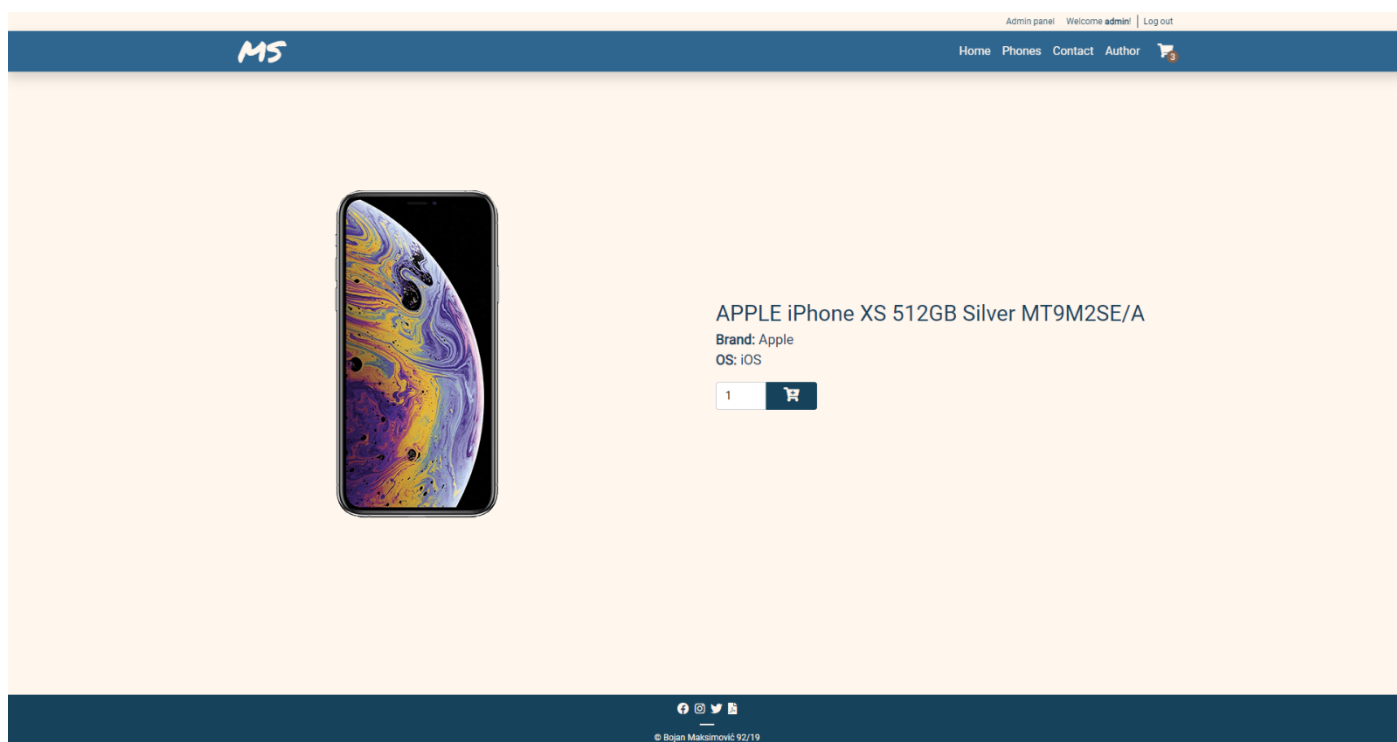


Na početnoj stranici se nalazi slider u kom su slike dinamički učitane iz tabele u bazi podataka.



Na stranici 'Phones' nalaze se elementi za filtriranje I sortiranje prikazanih telefona. Checkbox-ovi u ovom div-u su dinamički učitani iz baze podataka iz tabela 'operating\_systems' I 'brands'. Svi prikazani uređaji su takođe učitani iz baze podataka na osnovu izabranih filtera i prikazuje se slika manjih dimenzija. Na ovoj stranici postoji straničenje.

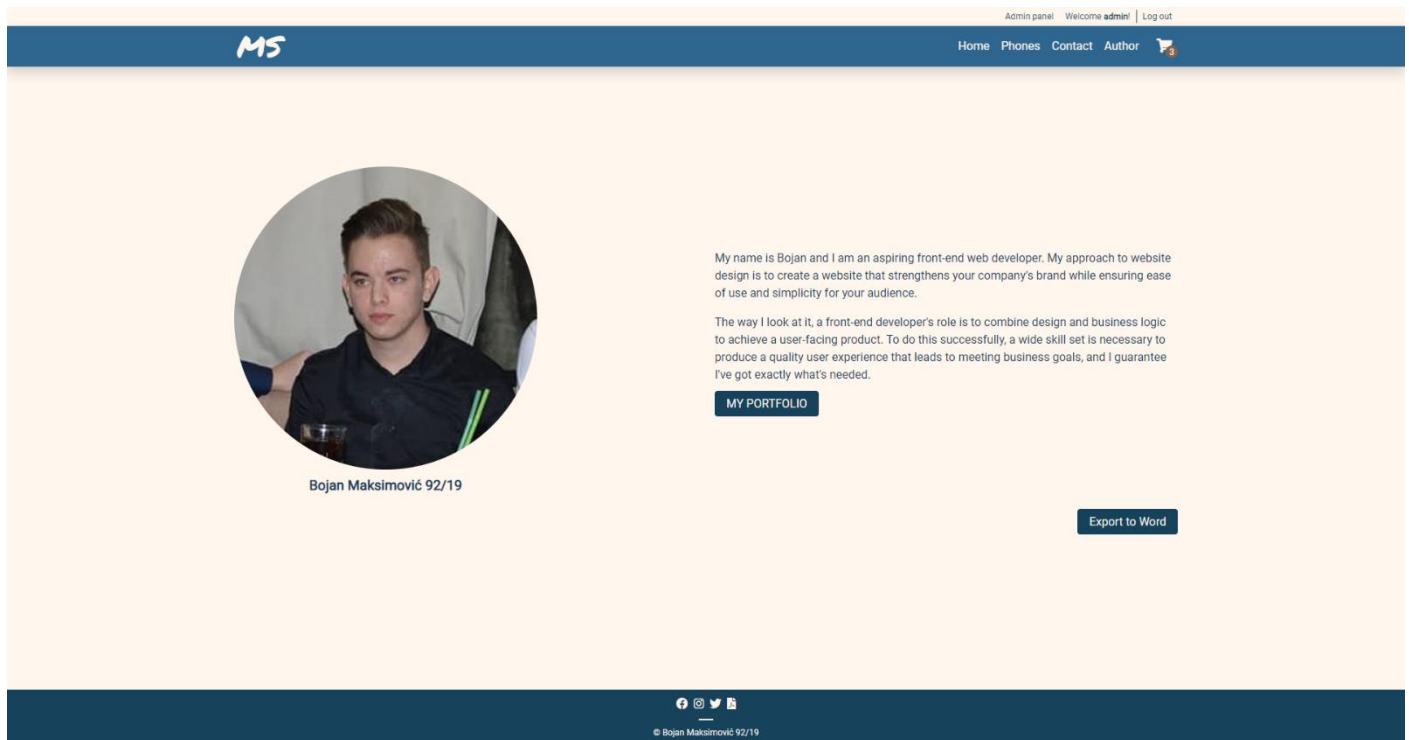
Hoverovanjem cursorom preko nekog uređaja pojavljuju se tasteri za dodavanje u korpu I taster koji otvara stranicu sa više detalja o uređaju. Ako je korisnik ulogovan klikom na taster za dodavanje u korpu uređaj se upisuje u tabelu 'cart' ili ako već postoji povećava mu se količina, a ako korisnik nije ulogovan preusmeren je na stranicu za logovanje.



Na ovoj stranici se prikazuje slika telefona originalnih dimenzija, naziv telefona i neke karakteristike. Sa ove stranice takođe može telefon da se doda u korpu uz mogućnost da korisnik unese količinu.

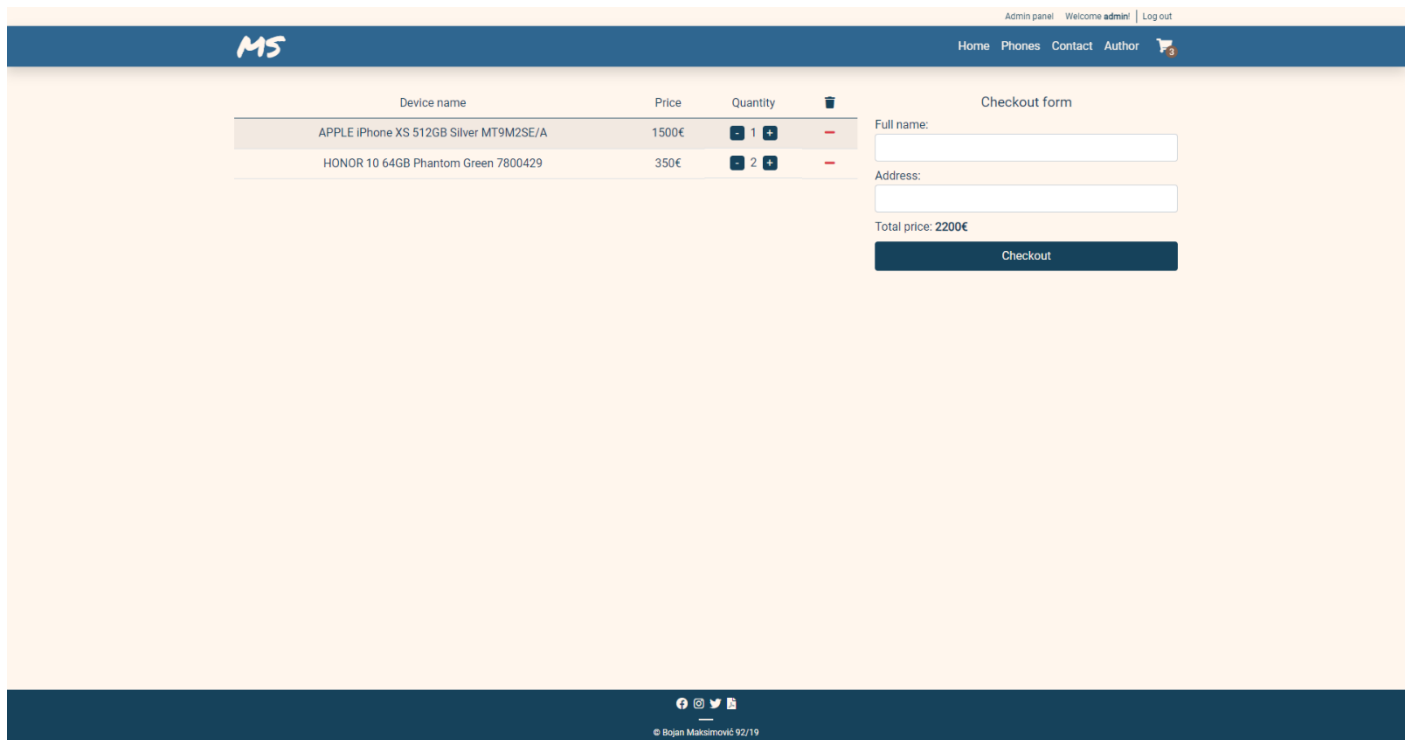
The screenshot shows a web page with a dark blue header and footer. The header contains the 'MS' logo, navigation links (Home, Phones, Contact, Author), and user information (Admin panel, Welcome admin!, Log out). The main content area is light orange and features the title 'Contact us' with a horizontal line underneath. Below the title, there is a paragraph: 'If you wish to reach out to us, you can do so via social media, the information listed below, or by filling out the contact form.' To the left of this paragraph are two lines of contact information: a phone number '+381 65/518-3225' and an email address 'bojan.maksimovic.92.19@ict.edu.rs'. To the right is a 'Contact form' with three input fields: 'Email:', 'Name:', and 'Message:'. The 'Message:' field is a larger text area. Below the form is a dark blue button with a white paper plane icon. The footer contains social media icons (Facebook, Instagram, Twitter, YouTube) and a copyright notice: '© Bojan Maksimović 92/19'.

Na stranici se nalazi forma za kontaktiranje administratora. Unos u formu se i sa klijentske i sa serverske strane proverava regularnim izrazima. Ako je sve pravilno uneto, uneti podaci se upisuju u tabelu 'contact\_messages' u bazi podataka. U slučaju da je korisnik bio ulogovan kad je poslao u poruku, u tabelu se takođe upisuje i ID tog korisnika, a u slučaju da nije, to polje dobija vrednost NULL.



Na stranici `author.php`, linkovi do autorovih socijalnih mreža se učitavaju iz tabele 'authorsocials' iz baze podataka. Ovde takođe postoji i mogućnost da se podaci o autoru preuzmu u obliku Word dokumenta klikom na taster 'Export to Word' (web server na kom je sajt hostovan radi na Linux-u a klasa 'COM' je ekskluzivna za Windows pa će klik na taster da vrati error kod 500 ali kod za export I download postoji u <models/author/export-to-word.php>).





Na stranici cart.php, u slučaju da korisnik proba da joj pristupi preko url adrese, proverava se da li je korisnik ulogovan. U slučaju da nije, korisnik se redirektuje na Login stranicu koja mu govori da nema pristup korpi dok se ne uloguje.

Ako je korisnik ulogovan i uspešno je pristupio stranici, ispisuje mu se tabela sa svim uređajima koji se nalaze u korpi i div sa formom za kreiranje porudžbine. Uneti sadržaj u formi se proverava regularnim izrazima sa klijentske i serverske strane, i ako je forma ispravno popunjena, korpa se prazni i uneti podaci sa sve ID-em korisnika i ukupnom cenom se unose u tabelu 'orders', a svi naručeni telefoni sa njihovim cenama i količinama se unose u svoje zasebne redove u tabeli 'order\_details' i svi dobijaju kolonu u koju se automatski upisuje ID kreirane porudžbine.

Log in | Register

MS Home Phones Contact Author

Log in

Email:

Password:

Log in

Don't have an account?  
Register instead

© Bojan Makarović 92/19

Stranica sa formom za logovanje sa proverom sa klijentske I serverske strane. Ako korisnik unese ispravan email ali pogrešnu lozinku tri puta u 5 minuta, nalog se zaključava i korisniku se šalje mejl sa obaveštenjem (free hosting na infinityfree.com ne dozvoljava slanje mejla sa mail() funkcijom da bi umanjili spam ali kod postoji u [models/users/login.php](#) i [models/users/functions.php](#)). U slučaju da već ulogovani korisnik pokuša da pristupi ovoj stranici on se redirektuje na početnu stranicu.

The screenshot shows a web registration form titled "Register". At the top, there is a dark blue header with the "MS" logo on the left and navigation links "Home", "Phones", "Contact", "Author", and a shopping cart icon on the right. The main content area has a light orange background. The registration form is centered and includes the following elements: a title "Register", input fields for "Username:", "Email:", "Password:", and "Repeat password:", a dark blue "Register" button, and a link "Already have an account? Log in instead". The footer is a dark blue bar containing social media icons and the copyright notice "© Bojan Maksimović 92/19".

MS

Home Phones Contact Author

Log in | Register

### Register

Username:

Email:

Password:

Repeat password:

Register

Already have an account?  
Log in instead

© Bojan Maksimović 92/19

Stranica sa formom za registraciju sa proverom sa klijentske I serverske strane. Korisniku se ispisuje greška u slušaju da unese email ili username koji je već zauzet. U slučaju da već ulogovani korisnik pokuša da pristupi ovoj stranici on se redirektuje na početnu stranicu.

## Page stats

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

Page	%	24h
home	11.56%	64
cart	4.63%	15
login	7.43%	52
phones	4.7%	37
contact	2.16%	7
author	3.37%	43
details	1.08%	6
admin	64.95%	435
register	0.13%	2

© Bojan Maksimović 92/19

## Logins (24h)

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

ID	Username	Email
5	admin	admin@gmail.com
1	korisnik1	korisnik1@gmail.com
4	boki200075	bojan.maxim075@gmail.com

Unique users: 3

© Bojan Maksimović 92/19

## Users

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

ID	Username	Email	Role	Active	Actions
1	korisnik1	korisnik1@gmail.com	regular	1	<a href="#">Edit</a> <a href="#">Delete</a>
2	korisnik2	korisnik2@gmail.com	regular	1	<a href="#">Edit</a> <a href="#">Delete</a>
4	boki200075	bojan.maxim075@gmail.com	regular	1	<a href="#">Edit</a> <a href="#">Delete</a>
5	admin	admin@gmail.com	admin	1	
6	korisnik3	korisnik3@gmail.com	admin	1	<a href="#">Edit</a> <a href="#">Delete</a>
7	korisnik4	korisnik4@gmail.com	regular	1	<a href="#">Edit</a> <a href="#">Delete</a>

[Add new user](#) [Export to Excel](#)

[Facebook](#) [Instagram](#) [Twitter](#) [LinkedIn](#)

© Bojan Maksimović 92/19

## Phones

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

Name	OS	Brand	Price	Active	Actions
APPLE iPhone 11 Pro Max 256GB Gold	iOS	Apple	1600€	1	<a href="#">Edit</a> <a href="#">Delete</a>
APPLE iPhone 11 Pro Max 256GB Space gray	iOS	Apple	1600€	1	<a href="#">Edit</a> <a href="#">Delete</a>
APPLE iPhone 11 Pro Max 64GB Green MWHH2SE/A	iOS	Apple	1350€	1	<a href="#">Edit</a> <a href="#">Delete</a>
APPLE iPhone 12 64GB Blue	iOS	Apple	1100€	1	<a href="#">Edit</a> <a href="#">Delete</a>
APPLE iPhone SE 128GB Red	iOS	Apple	650€	1	<a href="#">Edit</a> <a href="#">Delete</a>
APPLE iPhone XS 512GB Silver MT9M2SE/A	iOS	Apple	1500€	1	<a href="#">Edit</a> <a href="#">Delete</a>
HONOR 10 64GB Phantom Green 7800429	Android	Honor	350€	1	<a href="#">Edit</a> <a href="#">Delete</a>
HONOR 20 128GB Midnight black 51093VCM	Android	Honor	400€	1	<a href="#">Edit</a> <a href="#">Delete</a>
HONOR 7S 16GB Black 51094ABU	Android	Honor	90€	1	<a href="#">Edit</a> <a href="#">Delete</a>
HONOR 8A 32GB Black 51093WGU	Android	Honor	140€	1	<a href="#">Edit</a> <a href="#">Delete</a>
HONOR 9A 64GB Black	Android	Honor	170€	1	<a href="#">Edit</a> <a href="#">Delete</a>

[Add new phone](#)

[Facebook](#) [Instagram](#) [Twitter](#) [LinkedIn](#)

© Bojan Maksimović 92/19

## Brands

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats

Logins (24h)

Users

Phones

Brands

Operating systems

Orders

Messages

ID	Name	Actions
1	Apple	<a href="#">Edit</a> <a href="#">Delete</a>
2	Honor	<a href="#">Edit</a> <a href="#">Delete</a>
3	Huawei	<a href="#">Edit</a> <a href="#">Delete</a>
4	Samsung	<a href="#">Edit</a> <a href="#">Delete</a>

Add new brand

© Bojan Maksimović 92/19

## Operating systems

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats

Logins (24h)

Users

Phones

Brands

Operating systems

Orders

Messages

ID	Name	Actions
1	Android	<a href="#">Edit</a> <a href="#">Delete</a>
2	iOS	<a href="#">Edit</a> <a href="#">Delete</a>

Add new OS

© Bojan Maksimović 92/19

## Orders

MS

Admin panel | Welcome admin! | Log out

Home Phones Contact Author

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

ID	Buyer username	Buyer name	Buyer address	Total price	Actions
1	boki200075	Pera Peric	Dimitrija Tucovica 154A	12900€	<button>Edit</button> <button>Delete</button>
4	korisnik1	Bojan Maksimovic	Cingrijina 13	3920€	<button>Edit</button> <button>Delete</button>
5		Bojan Maksimovic	Ljubljanska 11	4850€	<button>Edit</button> <button>Delete</button>

© Bojan Maksimović 92/19

## Messages

MS

[Home](#)
[Phones](#)
[Contact](#)
[Author](#)

Page stats  
Logins (24h)  
Users  
Phones  
Brands  
Operating systems  
Orders  
Messages

Username:  
Name: Bojan Maksimović  
Email: bojan.maxim075@gmail.com  
Message:  
aa  
aa  
Delete

Username: admin  
Name: Bojan Maksimović  
Email: bojan.maxim075@gmail.com  
Message:  
aa  
aa  
Delete

Username: admin  
Name: Bojan Maksimović  
Email: bojan.maxim075@gmail.com  
Message:  
cc  
cc  
cc  
Delete

Username:

© Bojan Maksimović 92/19

Admin panelu imaju pristup samo korisnici sa privilegijama administratora. U slučaju da regularan korisnik pokuša da pristupi admin panelu putem url adrese biće redirektovan na početnu stranu.

U admin panelu moguće je vršiti CRUD operacije nad korisnicima, telefonima, brendovima, operativnim sistemima, porudžbinama i porukama poslatim putem kontakt forme.

Prilikom dodavanja novog telefona ili izmene postojećeg, uploadovana slika se čuva i u originalu i smanjenih dimenzija i obe putanje se upisuju u bazu.

U admin panelu takođe se nalazi i uvid u to koliko je koja stranica posećivana u procentima, koliko koja stranica ima poseta u poslednjih 24h i koji i koliko korisnika se ulogovalo u poslednjih 24h.

U sekciji za korisnike se nalazi i 'Export to Excel' taster koji tabelu sa korisnicima exportuje u Excel fajl i downloaduje ga (web server na kom je sajt hostovan radi na Linux-u a klasa 'COM' je ekskluzivna za Windows pa će klik na taster da vrati error kod 500 ali kod za export i download postoji u <models/admin/users/export-to-excel.php>).



## 3. Kodovi

### 3.1 PHP

#### index.php

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

    require_once('views/fixed/head.php');
    require_once('views/fixed/header.php');

    if(isset($_GET['p'])) {
        switch($_GET['p']) {
            case 'home':
                require_once('views/pages/home.php');
                break;
            case 'phones':
                require_once('views/pages/phones.php');
                break;
            case 'details':
                require_once('views/pages/details.php');
                break;
            case 'contact':
                require_once('views/pages/contact.php');
                break;
            case 'author':
                require_once('views/pages/author.php');
                break;
            case 'cart':
                require_once('views/pages/cart.php');
                break;
            case 'login':
                require_once('views/pages/login.php');
                break;
            case 'register':
                require_once('views/pages/register.php');
                break;
            case 'admin':
                require_once('views/pages/admin.php');
                break;
            default:
                require_once('views/pages/home.php');
        }
    } else {
        require_once('views/pages/home.php');
    }

    require_once('views/fixed/footer.php');
    require_once('views/fixed/scripts.php');
?>
```

## views/fixe d/header.php

```
<header class="fixed-top">
  <div id="loginbar">
    <div class="container d-flex align-items-center justify-content-end">
      <ul class="d-flex align-items-center m-0">
        <?php
          if(!isset($_SESSION['user'])): // checks if user is logged in
            ?>
              <li>
                <a href="index.php?p=login" class="px-2 right-border font-xs d-
block my-1">Log in</a>
              </li>
              <li>
                <a href="index.php?p=register" class="px-2 font-xs d-block my-
1">Register</a>
              </li>
            <?php
              else:
                $user = $_SESSION['user'];
                if($user->role_name == 'admin'): // checks if user is an admin
                  ?>
                    <li>
                      <a href="index.php?p=admin" class="px-2 font-xs d-block my-
1">Admin panel</a>
                    </li>
                    <?php
                      endif;
                    ?>
                    <li class="px-2 right-border font-xs d-block my-
1"><label>Welcome <span class="fw-bold"><?= $user->username ?></span>!</label></li>
                    <li>
                      <a href="models/users/logout.php" class="px-2 font-xs d-block my-
1">Log out</a>
                    </li>
                  <?php
                    endif;
                  ?>
                </ul>
              </div>
            </div>
          <nav id="navbar" class="shadow">
            <div class="container d-flex justify-content-between align-items-center">
              <div id="logo">
                <h1 class="m-0">
                  <a href="index.php?p=home" class="font-xl p-1 d-block">MS</a>
                </h1>
              </div>
              <div id="navAndCart" class="d-flex align-items-center">
                <div id="nav" class="d-none d-md-block me-2">
                  <ul class="d-flex m-0">
                    <?php
```

```

        require_once('models/nav/functions.php');
        $links = getNavLinks();
        foreach($links as $l):
            ?>
            <li class="px-2">
                <a href="index.php?p=<?= $l->href ?>" class="py-1 font-
small"><?= $l->page_name ?></a>
            </li>
        <?php
            endforeach;
        ?>
    </ul>
</div>
<a href="index.php?p=cart" id="cart" class="font-medium p-2 position-
relative">
    <i class="fas fa-shopping-cart"></i>
    <span id="cartQuantity" class="position-absolute font-xs rounded-
circle d-flex align-items-center justify-content-center"></span>
</a>
<a href="#" id="hamburgerMenu" class="font-medium d-block d-md-none ms-
1 p-2">
    <i class="fas fa-bars"></i>
</a>
</div>
</div>
</nav>
<div id="sidenav" class="cover">
    <div id="sidenavContent"><ul class="d-flex m-0 d-flex flex-column p-0 py-
2"><?php foreach($links as $l):?><li><a href='index.php?p=<?= $l->href ?>' class='py-
1 font-small d-block p-2'><?= $l->page_name ?></a></li><?php endforeach;?></ul></div>
    </div>
</header>
<main id="main" class="min-vh-100">

```

## views/fixed/head.php

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>Mobishop</title>
    <meta name="description" content="Mobi Shop gives you a chance to quickly and easily find the phone you want and have it delivered to your home in no time..." />
    <meta name="keywords" content="Mobi Shop, smartphones, cell phones, Shop, Home" />
    <link rel="icon" href="assets/icon/favicon.ico" />
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css" crossorigin="anonymous" />
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-e0JMYsd53ii+sc0/bJGFsiCZc+5NDVN2yr8+0RDqr0Ql0h+rP48ckxlpbzKgwra6" crossorigin="anonymous" />
    <link rel="stylesheet" href="assets/css/style.min.css" />
  </head>
  <body>
```

## views/fixed/footer.php

```
</main>
<footer id="footer">
    <div class="container py-2 d-flex align-items-center justify-content-center flex-
column">
        <div id="footerLinks">
            <ul class="d-flex m-0 p-0">
                <?php
                    require_once('models/footer/functions.php');
                    $links = getFooterLinks();
                    foreach($links as $l):
                        ?>
                            <li>
                                <a href="<?= $l->href ?>" target="_blank" class="p-1 font-
small"><i class="<?= $l->icon ?>"></i></a>
                            </li>
                        <?php
                            endforeach;
                        ?>
                    </ul>
                </div>
                <div class="footerSeparator my-2">
                </div>
                <span class="font-xs">&copy; Bojan Maksimović 92/19</span>
            </div>
        </footer>
        <a href="#" id="toTop" class="position-fixed p-3 rounded-circle shadow">
            <div id="toTopArrow" class="position-relative m-1"></div>
        </a>
```

## views/fixed/scripts.php

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta3/dist/js/bootstrap.bundle.min.js" integrity="sha384-JEW9xMcG8R+pH31jmWH6WP0WintQrMb4s7Z0dauHnUtxwoG2vI5DkLtS3qm9Ekf" crossorigin="anonymous">
</script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
<script src="assets/js/main.js"></script>
<?php
    if(isset($_GET['p'])) {
        switch($_GET['p']) {
            case 'phones':
                echo("<script src='assets/js/phones.min.js'></script>");
                break;
            case 'details':
                echo("<script src='assets/js/phones.min.js'></script>");
                break;
            case 'contact':
                echo("<script src='assets/js/form-validation.min.js'></script>");
                echo("<script src='assets/js/contact.min.js'></script>");
                break;
            case 'cart':
                echo("<script src='assets/js/form-validation.min.js'></script>");
                echo("<script src='assets/js/cart.min.js'></script>");
                break;
            case 'login':
                echo("<script src='assets/js/form-validation.min.js'></script>");
                echo("<script src='assets/js/login.min.js'></script>");
                break;
            case 'register':
                echo("<script src='assets/js/form-validation.min.js'></script>");
                echo("<script src='assets/js/register.min.js'></script>");
                break;
            case 'admin':
                echo("<script src='assets/js/form-validation.min.js'></script>");
                echo("<script src='assets/js/admin.min.js'></script>");
                break;
        }
    }
?>
</body>
</html>
```

## views/pages/home.php

```
<div id="slider" class="carousel slide vh-100 position-relative" data-bs-ride="carousel">
  <div class="carousel-inner h-100">
    <?php
      require_once('models/slider/functions.php');
      $sliderImages = getSliderImages();
      foreach($sliderImages as $key=>$i):
    ?>
      <div class="carousel-item slider-image <?php if($key == 0) echo('active'); ?> h-100" style="background-image: url('assets/img/slider/<?=$i->href ?>');">
      </div>
    <?php
      endforeach;
    ?>
  </div>
  <div id="sliderText" class="position-absolute cover">
    <div class="container d-flex flex-column align-items-center align-items-md-start justify-content-center h-100">
      <h2 class="font-xxl">All New Phones</h2>
      <p class="font-medium text-center mt-2">Special deals on the latest cell phones and smartphones. Stay connected in style, we've got you covered.</p>
      <a href="index.php?p=phones" class="font-small d-block mt-2 p-2 rounded">Shop now</a>
    </div>
  </div>
</div>
<div id="about">
  <div class="container py-5">
    <div class="row align-items-center">
      <div class="col-12 col-lg-4">
        <div id="aboutUsImage" class="mx-auto mb-4 mb-lg-0">
          
        </div>
      </div>
      <div class="col-12 col-lg-8 text-center text-lg-end">
        <h3 class="font-large">About us</h3>
        <h4 class="font-xl fw-bold">Mobishop</h4>
        <p class="font-small">
          Mobi Shop gives you a chance to quickly and easily find the phone you want and have it delivered to your home in no time, regardless of your location, as long as it is in one of the countries of the EU.
        </p>
        <p class="font-small">
          We have been in the business for quite a while now, and it that time we have not only managed to make close relationships with numerous suppliers all over the world, but also to recognize what people need.
        </p>
      </div>
    </div>
  </div>
</div>
```

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



## views/pages/phones.php

```
<div class="d-flex min-vh-100 py-5">
  <div class="container mt-5">
    <div class="mt-0 mt-md-3">
      <div class="row">
        <div class="col-12 col-md-3">
          <form class="pt-0 pt-md-5">
            <div class="mb-2 mt-0 mt-md-5">
              <label>Brands:</label>
              <?php
                require_once("models/phones/functions.php");
                $brands = getBrands();
                foreach($brands as $b):
              ?>
              <div class="form-check">
                <input class="form-check-
input chbBrand" type="checkbox" value="<?= $b->id ?>" id="chbBrand<?= $b->id ?>">
                <label class="form-check-
label" for="chbBrand<?= $b->id ?>">
                  <?= $b->name ?>
                </label>
              </div>
              <?php
                endforeach;
              ?>
            </div>
            <div class="mb-2">
              <label>Operating systems:</label>
              <?php
                $operatingSystems = getOperatingSystems();
                foreach($operatingSystems as $o):
              ?>
              <div class="form-check">
                <input class="form-check-
input chbOS" type="checkbox" value="<?= $o->id ?>" id="chbOS<?= $o->id ?>">
                <label class="form-check-label" for="chbOS<?= $o-
>id ?>">
                  <?= $o->name ?>
                </label>
              </div>
              <?php
                endforeach;
              ?>
            </div>
          </form>
        </div>
        <div class="col-12 col-md-9">
          <form>
            <div class="row">
              <div class="col-12 col-md-3">
```

```

        <label for="tbMaxPrice" class="font-
small">Max. price (&euro;):</label>
        <input type="number" name="tbMaxPrice" id="tbMaxPrice"
class="form-control font-small" placeholder="Enter max. price"/>
    </div>
    <div class="col-12 col-md-5">
        <label for="tbSearch" class="font-
small">Search products:</label>
        <input type="search" name="tbSearch" id="tbSearch" cla
ss="form-control font-small" placeholder="Enter product name" autocomplete="off"/>
    </div>
    <div class="col-12 col-md-4">
        <label for="ddlSortOrder" class="font-
small">Sort order:</label>
        <select name="ddlSortOrder" id="ddlSortOrder" class="f
orm-select">
            <option value="0">Recommended</option>
            <option value="1">Name A-Z</option>
            <option value="2">Name Z-A</option>
            <option value="3">Price ascending</option>
            <option value="4">Price descending</option>
        </select>
    </div>
</div>
</form>
    <div id="phones" class="mt-5">
        <div class="row">
            <div id="pagination" class="mt-4 d-flex align-items-
center justify-content-center">
                </div>
            </div>
        </div>
    </div>
</div>
</div>

```

## views/pages/details.php

```
<?php
    if(!isset($_GET['id'])) {
        header("Location: index.php?p=phones");
        exit();
    }
    require_once("models/phones/functions.php");
    $phone = getPhoneById($_GET['id']);
    if(!$phone) {
        header("Location: index.php?p=phones");
        exit();
    } else {
        if(!$phone->active) {
            header("Location: index.php?p=phones");
            exit();
        }
    }
}

?>
<div class="d-flex align-items-center min-vh-100 py-5">
    <div class="container mt-5 mt-md-0">
        <div class="row">
            <div class="col-12 col-md-6">
                <div id="phoneImage">
                    name ?>" class="img-fluid"/>
                </div>
            </div>
            <div class="col-12 col-md-6 d-flex align-items-center align-items-md-start justify-content-center flex-column">
                <label id="phoneName" class="font-large text-center text-md-start"><?= $phone->name ?></label>
                <?php
                    $brand = getBrandById($phone->brand);
                    $os = getOSById($phone->os);
                ?>
                <div id="phoneBrand">
                    <label class="font-medium"><span class="fw-bold">Brand: </span><?= $brand->name ?></label>
                </div>
                <div id="phoneOS">
                    <label class="font-medium"><span class="fw-bold">OS: </span><?= $os->name ?></label>
                </div>
                <div id="addToCart" class="mt-3 d-flex justify-content-center justify-content-md-start">
                    <input type="number" name="tbAddToCart" id="tbAddToCart" class="form-control font-small w-25 rounded-0 rounded-start" value="1"/>
                    <a href="#" data-id="<?= $_GET['id'] ?>" id="btnAddToCart" class="text-center btn-primary font-medium py-1 px-4 rounded-end">
                        <i class="fas fa-cart-plus"></i>
                    </a>
                </div>
            </div>
        </div>
    </div>
</div>
```

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

## views/pages/contact.php

```
<div class="d-flex align-items-center min-vh-100 py-5">
<div class="container mt-5 mt-md-0">
  <h2 class="font-xl text-center">Contact us</h2>
  <div class="underline mb-5"></div>
  <div class="row mt-5 mt-md-0 justify-content-between">
    <div class="col-12 col-md-6 mb-4 mb-md-0">
      <p class="font-
small">If you wish to reach out to us, you can do so via social media, the information lis
ted below, or by filling out the contact form.</p>
      <ul class="p-0">
        <li class="font-small"><i class="fas fa-phone me-1"></i>+381 65/518-
3225</li>
        <li class="font-small"><i class="fas fa-envelope me-
1"></i>bojan.maksimovic.92.19@ict.edu.rs</li>
      </ul>
    </div>
    <div class="col-12 col-md-5">
      <h3 class="font-medium text-center mb-4">Contact form</h3>
      <form action="models/contact/send-message.php" method="POST">
        <div class="mb-2">
          <label for="tbEmail">Email:</label>
          <input type="email" name="email" id="tbEmail" class="form-control"/>
          <label class="font-small error-message text-
danger">examplename@example.com</label>
        </div>
        <div class="mb-2">
          <label for="tbName">Name:</label>
          <input type="text" name="name" id="tbName" class="form-control"/>
          <label class="font-small error-message">
            <ul class="p-0">
              <li class="text-
danger">- A name cannot have less than 3 characters</li>
              <li class="text-
danger">- Each word of your name must be capitalized</li>
            </ul>
          </label>
        </div>
        <div class="mb-3">
          <label for="tbMessage">Message:</label>
          <textarea name="message" id="tbMessage" class="form-
control" style="resize: none; height: 200px;"></textarea>
          <label class="font-small error-message text-
danger">Your message (without spacing characters) cannot be shorter than 20 characters or
longer than 500 characters in total.</label>
        </div>
        <div class="text-center mb-3">
          <button id="btnSend" name="btnSend" class="py-1 px-3 rounded btn-
primary"><i class="fas fa-paper-plane font-small"></i></button>
        </div>
      </form>
    </div>
  </div>
</div>
```

```
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
</div>
</div>
</div>
</div>
```

## views/pages/author.php

```
<div class="d-flex align-items-center min-vh-100 py-5">
<div class="container">
    <div class="row align-items-center justify-content-between mt-5 mt-md-0">
        <div class="col-12 col-md-4 d-flex flex-column align-items-center justify-content-center">
            <div id="authorImage" class="position-relative rounded-circle overflow-hidden">
                
            <div id="authorSocials" class="position-absolute cover-white align-items-center justify-content-center">
                <ul class="align-items-center justify-content-center m-0 p-0">
                    <?php
                        require_once("models/author/functions.php");
                        $links = getAuthorSocials();
                        foreach($links as $l):
                    <?>
                    <li>
                        <a href="<?= $l->href ?>" target="_blank" class="p-1 font-medium"><i class="<?= $l->icon ?>" style="color: <?= $l->color ?>";"></i></a>
                    </li>
                <?php
                    endforeach;
                <?>
            </ul>
        </div>
    </div>
    <span class="font-medium fw-bold mt-2 text-center">Bojan Maksimović 92/19</span>
    </div>
    <div class="col-12 col-md-6 text-center text-md-start mt-4 mt-md-0">
        <p class="font-small">My name is Bojan and I am an aspiring front-end web developer. My approach to website design is to create a website that strengthens y our company's brand while ensuring ease of use and simplicity for your audience.</p>
        <p class="font-small">The way I look at it, a front-end developer's role is to combine design and business logic to achieve a user-facing product. To do this successfully, a wide skill set is necessary to produce a quality user experience that leads to meeting business goals, and I guarantee I've got exactly what's needed.</p>
        <a href="https://bokule.github.io/portfolio/" target="_blank" class="font-small rounded py-2 px-3 text-uppercase btn-primary" id="portfolioLink">My portfolio</a>
    </div>
    </div>
    <div class="mt-4 text-center text-md-end">
        <a href="models/author/export-to-word.php" class="font-small rounded py-2 px-3 btn-primary">Export to Word</a>
    </div>
</div>
</div>
```

## views/pages/cart.php

```
<?php
    if(!isset($_SESSION['user'])) {
        $_SESSION['errors'] = ["You need to be logged in in order to access your cart."];
        header("Location: index.php?p=login");
        exit();
    }
?>
<div class="d-flex min-vh-100 py-5">
    <div class="container mt-5">
        <div class="mt-0 mt-md-3">
            <div class="row flex-column-reverse flex-md-row">
                <div class="col-12 col-md-8 mt-4 mt-md-0">
                    <div class="table-responsive">
                        <table id="cartTable" class="font-small table table-striped table-
hover align-middle" cellspacing="0">
                            <thead>
                                <tr>
                                    <td class="colDeviceName text-center">Device name</td>
                                    <td class="colPrice text-center">Price</td>
                                    <td class="colQuantity text-center">Quantity</td>
                                    <td class="colRemove text-center px-
3"><i class="fas fa-trash"></i></td>
                                </tr>
                            </thead>
                            <tbody>
                            </tbody>
                        </table>
                    </div>
                </div>
                <div class="col-12 col-md-4 mt-2">
                    <form action="models/cart/create-
order.php" method="POST" id="checkoutForm">
                        <h3 class="font-medium text-center">Checkout form</h3>
                        <div class="mb-2 mt-2">
                            <label for="tbName" class="font-small">Full name:</label>
                            <input type="text" name="tbName" id="tbName" class="form-
control font-small" autocomplete="off"/>
                            <label class="error-message">
                                <ul class="font-small p-0">
                                    <li class="text-
danger">- A name cannot have less than 3 characters</li>
                                    <li class="text-
danger">- Has to be at least 2 words</li>
                                    <li class="text-
danger">- Each word must be capitalized</li>
                                </ul>
                            </label>
                        </div>
                        <div class="mb-2">
                            <label for="tbAddress" class="font-small">Address:</label>
```



```

        <input type="text" name="tbAddress" id="tbAddress" class="form
-control font-small" autocomplete="off"/>
        <label class="error-message font-small text-
danger">[Street Name] [Home Number]/[Apartment Number]</label>
    </div>
    <label>Total price: <span class="fw-
bold"><span id="totalPrice">0</span>&euro;</span></label>
    <div class="mt-2 mb-3">
        <input type="submit" value="Checkout" class="font-small btn-
primary d-block p-2 w-100 rounded" id="btnCheckout" name="btnCheckout"/>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
</div>
</div>
</div>
</div>
</div>
</div>

```

## views/pages/login.php

```
<?php
    if(isset($_SESSION['user'])) {
        header("Location: index.php?p=home");
        exit();
    }
?>
<div class="d-flex align-items-center min-vh-100 py-5">
    <div class="container">
        <div class="row">
            <div id="login-form" class="col-12 col-sm-8 col-md-6 col-lg-4 mx-auto">
                <h3 class="font-large mb-3 text-center">Log in</h3>
                <form action="models/users/login.php" method="POST">
                    <div class="mb-2">
                        <label for="tbEmail">Email:</label>
                        <input type="email" name="email" id="tbEmail" class="form-
control"/>
                        <label class="font-small error-message text-
danger">examplename@example.com</label>
                    </div>
                    <div class="mb-3">
                        <label for="tbPassword">Password:</label>
                        <input type="password" name="password" id="tbPassword" class="form-
-control"/>
                        <label class="font-small error-message text-
danger">Your password must:
                            <ul class="p-0">
                                <li class="text-danger">- contain 1 lowercase letter</li>
                                <li class="text-danger">- contain 1 uppercase letter</li>
                                <li class="text-danger">- contain 1 number</li>
                                <li class="text-danger">- contain 1 special character</li>
                                <li class="text-danger">- be 8 characters or longer</li>
                            </ul>
                        </label>
                    </div>
                    <div class="text-center mb-3">
                        <button id="btnLogIn" name="btnLogIn" class="py-1 px-
3 rounded btn-primary font-small">Log in</button>
                    </div>
                    <?php
                        require_once('models/forms/functions.php');
                        displayMessages();
                    ?>
                </form>
                <a href="index.php?p=register">Don't have an account?<br/>Register instead
</a>
            </div>
        </div>
    </div>
</div>
```

## views/pages/register.php

```
<?php
    if(isset($_SESSION['user'])) {
        header("Location: index.php?p=home");
        exit();
    }
?>
<div class="d-flex align-items-center min-vh-100 py-5">
    <div class="container">
        <div class="row">
            <div id="register-form" class="col-12 col-sm-8 col-md-6 col-lg-4 mx-auto">
                <h3 class="font-large mb-3 text-center">Register</h3>
                <form action="models/users/register.php" method="POST">
                    <div class="mb-2">
                        <label for="tbUsername">Username:</label>
                        <input type="text" name="username" id="tbUsername" class="form-
control"/>
                        <label class="font-small error-message text-
danger">Your username must:
                            <ul class="p-0">
                                <li class="text-danger">- start with a letter</li>
                                <li class="text-
danger">- be between 5 and 20 characters long</li>
                                <li class="text-
danger">- not contain special characters other than '_'</li>
                            </ul>
                        </label>
                    </div>
                    <div class="mb-2">
                        <label for="tbEmail">Email:</label>
                        <input type="email" name="email" id="tbEmail" class="form-
control"/>
                        <label class="font-small error-message text-
danger">examplename@example.com</label>
                    </div>
                    <div class="mb-2">
                        <label for="tbPassword">Password:</label>
                        <input type="password" name="password" id="tbPassword" class="form-
-control"/>
                        <label class="font-small error-message text-
danger">Your password must:
                            <ul class="p-0">
                                <li class="text-danger">- contain 1 lowercase letter</li>
                                <li class="text-danger">- contain 1 uppercase letter</li>
                                <li class="text-danger">- contain 1 number</li>
                                <li class="text-danger">- contain 1 special character</li>
                                <li class="text-danger">- be 8 characters or longer</li>
                            </ul>
                        </label>
                    </div>
                    <div class="mb-3">
```

```

        <label for="tbRepeatPassword">Repeat password:</label>
        <input type="password" name="repeatPassword" id="tbRepeatPassword"
class="form-control"/>
        <label class="font-small error-message text-
danger">Passwords don't match</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnRegister" name="btnRegister" class="py-1 px-
3 rounded btn-primary font-small">Register</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
    <a href="index.php?p=login">Already have an account?<br/>Log in instead</a
>
    </div>
</div>
</div>
</div>
</div>

```

## views/pages/admin.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            header('Location: index.php?p=home');
            exit();
        }
    } else {
        header('Location: index.php?p=home');
        exit();
    }
?>
<div class="d-flex min-vh-100 py-5">
    <div class="container mt-5">
        <div class="mt-0 mt-md-3">
            <div class="row">
                <div class="col-12 col-md-4">
                    <ul class="p-0 font-medium">
                        <?php
                            require_once('models/admin/functions.php');
                            $links = getAdminLinks();
                            foreach($links as $l):
                                ?>
                                <li>
                                    <a href="index.php?p=admin&ap=<?= $l->href ?>"><?= $l->page_name ?></a>
                                </li>
                                <?php
                                    endforeach;
                                ?>
                            </ul>
                        </div>
                        <div class="col-12 col-md-8">
                            <?php
                                if(isset($_GET['ap'])) {
                                    switch($_GET['ap']) {
                                        case 'page-stats':
                                            require_once('admin/page-stats.php');
                                            break;
                                        case 'logins':
                                            require_once('admin/logins.php');
                                            break;
                                        case 'users':
                                            require_once('admin/users.php');
                                            break;
                                        case 'add-user':
                                            require_once('admin/users/add-user-form.php');
                                            break;
                                        case 'edit-user':
                                            require_once('admin/users/edit-user-form.php');
                                            break;
                                    }
                                }
                            ?>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

        case 'phones':
            require_once('admin/phones.php');
            break;
        case 'add-phone':
            require_once('admin/phones/add-phone-form.php');
            break;
        case 'edit-phone':
            require_once('admin/phones/edit-phone-form.php');
            break;
        case 'brands':
            require_once('admin/brands.php');
            break;
        case 'add-brand':
            require_once('admin/brands/add-brand-form.php');
            break;
        case 'edit-brand':
            require_once('admin/brands/edit-brand-form.php');
            break;
        case 'operating-systems':
            require_once('admin/operating-systems.php');
            break;
        case 'add-os':
            require_once('admin/os/add-os-form.php');
            break;
        case 'edit-os':
            require_once('admin/os/edit-os-form.php');
            break;
        case 'orders':
            require_once('admin/orders.php');
            break;
        case 'edit-order':
            require_once('admin/orders/edit-order-form.php');
            break;
        case 'messages':
            require_once('admin/messages.php');
            break;
        default:
            require_once('admin/page-stats.php');
    }
} else {
    require_once('admin/page-stats.php');
}

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

```

## views/pages/admin/page-stats.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">Page</td>
                <td class="text-center">%</td>
                <td class="text-center">24h</td>
            </tr>
        </thead>
        <tbody>
            <?php
                $file = file(LOG_FILE);
                $totalVisits = 0;
                $pages = [];
                $visits = [];
                $visits24h = [];
                foreach($file as $row) {
                    $values = explode(SEPARATOR, $row);
                    $pageName = $values[2];
                    $dateAndTime = trim($values[4]);

                    if($pageName == '') continue;

                    if(!in_array($pageName, $pages)) {
                        array_push($pages, $pageName);
                        array_push($visits, 1);
                        if(time() - getTime($dateAndTime) <= 60*60*24) {
                            array_push($visits24h, 1);
                        } else {
                            array_push($visits24h, 0);
                        }
                    } else {
                        foreach($pages as $key=>$p) {
                            if($p == $pageName) {
                                $index = $key;
                                break;
                            }
                        }
                    }
                }
            ?>
        </tbody>
    </table>
</div>
```

```

        $visits[$index]++;
        if(time() - getTime($dateAndTime) <= 60*60*24) {
            $visits24h[$index]++;
        }
    }
    $totalVisits++;
}

foreach($pages as $key=>$p):
    $percentage = round($visits[$key] * 100 / $totalVisits, 2);
?>
    <tr>
        <td class="text-center"><?= $p ?></td>
        <td class="text-center"><?= $percentage ?>%</td>
        <td class="text-center"><?= $visits24h[$key] ?></td>
    </tr>
<?php
    endforeach;
?>
</tbody>
</table>
</div>

```



## views/pages/admin/logins.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellpadding="0">
        <thead>
            <tr>
                <td class="text-center">ID</td>
                <td class="text-center">Username</td>
                <td class="text-center">Email</td>
            </tr>
        </thead>
        <tbody>
            <?php
                require_once('models/users/functions.php');
                $file = file(LOGIN_LOG);
                $userIds = [];
                foreach($file as $row) {
                    $values = explode(SEPARATOR, $row);
                    $userId = $values[0];
                    $dateAndTime = $values[2];
                    $result = trim($values[3]);
                    if(!in_array($userId, $userIds) && time() - getTime($dateAndTime) <= 6
0*60*24 && $result == '1') {
                        array_push($userIds, $userId);
                    }
                }

                $numberOfUsers = 0;
                foreach($userIds as $u):
                    $user = getUser('id', $u);
                    if($user):
                        $numberOfUsers++;
                ?>
            <tr>
                <td class="text-center"><?= $user->id ?></td>
                <td class="text-center"><?= $user->username ?></td>
                <td class="text-center"><?= $user->email ?></td>
            </tr>
            <?php
                endif;
```

```
        endforeach;  
        ?>  
    </tbody>  
</table>  
</div>  
<label class="mt-3 font-medium">Unique users: <span class="fw-  
bold"><?= $numberOfUsers ?></span></label>
```

## views/pages/admin/users.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>

<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">ID</td>
                <td class="text-center">Username</td>
                <td class="text-center">Email</td>
                <td class="text-center">Role</td>
                <td class="text-center">Active</td>
                <td class="text-center colActions">Actions</td>
            </tr>
        </thead>
        <tbody>
            <?php
                $users = getAllUsers();

                foreach($users as $u):
                    ?>
                    <tr>
                        <td class="text-center"><?= $u->id ?></td>
                        <td class="text-center"><?= $u->username ?></td>
                        <td class="text-center"><?= $u->email ?></td>
                        <td class="text-center"><?= $u->role_name ?></td>
                        <td class="text-center"><?= $u->active ?></td>
                        <td class="text-center colActions">
                            <?php
                                if($u->id != $_SESSION['user']->id):
                                    ?>
                                    <a href="index.php?p=admin&ap=edit-user&id=<?= $u-
>id ?>" class="font-small btn-primary px-2 py-1 rounded">Edit</a>
                                    <a href="models/admin/delete.php?id=<?= $u-
>id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-1 rounded">Delete</a>
                                <?php
                                    endif;
                                ?>
                            </td>
                        </tr>
                    </foreach>
                </tbody>
            </table>
        </div>
```

```
        </td>
    </tr>
<?php
    endforeach;
?>
</tbody>
</table>
</div>
<a href="index.php?p=admin&ap=add-user" class="font-small btn-primary px-2 py-
1 rounded">Add new user</a>
<a href="models/admin/users/export-to-excel.php" class="font-small btn-primary px-2 py-
1 rounded">Export to Excel</a>
```

## views/pages/admin/phones.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>
<div class="table-responsive mb-2">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">Name</td>
                <td class="text-center">OS</td>
                <td class="text-center">Brand</td>
                <td class="text-center">Price</td>
                <td class="text-center">Active</td>
                <td class="text-center colActions">Actions</td>
            </tr>
        </thead>
        <tbody>
            <?php
                require_once('models/phones/functions.php');
                $phones = getAllPhones();

                foreach($phones as $p):
                    ?>
                    <tr>
                        <td class="text-center"><?= $p->phone_name ?></td>
                        <td class="text-center"><?= $p->os_name ?></td>
                        <td class="text-center"><?= $p->brand_name ?></td>
                        <td class="text-center"><?= $p->price ?>&euro;</td>
                        <td class="text-center"><?= $p->active ?></td>
                        <td class="text-center colActions">
                            <a href="index.php?p=admin&ap=edit-phone&id=<?= $p-
>phone_id ?>" class="font-small btn-primary px-2 py-1 rounded">Edit</a>
                            <a href="models/admin/delete.php?id=<?= $p-
>phone_id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-
1 rounded">Delete</a>
                        </td>
                    </tr>
                <?php
                    endforeach;
            </tbody>
        </table>
    </div>
```

```
        ?>
    </tbody>
</table>
</div>
<a href="index.php?p=admin&ap=add-phone" class="font-small btn-primary px-2 py-
1 rounded">Add new phone</a>
```

## views/pages/admin/brands.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>

<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">ID</td>
                <td class="text-center">Name</td>
                <td class="text-center colActions">Actions</td>
            </tr>
        </thead>
        <tbody>
            <?php
                $brands = getAllBrands();

                foreach($brands as $b):
            ?>
                <tr>
                    <td class="text-center"><?= $b->id ?></td>
                    <td class="text-center"><?= $b->name ?></td>
                    <td class="text-center colActions">
                        <a href="index.php?p=admin&ap=edit-brand&id=<?= $b-
>id ?>" class="font-small btn-primary px-2 py-1 rounded">Edit</a>
                        <a href="models/admin/delete.php?id=<?= $b-
>id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-1 rounded">Delete</a>
                    </td>
                </tr>
            <?php
                endforeach;
            ?>
        </tbody>
    </table>
</div>

<a href="index.php?p=admin&ap=add-brand" class="font-small btn-primary px-2 py-
1 rounded">Add new brand</a>
```

```

<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>

<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">ID</td>
                <td class="text-center">Name</td>
                <td class="text-center colActions">Actions</td>
            </tr>
        </thead>
        <tbody>
            <?php
                $os = getAllOS();

                foreach($os as $o):
            ?>
                <tr>
                    <td class="text-center"><?= $o->id ?></td>
                    <td class="text-center"><?= $o->name ?></td>
                    <td class="text-center colActions">
                        <a href="index.php?p=admin&ap=edit-os&id=<?= $o-
>id ?>" class="font-small btn-primary px-2 py-1 rounded">Edit</a>
                        <a href="models/admin/delete.php?id=<?= $o-
>id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-1 rounded">Delete</a>
                    </td>
                </tr>
            <?php
                endforeach;
            ?>
        </tbody>
    </table>
</div>
<a href="index.php?p=admin&ap=add-os" class="font-small btn-primary px-2 py-
1 rounded">Add new OS</a>

```



## views/pages/admin/orders.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>

<div class="table-responsive">
    <table class="font-small table table-striped table-hover align-
middle" cellspacing="0">
        <thead>
            <tr>
                <td class="text-center">ID</td>
                <td class="text-center">Buyer username</td>
                <td class="text-center">Buyer name</td>
                <td class="text-center">Buyer address</td>
                <td class="text-center">Total price</td>
                <td class="text-center colActions">Actions</td>
            </tr>
        </thead>
        <tbody>
            <?php
                $orders = getAllOrders();

                foreach($orders as $o):
                    ?>
                    <tr>
                        <td class="text-center"><?= $o->order_id ?></td>
                        <td class="text-center"><?= $o->username ?></td>
                        <td class="text-center"><?= $o->name ?></td>
                        <td class="text-center"><?= $o->address ?></td>
                        <td class="text-center"><?= $o->total_price ?>&euro;</td>
                        <td class="text-center colActions">
                            <a href="index.php?p=admin&ap=edit-order&id=<?= $o-
>order_id ?>" class="font-small btn-primary px-2 py-1 rounded">Edit</a>
                            <a href="models/admin/delete.php?id=<?= $o-
>order_id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-
1 rounded">Delete</a>
                        </td>
                    </tr>
                <?php
                    endforeach;
            ?>
        </tbody>
    </table>
</div>
```

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

## views/pages/admin/messages.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('models/forms/functions.php');
    displayMessages();
?>
<div class="messagesDiv overflow-auto">
<?php
    $messages = getAllMessages();

    foreach($messages as $m):
?>
<div class="messageDiv">
    <hr/>
    <label class="font-small">
        <span class="fw-bold">Username: </span><?= $m->username ?>
    </label>
    <br/>
    <label class="font-small">
        <span class="fw-bold">Name: </span><?= $m->name ?>
    </label>
    <br/>
    <label class="font-small">
        <span class="fw-bold">Email: </span><?= $m->email ?>
    </label>
    <br/>
    <label class="font-small">
        <span class="fw-bold">Message: </span>
    </label>
    <br/>
    <span class="text-break d-block mb-2"><?= $m->message ?></span>
    <a href="models/admin/delete.php?id=<?= $m-
>message_id ?>&ap=<?= $_GET['ap'] ?>" class="font-small btn-primary px-2 py-
1 rounded">Delete</a>
</div>
<?php
    endforeach;
?>
</div>
```

## views/pages/admin/users/add-user-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<form action="models/admin/users/add-user.php" method="POST">
    <div class="mb-2">
        <label for="tbAddUserUsername">Username:</label>
        <input type="text" name="username" id="tbAddUserUsername" class="form-control"/>
        <label class="font-small error-message text-danger">Your username must:
            <ul class="p-0">
                <li class="text-danger">- start with a letter</li>
                <li class="text-danger">- be between 5 and 20 characters long</li>
                <li class="text-
danger">- not contain special characters other than '_'</li>
            </ul>
        </label>
    </div>
    <div class="mb-2">
        <label for="tbAddUserEmail">Email:</label>
        <input type="email" name="email" id="tbAddUserEmail" class="form-control"/>
        <label class="font-small error-message text-
danger">examplename@example.com</label>
    </div>
    <div class="mb-2">
        <label for="tbAddUserPassword">Password:</label>
        <input type="password" name="password" id="tbAddUserPassword" class="form-
control"/>
        <label class="font-small error-message text-danger">Your password must:
            <ul class="p-0">
                <li class="text-danger">- contain 1 lowercase letter</li>
                <li class="text-danger">- contain 1 uppercase letter</li>
                <li class="text-danger">- contain 1 number</li>
                <li class="text-danger">- contain 1 special character</li>
                <li class="text-danger">- be 8 characters or longer</li>
            </ul>
        </label>
    </div>
    <div class="mb-2">
        <label for="ddlAddUserRole">Role:</label>
        <select name="role" id="ddlAddUserRole" class="form-select">
            <?php
                $roles = getAllRoles();
                foreach($roles as $r):
```

```

        ?>
        <option value="<?= $r->role_id ?>"><?= $r->role_name ?></option>
    <?php
        endforeach;
    ?>
</select>
</div>
<div class="text-center mb-3">
    <button id="btnAddUser" name="btnAddUser" class="py-1 px-3 rounded btn-
primary font-small">Add User</button>
</div>
<?php
    require_once('models/forms/functions.php');
    displayMessages();
?>
</form>

```

## views/pages/admin/users/edit-user-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    if(!isset($_GET['id'])) {
        http_response_code(400);
        exit();
    }

    require_once('models/users/functions.php');
    $user = getUser('id', $_GET['id']);
?>
<form action="models/admin/users/edit-user.php" method="POST">
    <input type="hidden" name="id" value="<?= $user->id ?>" />
    <div class="mb-2">
        <label for="tbEditUserUsername">Username:</label>
        <input type="text" name="username" id="tbEditUserUsername" value="<?= $user->username ?>" class="form-control"/>
        <label class="font-small error-message text-danger">Your username must:
            <ul class="p-0">
                <li class="text-danger">- start with a letter</li>
                <li class="text-danger">- be between 5 and 20 characters long</li>
                <li class="text-
danger">- not contain special characters other than '_'</li>
            </ul>
        </label>
    </div>
    <div class="mb-2">
        <label for="tbEditUserEmail">Email:</label>
        <input type="email" name="email" id="tbEditUserEmail" value="<?= $user->email ?>" class="form-control"/>
        <label class="font-small error-message text-
danger">exemplename@example.com</label>
    </div>
    <div class="mb-2">
        <label for="tbEditUserPassword">Password:</label>
        <input type="password" name="password" id="tbEditUserPassword" class="form-
control mb-1"/>
        <div class="form-check">
            <input class="form-check-
input" type="checkbox" name="chbChangePassword" value="changePassword" id="chbChangePasswo
rd">
            <label class="form-check-label" for="chbChangePassword">
```

```

        Change password
    </label>
</div>
<label class="font-small error-message text-danger">Your password must:
    <ul class="p-0">
        <li class="text-danger">- contain 1 lowercase letter</li>
        <li class="text-danger">- contain 1 uppercase letter</li>
        <li class="text-danger">- contain 1 number</li>
        <li class="text-danger">- contain 1 special character</li>
        <li class="text-danger">- be 8 characters or longer</li>
    </ul>
</label>
</div>
<div class="mb-2">
    <label for="ddlEditUserRole">Role:</label>
    <select name="role" id="ddlEditUserRole" class="form-select">
        <?php
            $roles = getAllRoles();
            foreach($roles as $r):
                ?>
                <option <?php if($r->role_id == $user-
>role) echo('selected'); ?> value="<?=$r->role_id ?>"<?=$r->role_name ?></option>
                <?php
                    endforeach;
                ?>
            </select>
        </div>
        <div class="mb-2">
            <label for="ddlEditUserActive">Active:</label>
            <select name="active" id="ddlEditUserActive" class="form-select">
                <?php
                    for($i = 0; $i <=1; $i++):
                        ?>
                        <option <?php if($i == $user-
>active) echo('selected'); ?> value="<?=$i ?>"<?=$i ?></option>
                        <?php
                            endfor;
                        ?>
                    </select>
                </div>
                <div class="text-center mb-3">
                    <button id="btnEditUser" name="btnEditUser" class="py-1 px-3 rounded btn-
primary font-small">Edit User</button>
                </div>
                <?php
                    require_once('models/forms/functions.php');
                    displayMessages();
                ?>
            </form>

```

## views/pages/admin/phones/add-phone-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<form action="models/admin/phones/add-phone.php" enctype="multipart/form-
data" method="POST">
    <div class="mb-2">
        <label for="tbAddPhoneName">Name:</label>
        <input type="text" name="name" id="tbAddPhoneName" class="form-control"/>
        <label class="font-small error-message text-
danger">Phone name cannot contain special characters other than '-' and '/'</label>
    </div>
    <div class="mb-2">
        <label for="ddlAddPhoneOS">OS:</label>
        <select name="os" id="ddlAddPhoneOS" class="form-select">
            <?php
                $os = getAllOS();
                foreach($os as $o):
                    ?>
                    <option value="<?=$o->id ?>"><?=$o->name ?></option>
                <?php
                endforeach;
            ?>
        </select>
    </div>
    <div class="mb-2">
        <label for="ddlAddPhoneBrand">Brand:</label>
        <select name="brand" id="ddlAddPhoneBrand" class="form-select">
            <?php
                $brands = getAllBrands();
                foreach($brands as $b):
                    ?>
                    <option value="<?=$b->id ?>"><?=$b->name ?></option>
                <?php
                endforeach;
            ?>
        </select>
    </div>
    <div class="mb-2">
        <label for="tbAddPhonePrice">Price:</label>
        <input type="number" name="price" id="tbAddPhonePrice" class="form-control"/>
        <label class="font-small error-message text-danger">You must enter a price</label>
    </div>
```



```

<div class="mb-2">
  <label for="fileAddPhoneImage" class="form-label mb-1">Image:</label>
  <input class="form-control" type="file" id="fileAddPhoneImage" name="image"/>
  <label class="font-small error-message text-
danger">You must choose an image</label>
</div>
<div class="text-center mb-3">
  <button id="btnAddPhone" name="btnAddPhone" class="py-1 px-3 rounded btn-
primary font-small">Add Phone</button>
</div>
<?php
  require_once('models/forms/functions.php');
  displayMessages();
?>
</form>

```

## views/pages/admin/phones/edit-phone-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    if(!isset($_GET['id'])) {
        http_response_code(400);
        exit();
    }

    require_once('models/users/functions.php');
    $phone = getPhone('id', $_GET['id']);
?>
<form action="models/admin/phones/edit-phone.php" enctype="multipart/form-
data" method="POST">
    <input type="hidden" name="id" value="<?= $phone->id ?>" />
    <div class="mb-2">
        <label for="tbEditPhoneName">Name:</label>
        <input type="text" name="name" id="tbEditPhoneName" class="form-
control" value="<?= $phone->name ?>" />
        <label class="font-small error-message text-
danger">Phone name cannot contain special characters other than '-' and '/'</label>
    </div>
    <div class="mb-2">
        <label for="ddlEditPhoneOS">OS:</label>
        <select name="os" id="ddlEditPhoneOS" class="form-select">
            <?php
                $os = getAllOS();
                foreach($os as $o):
                    ?>
                    <option <?php if($o->id == $phone->os) echo('selected'); ?> value="<?= $o-
>id ?>"><?= $o->name ?></option>
                    <?php
                        endforeach;
                    ?>
                </select>
            </div>
            <div class="mb-2">
                <label for="ddlEditPhoneBrand">Brand:</label>
                <select name="brand" id="ddlEditPhoneBrand" class="form-select">
                    <?php
                        $brands = getAllBrands();
                        foreach($brands as $b):
                            ?>
```

```

        <option <?php if($b->id == $phone->brand) echo('selected'); ?> value="<?= $b-
>id ?>"><?= $b->name ?></option>
        <?php
            endforeach;
        ?>
    </select>
</div>
<div class="mb-2">
    <label for="tbEditPhonePrice">Price:</label>
    <input type="number" name="price" id="tbEditPhonePrice" value="<?= $phone-
>price ?>" class="form-control"/>
    <label class="font-small error-message text-danger">You must enter a price</label>
</div>
<div class="mb-2">
    <label for="fileEditPhoneImage" class="form-label mb-1">Image:</label>
    <input class="form-control mb-
1" type="file" id="fileEditPhoneImage" name="image"/>
    <div class="form-check">
        <input class="form-check-
input" type="checkbox" name="chbChangeImage" value="changeImage" id="chbChangeImage">
        <label class="form-check-label" for="chbChangeImage">
            Change image
        </label>
    </div>
    <label class="font-small error-message text-
danger">You must choose an image</label>
</div>
<div class="mb-2">
    <label for="ddlEditPhoneActive">Active:</label>
    <select name="active" id="ddlEditPhoneActive" class="form-select">
        <?php
            for($i = 0; $i <= 1; $i++):
        ?>
        <option <?php if($i == $phone-
>active) echo('selected'); ?> value="<?= $i ?>"><?= $i ?></option>
        <?php
            endfor;
        ?>
    </select>
</div>
<div class="text-center mb-3">
    <button id="btnEditPhone" name="btnEditPhone" class="py-1 px-3 rounded btn-
primary font-small">Edit Phone</button>
</div>
<?php
    require_once('models/forms/functions.php');
    displayMessages();
?>
</form>

```

## views/pages/admin/os/add-os-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<form action="models/admin/os/add-os.php" method="POST">
    <div class="mb-2">
        <label for="tbAddOSName">Name:</label>
        <input type="text" name="name" id="tbAddOSName" class="form-control"/>
        <label class="font-small error-message text-
danger">OS name cannot contain special characters or numbers</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnAddOS" name="btnAddOS" class="py-1 px-3 rounded btn-primary font-
small">Add OS</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
```

## views/pages/admin/os/edit-os-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    if(!isset($_GET['id'])) {
        http_response_code(400);
        exit();
    }

    $os = getOS('id', $_GET['id']);
?>
<form action="models/admin/os/edit-os.php" method="POST">
    <input type="hidden" name="id" value="<?= $os->id ?>"/>
    <div class="mb-2">
        <label for="tbEditOSName">Name:</label>
        <input type="text" name="name" id="tbEditOSName" class="form-
control" value="<?= $os->name ?>"/>
        <label class="font-small error-message text-
danger">OS name cannot contain special characters or numbers</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnEditOS" name="btnEditOS" class="py-1 px-3 rounded btn-primary font-
small">Edit OS</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
```

## views/pages/admin/brands/add-brand-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }
?>
<form action="models/admin/brands/add-brand.php" method="POST">
    <div class="mb-2">
        <label for="tbAddBrandName">Name:</label>
        <input type="text" name="name" id="tbAddBrandName" class="form-control"/>
        <label class="font-small error-message text-
danger">Brand name cannot contain special characters or numbers</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnAddBrand" name="btnAddBrand" class="py-1 px-3 rounded btn-
primary font-small">Add brand</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
```

## views/pages/admin/brands/edit-brand-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    if(!isset($_GET['id'])) {
        http_response_code(400);
        exit();
    }

    $brand = getBrand('id', $_GET['id']);
?>
<form action="models/admin/brands/edit-brand.php" method="POST">
    <input type="hidden" name="id" value="<?= $brand->id ?>" />
    <div class="mb-2">
        <label for="tbEditBrandName">Name:</label>
        <input type="text" name="name" id="tbEditBrandName" class="form-
control" value="<?= $brand->name ?>" />
        <label class="font-small error-message text-
danger">Brand name cannot contain special characters or numbers</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnEditBrand" name="btnEditBrand" class="py-1 px-3 rounded btn-
primary font-small">Edit Brand</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
```

## views/pages/admin/orders/edit-order-form.php

```
<?php
    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    if(!isset($_GET['id'])) {
        http_response_code(400);
        exit();
    }

    $order = getOrder('order_id', $_GET['id']);
?>
<form action="models/admin/orders/edit-order.php" method="POST">
    <input type="hidden" name="id" value="<?= $order->order_id ?>" />
    <div class="mb-2">
        <label for="tbEditOrderName">Name:</label>
        <input type="text" name="name" id="tbEditOrderName" class="form-
control" value="<?= $order->name ?>" />
        <label class="error-message">
            <ul class="font-small p-0">
                <li class="text-danger">- A name cannot have less than 3 characters</li>
                <li class="text-danger">- Has to be at least 2 words</li>
                <li class="text-danger">- Each word must be capitalized</li>
            </ul>
        </label>
    </div>
    <div class="mb-2">
        <label for="tbEditOrderAddress">Address:</label>
        <input type="text" name="address" id="tbEditOrderAddress" class="form-
control" value="<?= $order->address ?>" />
        <label class="font-small error-message text-
danger">[Street Name] [Home Number]/[Apartment Number]</label>
    </div>
    <div class="text-center mb-3">
        <button id="btnEditOrder" name="btnEditOrder" class="py-1 px-3 rounded btn-
primary font-small">Edit Order</button>
    </div>
    <?php
        require_once('models/forms/functions.php');
        displayMessages();
    ?>
</form>
```



## models/nav/functions.php

```
<?php
    function getNavLinks() {
        return executeQuery("SELECT * FROM page_links pl INNER JOIN link_categories lc ON
pl.link_category = lc.lc_id WHERE lc.name = 'nav'");
    }
?>
```

## models/slider/functions.php

```
<?php
    function getSliderImages() {
        return executeQuery("SELECT * FROM slider_images");
    }
?>
```

## models/footer/functions.php

```
<?php
    function getFooterLinks() {
        return executeQuery("SELECT * FROM footer_links");
    }
?>
```

## models/users/functions.php

```
<?php
function getUser($column, $value) {
    global $conn;
    $exec = $conn-
>prepare("SELECT * FROM users u INNER JOIN roles r ON u.role = r.role_id WHERE $column = ?
");
    $exec->execute([$value]);
    $result = $exec->fetch();
    return $result;
}

function registerUser($username, $email, $password) {
    try {
        $encPassword = md5($password);
        global $conn;
        $exec = $conn-
>prepare("INSERT INTO users(username, email, password, role) VALUES (?, ?, ?, 2)");
        $result = $exec->execute([$username, $email, $encPassword]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function updateLoginLog($userId, $result) {
    $file = fopen(LOGIN_LOG, 'a');
    if($file) {
        $ipAddress = $_SERVER['REMOTE_ADDR'];
        $text = $userId . SEPARATOR . $ipAddress . SEPARATOR . getDateAndTime() . SEPA
RATOR . $result . "\n";
        fwrite($file, $text);
        fclose($file);
    }
}

function getFailedAttempts($id, $minutes) {
    $attempts = 0;
    $currentTime = time();
    $file = file(LOGIN_LOG);
    foreach($file as $row) {
        $row = trim($row);
        $values = explode(SEPARATOR, $row);
        if(intval($values[0]) == $id) {
            $loginTime = getTime($values[2]);
            if($currentTime - $loginTime <= 60 * $minutes && $values[3] == '0') {
                $attempts++;
            }
        }
    }
}
```

```

        return $attempts;
    }

    function lockAccount($id) {
        try {
            global $conn;
            $exec = $conn->prepare("UPDATE users SET active = 0 WHERE id = ?");
            $result = $exec->execute([$id]);
            return $result;
        } catch (PDOException $ex) {
            updateErrorLog($ex->getMessage());
            return false;
        }
    }

    function sendEmail($user) {
        $to = $user->email;
        $subject = 'Your account has been locked';
        $message = "Your account <b>$user->
>username</b> has been locked due to suspicious activity. To unlock your account, contact
the website administrator.";
        $headers = "From: mobishop@noreply.com" . "\r\n" . "Content-
Type: text/html; charset=ISO-8859-1\r\n";

        return mail($to, $subject, $message, $headers);
    }
?>

```

## models/users/get-current-user.php

```
<?php
    session_start();
    $response = false;
    if(isset($_SESSION['user'])) {
        $response = $_SESSION['user'];
    } else {
        if(isset($_GET['message'])) {
            $_SESSION['errors'] = [$_GET['message']];
        }
    }
    http_response_code(200);
    echo(json_encode($response));
?>
```

## models/users/login.php

```
<?php
    if(isset($_POST['btnLogIn'])) {
        session_start();
        require_once("../../config/connection.php");
        require_once("functions.php");

        $email = $_POST['email'];
        $password = $_POST['password'];

        // regular expressions
        $regExpEmail = "/^[a-z][a-z0-9-_\\.]{2,}@([a-z0-9-_]{2,}\\.)+[a-z]{2,}$/";
        $regExpPassword = "/^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*])(?=.*{8,})/";

        // validation
        $errors = [];

        if(!preg_match($regExpEmail, $email)) {
            array_push($errors, "Email example: example@example.com;");
        }
        if(!preg_match($regExpPassword, $password)) {
            array_push($errors, "Your password must: contain 1 lowercase letter; contain 1 uppercase letter; contain 1 number; contain 1 special character; be 8 characters or longer;");
        }

        if(count($errors)) {
            $_SESSION['errors'] = $errors;
            http_response_code(400);
        } else {
            $user = getUser('email', $email);
            if(!$user) { // checks if user exists
                $_SESSION['errors'] = ["Invalid credentials."];
                http_response_code(409);
            } else {
                $encPassword = md5($password);
                if($user->password != $encPassword) { // checks if password is correct
                    $_SESSION['errors'] = ["Invalid credentials."];

                    updateLoginLog($user->id, 0);

                    if(getFailedAttempts($user->id, 5) >= 3) {
                        if(lockAccount($user->id)) sendEmail($user);
                    }

                    http_response_code(409);
                } else {
                    if($user->active) { // checks if account is active
                        $_SESSION['user'] = $user;
                        updateLoginLog($user->id, 1);
                    }
                }
            }
        }
    }
}
```

```

        http_response_code(200);
    } else {
        $errors = ["Your account is currently locked. Contact the website
administrator for more info."];
        $_SESSION['errors'] = $errors;
        updateLoginLog($user->id, 0);
        http_response_code(409);
    }
}

}

header('Location: ../../index.php?p=login');
exit();
} else {
    http_response_code(400);
}

?>

```



## models/users/register.php

```
<?php
if(isset($_POST['btnRegister'])) {
    session_start();
    require_once("../../config/connection.php");
    require_once("functions.php");

    $username = $_POST['username'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $repeatPassword = $_POST['repeatPassword'];

    // regular expressions
    $regExpUsername = "/^[a-zA-Z]\w{4,20}$/";
    $regExpEmail = "/^[a-z][a-z0-9-_\.\.]{2,}@[a-z0-9-_\.\.]{2,}\.[a-z]{2,}$/";
    $regExpPassword = "/^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%^&*])(?={8,})/";

    // validation
    $errors = [];

    if(!preg_match($regExpUsername, $username)) {
        array_push($errors, "Your username must: start with a letter; be between 5 and 20 characters long; not contain special characters other than '_'");
    }
    if(!preg_match($regExpEmail, $email)) {
        array_push($errors, "Email example: example@example.com;");
    }
    if(!preg_match($regExpPassword, $password)) {
        array_push($errors, "Your password must: contain 1 lowercase letter; contain 1 uppercase letter; contain 1 number; contain 1 special character; be 8 characters or longer;");
    }
    if($password != $repeatPassword) {
        array_push($errors, "Passwords don't match;");
    }

    if(count($errors)) {
        $_SESSION['errors'] = $errors;
        http_response_code(400);
    } else {
        if(getUser('email', $email)) {
            array_push($errors, "That email is taken;");
        }
        if(getUser('username', $username)) {
            array_push($errors, "That username is taken;");
        }
        if(count($errors)) {
            $_SESSION['errors'] = $errors;
            http_response_code(400);
        } else {
```

```
        if(registerUser($username, $email, $password)) {
            $_SESSION['messages'] = ["Your account was created successfully."];
            http_response_code(200);
        } else {
            $_SESSION['errors'] = ["We encountered an error. Contact the website administrator for more info."];
            http_response_code(500);
        }
    }
    header('Location: ../../index.php?p=register');
    exit();
} else {
    http_response_code(400);
}
?>
```

## models/users/logout.php

```
<?php
    session_start();
    if(isset($_SESSION['user'])) {
        unset($_SESSION['user']);
    }
    header("Location: ../../index.php?p=home");
    exit();
?>
```

## models/phones/functions.php

```
<?php
function getAllPhones() {
    return executeQuery("SELECT p.id AS phone_id, p.name AS phone_name, price, active,
b.name AS brand_name, os.name AS os_name FROM phones p INNER JOIN brands b ON p.brand = b
.id INNER JOIN operating_systems os ON p.os = os.id ORDER BY p.name");
}

function getBrands() {
    return executeQuery("SELECT * FROM brands ORDER BY name");
}

function getOperatingSystems() {
    return executeQuery("SELECT * FROM operating_systems ORDER BY name");
}

function getPhoneById($id) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM phones WHERE id = ?");
    $exec->execute([$id]);
    return $exec->fetch();
}

function getBrandById($id) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM brands WHERE id = ?");
    $exec->execute([$id]);
    return $exec->fetch();
}

function getOSById($id) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM operating_systems WHERE id = ?");
    $exec->execute([$id]);
    return $exec->fetch();
}

?>
```

## models/phones/get-phones.php

```
<?php
require_once("../../config/connection.php");
// creating the query
$query = "SELECT * FROM phones WHERE active = 1";
$params = [];
if(isset($_GET['search'])) {
    $query .= " AND name LIKE ?";
    $search = "%" . $_GET['search'] . "%";
    array_push($params, $search);
}
if(isset($_GET['os'])) {
    $placeholders = implode(',', array_fill(0, count($_GET['os']), '?'));
    $query .= " AND os IN ($placeholders)";
    foreach($_GET['os'] as $os) {
        array_push($params, $os);
    }
}
if(isset($_GET['brand'])) {
    $placeholders = implode(',', array_fill(0, count($_GET['brand']), '?'));
    $query .= " AND brand IN ($placeholders)";
    foreach($_GET['brand'] as $brand) {
        array_push($params, $brand);
    }
}
if(isset($_GET['maxPrice'])) {
    if(preg_match("/^[0-9]+(\.[0-9]*)?$/", $_GET['maxPrice'])) {
        $query .= " AND price <= ?";
        $maxPrice = floatval($_GET['maxPrice']);
        array_push($params, $maxPrice);
    }
}
$orderValues = ['1', '2', '3', '4'];
if(isset($_GET['order'])) {
    $order = $_GET['order'];
    if(in_array($order, $orderValues)) {
        $text = '';
        switch($order) {
            case '1':
                $text = "name";
                break;
            case '2':
                $text = "name DESC";
                break;
            case '3':
                $text = "price";
                break;
            case '4':
                $text = "price DESC";
                break;
        }
    }
}
```

```

        $query .= " ORDER BY $text";
    }
}
// executing the query
global $conn;
$exec = $conn->prepare($query);
$exec->execute($params);
$result = $exec->fetchAll();
$numberOfPhones = count($result);
// getting only 6 phones
$phonesPerPage = $_GET['phonesPerPage'];
$page = ($_GET['page'] - 1) * $phonesPerPage;
$phones = [];
for($i = $page; $i < $phonesPerPage + $page; $i++) {
    if($i >= count($result)) break;
    array_push($phones, $result[$i]);
}
// response
$response = ['phones' => $phones, 'numberOfPhones' => $numberOfPhones];
http_response_code(200);
header("Content-type: application/json");
echo(json_encode($response));
?>

```

## models/contact/functions.php

```
<?php
function sendMessage($email, $name, $message) {
    $userId = NULL;
    if(isset($_SESSION['user'])) {
        $userId = $_SESSION['user']->id;
    }
    $message = addslashes($message);
    global $conn;
    $exec = $conn->prepare("INSERT INTO contact_messages(user_id, email, name, message) VALUES (?, ?, ?, ?)"
);
    $result = $exec->execute([$userId, $email, $name, $message]);
    return $result;
}
?>
```

## models/contact/send-message.php

```
<?php
    if(isset($_POST['btnSend'])) {
        session_start();
        require_once("../../config/connection.php");
        require_once("functions.php");

        $email = $_POST['email'];
        $name = $_POST['name'];
        $message = $_POST['message'];

        $errors = [];

        // regular expressions
        $regExpName = "/^[A-ZŠĐČĆŽ][a-zšđćčž]{2,}(\s[A-ZŠĐČĆŽ][a-zšđćčž]{2,})*/";
        $regExpEmail = "/^[a-z][a-z0-9\-\_\.\.]{2,}@([a-z0-9\-\_\.\.]{2,}\.)+[a-z]{2,}$/";

        // validation
        if(!preg_match($regExpEmail, $email)) {
            array_push($errors, "Email example: example@example.com;");
        }
        if(!preg_match($regExpName, $name)) {
            array_push($errors, "A name cannot have less than 3 characters; Each word of your name must be capitalized;");
        }
        $numberOfSpaces = substr_count($message, ' ');
        if(strlen($message) - $numberOfSpaces < 20 || strlen($message) > 500) {
            array_push($errors, "Your message (without spacing characters) cannot be shorter than 20 characters or longer than 500 characters in total;");
        }

        if(!count($errors)) {
            try {
                if(sendMessage($email, $name, $message)) {
                    $_SESSION['messages'] = ["Your message was sent successfully."];
                    http_response_code(200);
                } else {
                    $_SESSION['errors'] = "We encountered an error. Please try again later.";
                    http_response_code(500);
                }
            } catch (PDOException $ex) {
                $_SESSION['errors'] = ["We encountered an error. Contact the website administrator for more info."];
                updateErrorLog($ex->getMessage());
                http_response_code(500);
            }
        } else {
            $_SESSION['errors'] = $errors;
            http_response_code(400);
        }
    }
}
```



```
        header("Location: ../../index.php?p=contact");  
        exit();  
    } else {  
        http_response_code(400);  
    }  
?>
```

## models/author/functions.php

```
<?php
    function getAuthorSocials() {
        return executeQuery("SELECT * FROM author_socials");
    }
?>
```

## models/author/export-to-word.php

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

    // creating the file
    $word = new COM("word.application") or die("Unable to instantiate Word.");

    $word->Visible = 1;

    $word->Documents->Add();

    $word->Selection->TypeText("My name is Bojan and I am an aspiring front-
end web developer. My approach to website design is to create a website that strengthens y
our company's brand while ensuring ease of use and simplicity for your audience.\nThe way
I look at it, a front-
end developer's role is to combine design and business logic to achieve a user-
facing product. To do this successfully, a wide skill set is necessary to produce a qualit
y user experience that leads to meeting business goals, and I guarantee I've got exactly w
hat's needed.");

    $location = ABSOLUTE_PATH . "data/author.doc";
    $word->Documents[1]->SaveAs($location);

    $word->Quit();

    $word = null;

    // download
    $file = file_get_contents($location);
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;filename=author.doc");
    echo($file);

    // delete
    if(file_exists($location)) {
        unlink($location);
    }
?>
```

## models/cart/functions.php

```
<?php
function addToCart($userId, $phoneId, $quantity) {
    global $conn;
    $cart = alreadyIsInCart($userId, $phoneId);
    try {
        if($cart) {
            $quantity += $cart->quantity;
            $exec = $conn->prepare("UPDATE cart SET quantity = ? WHERE id = ?");
            $result = $exec->execute([$quantity, $cart->id]);
        } else {
            $exec = $conn->
>prepare("INSERT INTO cart(user_id, phone_id, quantity) VALUES (?, ?, ?)");
            $result = $exec->execute([$userId, $phoneId, $quantity]);
        }
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function alreadyIsInCart($userId, $phoneId) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM cart WHERE user_id = ? AND phone_id = ?");
    $exec->execute([$userId, $phoneId]);
    return $exec->fetch();
}

function getCartQuantity($userId) {
    global $conn;
    $exec = $conn->
>prepare("SELECT IFNULL(SUM(quantity), 0) as quantity FROM cart WHERE user_id = ?");
    $exec->execute([$userId]);
    $result = $exec->fetch();
    return $result->quantity;
}

function getCart($userId) {
    global $conn;
    $exec = $conn->
>prepare("SELECT c.id, c.quantity, p.name, p.price, p.id AS phone_id FROM cart c INNER JOIN phones p ON c.phone_id = p.id WHERE c.user_id = ? AND p.active = 1");
    $exec->execute([$userId]);
    $result = $exec->fetchAll();
    return $result;
}

function changeQuantity($id, $change) {
    $quantity = getItemQuantity($id);
    if(!$quantity) return false;
```

```

        switch($change) {
            case 'increase':
                $quantity++;
                break;
            case 'decrease':
                if($quantity > 1) $quantity--;
                break;
        }
        try {
            global $conn;
            $exec = $conn->prepare("UPDATE cart SET quantity = ? WHERE id = ?");
            $result = $exec->execute([$quantity, $id]);
            return $result;
        } catch (PDOException $ex) {
            updateErrorLog($ex->getMessage());
            return false;
        }
    }

function getItemQuantity($id) {
    global $conn;
    $exec = $conn->prepare("SELECT quantity FROM cart WHERE id = ?");
    $exec->execute([$id]);
    $result = $exec->fetch();
    if($result) return $result->quantity;
    return false;
}

function removeItem($id) {
    try {
        global $conn;
        $exec = $conn->prepare("DELETE FROM cart WHERE id = ?");
        $result = $exec->execute([$id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function createOrder($userId, $name, $address, $cart) {
    $totalPrice = 0;
    foreach($cart as $c) {
        $totalPrice += $c->quantity * $c->price;
    }

    try {
        global $conn;
        $conn->beginTransaction();
        $exec = $conn->
>prepare("INSERT INTO orders(user_id, total_price, name, address) VALUES (?, ?, ?, ?)");
        $exec->execute([$userId, $totalPrice, $name, $address]);
    }
}

```

```

        $orderId = $conn->lastInsertId();
        foreach($cart as $c) {
            $exec = $conn-
>prepare("INSERT INTO order_details(order_id, phone_id, quantity) VALUES (?, ?, ?)");
            $exec->execute([$orderId, $c->phone_id, $c->quantity]);
        }
        $conn->commit();
        clearCart($userId);
        return true;
    } catch (PDOException $ex) {
        $conn->rollBack();
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function clearCart($userId) {
    try {
        global $conn;
        $exec = $conn->prepare("DELETE FROM cart WHERE user_id = ?");
        $result = $exec->execute([$userId]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

?>

```

## models/cart/add-to-cart.php

```
<?php
    if($_SERVER['REQUEST_METHOD'] == 'POST') {
        require_once(' ../../config/connection.php');
        require_once('functions.php');

        $userId = $_POST['userId'];
        $phoneId = $_POST['phoneId'];
        $quantity = $_POST['quantity'];

        if(addToCart($userId, $phoneId, $quantity)) {
            http_response_code(200);
            echo(json_encode('Item successfully added to cart.'));
        } else {
            http_response_code(500);
            echo(json_encode('We encountered an error.'));
        }
    } else {
        http_response_code(400);
    }
?>
```

## models/cart/change-quantity.php

```
<?php
    if($_SERVER['REQUEST_METHOD'] == "POST") {
        require_once(' ../../config/connection.php');
        require_once('functions.php');

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

        $changeValues = ['increase', 'decrease'];
        if(!in_array($change, $changeValues)) {
            http_response_code(400);
            exit();
        }

        if(changeQuantity($id, $change)) {
            http_response_code(200);
            echo(json_encode(1));
        } else {
            http_response_code(500);
            echo('We encountered an error.');
```



## models/cart/create-order.php

```
<?php
session_start();
if(isset($_POST['btnCheckout'])) {
    require_once('../../config/connection.php');
    require_once('functions.php');

    $userId = $_SESSION['user']->id;
    $name = $_POST['tbName'];
    $address = $_POST['tbAddress'];

    // regular expressions
    $regExpName = "/^[A-ZŠĐČŽ][a-zšđčž]{2,}(\s[A-ZŠĐČŽ][a-zšđčž]{2,})+$/";
    $regExpAddress = "/^[A-ZŠĐČŽ][a-zšđčž]{2,}(\s[A-ZŠĐČŽa-zšđčž][a-
zšđčž]{2,})*\s\d+[A-Z]?(\s\d+)*$/";

    // validation
    $errors = [];

    if(!preg_match($regExpName, $name)) {
        array_push($errors, "A name cannot have less than 3 characters; Has to be at l
east 2 words; Each word must be capitalized;");
    }
    if(!preg_match($regExpAddress, $address)) {
        array_push($errors, "Address example: [Street Name] [Home Number]/[Apartment N
umber];");
    }

    if(count($errors)) {
        $_SESSION['errors'] = $errors;
        http_response_code(400);
    } else {
        $cart = getCart($userId);
        if(!count($cart)) {
            $_SESSION['errors'] = ['Your cart is empty.'];
            http_response_code(409);
        } else {
            if(createOrder($userId, $name, $address, $cart)) {
                $_SESSION['messages'] = ['Your order was created successfully.'];
                http_response_code(200);
            } else {
                $_SESSION['errors'] = ['We encountered an error.'];
                http_response_code(500);
            }
        }
    }

    header("Location: ../../index.php?p=cart");
    exit();
} else {
    http_response_code(400);
}
```

```
}  
?>
```

## models/cart/get-cart-quantity.php

```
<?php
    session_start();
    $response = 0;
    if(isset($_SESSION['user'])) {
        require_once(' ../../config/connection.php');
        require_once('functions.php');
        $response = getCartQuantity($_SESSION['user']->id);
    }
    http_response_code(200);
    echo(json_encode($response));
?>
```

## models/cart/get-cart.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        require_once(' ../../config/connection.php');
        require_once('functions.php');

        $userId = $_SESSION['user']->id;
        $cart = getCart($userId);

        http_response_code(200);
        echo(json_encode($cart));
    } else {
        http_response_code(400);
    }
?>
```

## models/cart/remove-item.php

```
<?php
    if($_SERVER['REQUEST_METHOD'] == "POST") {
        require_once(' ../../config/connection.php');
        require_once('functions.php');
        $id = $_POST['id'];
        if(removeItem($id)) {
            http_response_code(200);
            echo(json_encode(1));
        } else {
            http_response_code(500);
            echo('We encountered an error. ');
        }
    } else {
        http_response_code(400);
    }
?>
```

## models/forms/functions.php

```
<?php
function displayMessages() {
    $text = "";
    if(isset($_SESSION['messages'])) {
        $text .= '<div class="alert alert-info">';
        foreach($_SESSION['messages'] as $key=>$m) {
            if($key > 0) $text .= '<br/>';
            $text .= "<p class='text-info text-center m-0'>$m</p>";
        }
        $text .= '</div>';
        unset($_SESSION['messages']);
    }
    if(isset($_SESSION['errors'])) {
        $text .= '<div class="alert alert-danger">';
        foreach($_SESSION['errors'] as $key=>$e) {
            if($key > 0) $text .= '<br/>';
            $text .= "<p class='text-danger text-center m-0'>$e</p>";
        }
        $text .= '</div>';
        unset($_SESSION['errors']);
    }
    echo($text);
}
?>
```

## models/admin/functions.php

```
<?php
function getAdminLinks() {
    return executeQuery("SELECT * FROM page_links pl INNER JOIN link_categories lc ON
pl.link_category = lc.lc_id WHERE lc.name = 'admin'");
}

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

function getAllBrands() {
    return executeQuery("SELECT * FROM brands");
}

function getAllOS() {
    return executeQuery("SELECT * FROM operating_systems");
}

function getAllOrders() {
    return executeQuery("SELECT * FROM orders o LEFT OUTER JOIN users u ON o.user_id =
u.id");
}

function getAllMessages() {
    return executeQuery("SELECT message_id, u.username, cm.name, cm.email, message FRO
M contact_messages cm LEFT OUTER JOIN users u ON u.id = cm.user_id");
}

function deleteMessage($id) {
    try {
        global $conn;
        $exec = $conn->prepare("DELETE FROM contact_messages WHERE message_id = ?");
        $result = $exec->execute([$id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function deleteOrder($id) {
    try {
        global $conn;
        $conn->beginTransaction();
        $exec = $conn->prepare("DELETE FROM order_details WHERE order_id = ?");
        $result = $exec->execute([$id]);
        if($result) {
            $exec = $conn->prepare("DELETE FROM orders WHERE order_id = ?");
            $result = $exec->execute([$id]);
        }
    }
}
```

```

        if($result) {
            $conn->commit();
        }
    }
    return $result;
} catch (PDOException $ex) {
    $conn->rollBack();
    updateErrorLog($ex->getMessage());
    return false;
}
}

function deleteOS($id) {
    try {
        global $conn;
        $exec = $conn->prepare("SELECT * FROM phones WHERE os = ?");
        $exec->execute([$id]);
        $phones = $exec->fetchAll();
        if(count($phones)) {
            return false;
        } else {
            $exec = $conn->prepare("DELETE FROM operating_systems WHERE id = ?");
            $result = $exec->execute([$id]);
        }
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function deleteBrand($id) {
    try {
        global $conn;
        $exec = $conn->prepare("SELECT * FROM phones WHERE brand = ?");
        $exec->execute([$id]);
        $phones = $exec->fetchAll();
        if(count($phones)) {
            return false;
        } else {
            $exec = $conn->prepare("DELETE FROM brands WHERE id = ?");
            $result = $exec->execute([$id]);
        }
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function deletePhone($id) {
    try {

```



```

        global $conn;
        $exec = $conn->prepare("SELECT * FROM order_details WHERE phone_id = ?");
        $exec->execute([$id]);
        $orders = $exec->fetchAll();
        if(count($orders)) {
            return false;
        } else {
            $conn->beginTransaction();
            $exec = $conn->prepare("DELETE FROM cart WHERE phone_id = ?");
            $result = $exec->execute([$id]);
            if($result) {
                $exec = $conn->prepare("DELETE FROM phones WHERE id = ?");
                $result = $exec->execute([$id]);
                if($result) {
                    $conn->commit();
                }
            }
        }
        return $result;
    } catch (PDOException $ex) {
        $conn->rollBack();
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function deleteUser($id) {
    if($_SESSION['user']->id == $id) return false;
    try {
        global $conn;
        $conn->beginTransaction();
        $exec = $conn->prepare("DELETE FROM cart WHERE user_id = ?");
        $result = $exec->execute([$id]);
        if($result) {
            $exec = $conn->
>prepare("UPDATE orders SET user_id = NULL WHERE user_id = ?");
            $result = $exec->execute([$id]);
            if($result) {
                $exec = $conn->
>prepare("UPDATE contact_messages SET user_id = NULL WHERE user_id = ?");
                $result = $exec->execute([$id]);
                if($result) {
                    $exec = $conn->prepare("DELETE FROM users WHERE id = ?");
                    $result = $exec->execute([$id]);
                    if($result) {
                        $conn->commit();
                    }
                }
            }
        }
    }
    return $result;
} catch (PDOException $ex) {

```

```

        $conn->rollBack();
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function addBrand($name) {
    try {
        global $conn;
        $exec = $conn->prepare("INSERT INTO brands(name) VALUES (?)");
        $result = $exec->execute([$name]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function getBrand($column, $value) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM brands WHERE $column = ?");
    $exec->execute([$value]);
    return $exec->fetch();
}

function editBrand($id, $name) {
    try {
        global $conn;
        $exec = $conn->prepare("UPDATE brands SET name = ? WHERE id = ?");
        $result = $exec->execute([$name, $id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function addOS($name) {
    try {
        global $conn;
        $exec = $conn->prepare("INSERT INTO operating_systems(name) VALUES (?)");
        $result = $exec->execute([$name]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function getOS($column, $value) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM operating_systems WHERE $column = ?");

```

```

        $exec->execute([$value]);
        return $exec->fetch();
    }

    function editOS($id, $name) {
        try {
            global $conn;
            $exec = $conn->prepare("UPDATE operating_systems SET name = ? WHERE id = ?");
            $result = $exec->execute([$name, $id]);
            return $result;
        } catch (PDOException $ex) {
            updateErrorLog($ex->getMessage());
            return false;
        }
    }

    function getOrder($column, $value) {
        global $conn;
        $exec = $conn->prepare("SELECT * FROM orders WHERE $column = ?");
        $exec->execute([$value]);
        return $exec->fetch();
    }

    function editOrder($id, $name, $address) {
        try {
            global $conn;
            $exec = $conn->
>prepare("UPDATE orders SET name = ?, address = ? WHERE order_id = ?");
            $result = $exec->execute([$name, $address, $id]);
            return $result;
        } catch (PDOException $ex) {
            updateErrorLog($ex->getMessage());
            return false;
        }
    }

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

    function addUser($username, $email, $encPassword, $role) {
        try {
            global $conn;
            $exec = $conn->
>prepare("INSERT INTO users(username, email, password, role) VALUES (?, ?, ?, ?)");
            $result = $exec->execute([$username, $email, $encPassword, $role]);
            return $result;
        } catch (PDOException $ex) {
            updateErrorLog($ex->getMessage());
            return false;
        }
    }
}

```

```

function editUserWithPassword($id, $username, $email, $encPassword, $role, $active) {
    try {
        global $conn;
        $exec = $conn-
>prepare("UPDATE users SET username = ?, email = ?, password = ?, role = ?, active = ? WHERE id = ?");
        $result = $exec-
>execute([$username, $email, $encPassword, $role, $active, $id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function editUserWithoutPassword($id, $username, $email, $role, $active) {
    try {
        global $conn;
        $exec = $conn-
>prepare("UPDATE users SET username = ?, email = ?, role = ?, active = ? WHERE id = ?");
        $result = $exec->execute([$username, $email, $role, $active, $id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function getPhone($column, $value) {
    global $conn;
    $exec = $conn->prepare("SELECT * FROM phones WHERE $column = ?");
    $exec->execute([$value]);
    return $exec->fetch();
}

function addPhone($name, $price, $os, $brand, $originalImageLocation, $smallImageLocation) {
    try {
        global $conn;
        $exec = $conn-
>prepare("INSERT INTO phones(name, price, os, brand, image, image_small) VALUES (?, ?, ?, ?, ?, ?)");
        $result = $exec-
>execute([$name, $price, $os, $brand, $originalImageLocation, $smallImageLocation]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

```

```

function editPhoneWithImage($id, $name, $price, $os, $brand, $active, $originalImageName, $smallImageName) {
    try {
        global $conn;
        $exec = $conn->prepare("UPDATE phones SET name = ?, price = ?, os = ?, brand = ?, active = ?, image = ?, image_small = ? WHERE id = ?");
        $result = $exec->execute([$name, $price, $os, $brand, $active, $originalImageName, $smallImageName, $id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}

function editPhoneWithoutImage($id, $name, $price, $os, $brand, $active) {
    try {
        global $conn;
        $exec = $conn->prepare("UPDATE phones SET name = ?, price = ?, os = ?, brand = ?, active = ? WHERE id = ?");
        $result = $exec->execute([$name, $price, $os, $brand, $active, $id]);
        return $result;
    } catch (PDOException $ex) {
        updateErrorLog($ex->getMessage());
        return false;
    }
}
?>

```

## models/admin/delete.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

    require_once('.../config/connection.php');
    require_once('functions.php');

    if(isset($_GET['id']) && isset($_GET['ap'])) {
        $id = $_GET['id'];
        $adminPage = $_GET['ap'];
        $result = false;

        switch($adminPage) {
            case 'users':
                $result = deleteUser($id);
                $message = ["User was deleted successfully."];
                break;
            case 'phones':
                $result = deletePhone($id);
                $message = ["Phone was deleted successfully."];
                break;
            case 'brands':
                $result = deleteBrand($id);
                $message = ["Brand was deleted successfully."];
                break;
            case 'operating-systems':
                $result = deleteOS($id);
                $message = ["OS was deleted successfully."];
                break;
            case 'orders':
                $result = deleteOrder($id);
                $message = ["Order was deleted successfully."];
                break;
            case 'messages':
                $result = deleteMessage($id);
                $message = ["Message was deleted successfully."];
                break;
        }

        if($result) {
            http_response_code(200);
        }
    }
}
```

```
        $_SESSION['messages'] = $message;
    } else {
        http_response_code(400);
        $_SESSION['errors'] = ["We encountered an error."];
    }

    header("Location: ../../index.php?p=admin&ap=$adminPage");
    exit();
} else {
    http_response_code(400);
}

?>
```

## models/admin/users/add-user.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

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

        $errors = [];
        $regExpUsername = "/^[a-zA-Z]\w{4,20}$/";
        $regExpEmail = "/^[a-z][a-z0-9-_\.\.]{2,}@[a-z0-9-_\.\.]{2,}\.com$/";
        $regExpPassword = "/^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%^&*])(?=.{8,})/";

        // validation
        if(!preg_match($regExpUsername, $username)) {
            array_push($errors, "Your username must: start with a letter; be between 5 and 20 characters long; not contain special characters other than '_';");
        }
        if(!preg_match($regExpEmail, $email)) {
            array_push($errors, "Example email: examplename@example.com;");
        }
        if(!preg_match($regExpPassword, $password)) {
            array_push($errors, "Your password must: contain 1 lowercase letter; contain 1 uppercase letter; contain 1 number; contain 1 special character; be 8 characters or longer;");
        }

        $allRole = getAllRoles();
        $roleExists = false;
        foreach($allRole as $r) {
            if($r->role_id == $role) {
                $roleExists = true;
                break;
            }
        }
    }
}
```



```

    if(!$roleExists) {
        array_push($errors, "That role doesn't exist;");
    }
    // validation

    if(count($errors)) {
        http_response_code(400);
        $_SESSION['errors'] = $errors;
    } else {
        require_once(' ../../users/functions.php');
        if(getUser('username', $username)) {
            array_push($errors, "That username is taken;");
        }
        if(getUser('email', $email)) {
            array_push($errors, "That email is taken;");
        }
        if(count($errors)) {
            http_response_code(409);
            $_SESSION['errors'] = $errors;
        } else {
            $encPassword = md5($password);
            if(addUser($username, $email, $encPassword, $role)) {
                http_response_code(200);
                $_SESSION['messages'] = ["User '$username' was successfully created."];
            } else {
                http_response_code(500);
                $_SESSION['errors'] = ["We encountered an error."];
            }
        }
    }

    header("Location: ../../../../index.php?p=admin&ap=add-user");
    exit();
} else {
    http_response_code(400);
}
?>

```

## models/admin/users/edit-user.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $id = $_POST['id'];
        $username = $_POST['username'];
        $email = $_POST['email'];
        $role = $_POST['role'];
        $active = intval($_POST['active']);
        $password = '';
        var_dump($active);
        if(isset($_POST['chbChangePassword'])) {
            $password = $_POST['password'];
        }

        $errors = [];
        $regExpUsername = "/^[a-zA-Z]\w{4,20}$/";
        $regExpEmail = "/^[a-z][a-z0-9-_\.] {2,} @([a-z0-9-_\.] {2,} \. )+[a-z]{2,}$/";
        $regExpPassword = "/^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*])(?=. {8,})/";

        // validation
        if(!preg_match($regExpUsername, $username)) {
            array_push($errors, "Your username must: start with a letter; be between 5 and 20 characters long; not contain special characters other than '_';");
        }
        if(!preg_match($regExpEmail, $email)) {
            array_push($errors, "Example email: example@example.com;");
        }
        if(isset($_POST['chbChangePassword']) && !preg_match($regExpPassword, $password))
        {
            array_push($errors, "Your password must: contain 1 lowercase letter; contain 1 uppercase letter; contain 1 number; contain 1 special character; be 8 characters or longer;");
        }

        $allRole = getAllRoles();
```

```

$roleExists = false;
foreach($allRole as $r) {
    if($r->role_id = $role) {
        $roleExists = true;
        break;
    }
}

if(!$roleExists) {
    array_push($errors, "That role doesn't exist;");
}

$activeValues = [0, 1];
if(!in_array($active, $activeValues)) {
    array_push($errors, "Active value can either be 0 or 1;");
}
// validation

if(count($errors)) {
    http_response_code(400);
    $_SESSION['errors'] = $errors;
} else {
    require_once('../../users/functions.php');
    $userExists = getUser('username', $username);
    if($userExists) {
        if($userExists->id != $id) {
            array_push($errors, "That username is taken;");
        }
    }
    $userExists = getUser('email', $email);
    if($userExists) {
        if($userExists->id != $id) {
            array_push($errors, "That email is taken;");
        }
    }
    if(count($errors)) {
        http_response_code(409);
        $_SESSION['errors'] = $errors;
    } else {
        $encPassword = md5($password);
        if(isset($_POST['chbChangePassword'])) {
            $result = editUserWithPassword($id, $username, $email, $encPassword, $
role, $active);
        } else {
            $result = editUserWithoutPassword($id, $username, $email, $role, $acti
ve);
        }
        if($result) {
            http_response_code(200);
            $_SESSION['messages'] = ["User was edited successfully."];
        } else {
            http_response_code(500);

```

```
        $_SESSION['errors'] = ["We encountered an error."];
    }
}

header("Location: ../../../../index.php?p=admin&ap=edit-user&id=$id");
exit();
} else {
    http_response_code(400);
}
?>
```

## models/admin/users/export-to-excel.php

```
<?php
require_once(' ../../../../config/connection.php');
require_once(' ../../functions.php');

$excel = new COM("Excel.Application") or Die ("Konekcija sa Excel-
om nije bila uspešna!");
$Workbook = $excel->Workbooks->Add();
$Worksheet = $Workbook->Worksheets('Sheet1');

$field = $Worksheet->Range("A1");
$field->activate;
$field->Value = "ID:";

$field = $Worksheet->Range("B1");
$field->activate;
$field->Value = "Username:";

$field = $Worksheet->Range("C1");
$field->activate;
$field->Value = "Email:";

$field = $Worksheet->Range("D1");
$field->activate;
$field->Value = "Role:";

$field = $Worksheet->Range("E1");
$field->activate;
$field->Value = "Active:";

$users = getAllUsers();
foreach($users as $key=>$u) {
    $rowNumber = $key + 2;

    $field = $Worksheet->Range("A$rowNumber");
    $field->activate;
    $field->Value = $u->id;

    $field = $Worksheet->Range("B$rowNumber");
    $field->activate;
    $field->Value = $u->username;

    $field = $Worksheet->Range("C$rowNumber");
    $field->activate;
    $field->Value = $u->email;

    $field = $Worksheet->Range("D$rowNumber");
    $field->activate;
    $field->Value = $u->role_name;

    $field = $Worksheet->Range("E$rowNumber");
```

```

        $field->activate;
        $field->Value = $u->active;
    }

    $location = ABSOLUTE_PATH . "data/users.xls";
    $location = str_replace("/", "\\", $location);
    $Workbook->_SaveAs($location, -4143);

    $Workbook->Save();
    $Workbook->Saved = true;
    $Workbook->Close;

    unset($Workbook);

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

    unset($excel);

    // download
    $file = file_get_contents($location);
    header("Content-type: application/vnd.ms-excel");
    header("Content-Disposition: attachment;filename=users.xls");
    echo($file);

    // delete
    if(file_exists($location)) {
        unlink($location);
    }
?>

```

## models/admin/phones/add-phone.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $name = $_POST['name'];
        $price = $_POST['price'];
        $os = $_POST['os'];
        $brand = $_POST['brand'];
        $image = $_FILES['image'];

        $errors = [];
        $regExpPhoneName = "/^[A-Za-z0-9\\-\\_\\s]+$/";
        $regExpPrice = "/^[0-9]+(\\.[0-9]*)?$/";

        // validation
        if(!preg_match($regExpPhoneName, $name)) {
            array_push($errors, "Phone name cannot contain special characters other than '
- ' and '/';");
        }
        if(!preg_match($regExpPrice, $price)) {
            array_push($errors, "You must enter a price;");
        } else {
            $price = floatval($price);
        }

        $allowedTypes = ['image/jpeg', 'image/png', 'image/gif'];
        if(!in_array($image['type'], $allowedTypes)) {
            array_push($errors, "Uploaded file must be an image;");
        }

        $alloS = getAllOS();
        $osExists = false;
        foreach($alloS as $o) {
            if($o->id == $os) $osExists = true;
        }
        if(!$osExists) {
            array_push($errors, "That OS doesn't exist;");
        }
    }
}
```

```

}

$allBrands = getAllBrands();
$brandExists = false;
foreach($allBrands as $b) {
    if($b->id == $brand) $brandExists = true;
}
if(!$brandExists) {
    array_push($errors, "That brand doesn't exist;");
}

if(getPhone('name', $name)) {
    array_push($errors, "Phone with that name already exists;");
}
// validation

if(count($errors)) {
    http_response_code(400);
    $_SESSION['errors'] = $errors;
} else {
    // image upload
    // original image
    define('LOCATION', 'assets/img/phones/');
    $originalImageName = time() . '_' . $image['name'];
    $originalImageLocation = LOCATION . $originalImageName;
    $originalImageAbsolutePath = ABSOLUTE_PATH . $originalImageLocation;
    // original image
    if(move_uploaded_file($image['tmp_name'], $originalImageAbsolutePath)) {
        // small image
        list($width, $height) = getimagesize($originalImageAbsolutePath);
        $newWidth = 350;
        $newHeight = $height * ($newWidth / $width);
        $smallImage = imagecreatetruecolor($newWidth, $newHeight);
        $transparent = imagecolorallocatealpha($smallImage, 0, 0, 0, 127);
        imagefill($smallImage, 0, 0, $transparent);
        imagesavealpha($smallImage, true);
        switch($image['type']) {
            case 'image/jpeg':
                $originalImage = imagecreatefromjpeg($originalImageAbsolutePath);
                break;
            case 'image/png':
                $originalImage = imagecreatefrompng($originalImageAbsolutePath);
                break;
            case 'image/gif':
                $originalImage = imagecreatefromgif($originalImageAbsolutePath);
                break;
        }
        imagecopyresampled($smallImage, $originalImage, 0, 0, 0, 0, $newWidth, $newHeight, $width, $height);
    }
}

```



```

$smallImageName = "small_" . $originalImageName;
$smallImageLocation = LOCATION . $smallImageName;
$smallImageAbsolutePath = ABSOLUTE_PATH . $smallImageLocation;
switch($image['type']) {
    case 'image/jpeg':
        $result = imagejpeg($smallImage, $smallImageAbsolutePath);
        break;
    case 'image/png':
        $result = imagepng($smallImage, $smallImageAbsolutePath);
        break;
    case 'image/gif':
        $result = imagegif($smallImage, $smallImageAbsolutePath);
        break;
}
// small image
// image upload
if($result) {
    if(addPhone($name, $price, $os, $brand, $originalImageName, $smallImageName)) {
        http_response_code(200);
        $_SESSION['messages'] = ["Phone '$name' was successfully added."];
    } else {
        http_response_code(500);
        $_SESSION['errors'] = ["We encountered an error."];
    }
}
} else {
    http_response_code(500);
    $_SESSION['errors'] = ["We encountered an error."];
}
}

header("Location: ../../../../index.php?p=admin&ap=add-phone");
exit();
} else {
    http_response_code(400);
}
}
?>

```

## models/admin/phones/edit-phone.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $id = $_POST['id'];
        $name = $_POST['name'];
        $price = $_POST['price'];
        $os = $_POST['os'];
        $brand = $_POST['brand'];
        $active = $_POST['active'];
        if(isset($_POST['chbChangeImage'])) {
            $image = $_FILES['image'];
        }

        $errors = [];
        $regExpPhoneName = "/^[A-Za-z0-9\\/-\\s]+$/" ;
        $regExpPrice = "/^[0-9]+(\\.[0-9]*)?$/";

        // validation
        if(!preg_match($regExpPhoneName, $name)) {
            array_push($errors, "Phone name cannot contain special characters other than '
- ' and '/'");
        }
        if(!preg_match($regExpPrice, $price)) {
            array_push($errors, "You must enter a price;");
        } else {
            $price = floatval($price);
        }

        if(isset($_POST['chbChangeImage'])) {
            $allowedTypes = ['image/jpeg', 'image/png', 'image/gif'];
            if(!in_array($image['type'], $allowedTypes)) {
                array_push($errors, "Uploaded file must be an image;");
            }
        }

        $allOS = getAllOS();
```

```

$osExists = false;
foreach($allOS as $o) {
    if($o->id == $os) $osExists = true;
}
if(!$osExists) {
    array_push($errors, "That OS doesn't exist;");
}

$allBrands = getAllBrands();
$brandExists = false;
foreach($allBrands as $b) {
    if($b->id == $brand) $brandExists = true;
}
if(!$brandExists) {
    array_push($errors, "That brand doesn't exist;");
}

$phoneExists = getPhone('name', $name);
if($phoneExists) {
    if($phoneExists->id != $id) {
        array_push($errors, "Phone with that name already exists;");
    }
}

$activeValues = [0, 1];
if(!in_array($active, $activeValues)) {
    array_push($errors, "Active value can either be 0 or 1;");
}
// validation

if(count($errors)) {
    http_response_code(400);
    $_SESSION['errors'] = $errors;
} else {
    $continue = true;
    if(isset($_POST['chbChangeImage'])) {
        $continue = false;
        // image upload
        // original image
        define('LOCATION', 'assets/img/phones/');
        $originalImageName = time() . '_' . $image['name'];
        $originalImageLocation = LOCATION . $originalImageName;
        $originalImageAbsolutePath = ABSOLUTE_PATH . $originalImageLocation;
        // original image
        if(move_uploaded_file($image['tmp_name'], $originalImageAbsolutePath))
        {
            // small image
            list($width, $height) = getimagesize($originalImageAbsolutePath);
            $newWidth = 350;
            $newHeight = $height * ($newWidth / $width);
            $smallImage = imagecreatetruecolor($newWidth, $newHeight);
            $transparent = imagecolorallocatealpha($smallImage, 0, 0, 0, 127);

```

```

        imagefill($smallImage, 0, 0, $transparent);
        imagesavealpha($smallImage, true);
        switch($image['type']) {
            case 'image/jpeg':
                $originalImage = imagecreatefromjpeg($originalImageAbsoluteLoc
ation);

                break;
            case 'image/png':
                $originalImage = imagecreatefrompng($originalImageAbsoluteLoca
tion);

                break;
            case 'image/gif':
                $originalImage = imagecreatefromgif($originalImageAbsoluteLoca
tion);

                break;
        }
        imagecopyresampled($smallImage, $originalImage, 0, 0, 0, 0, $newWidth,
$newHeight, $width, $height);
        $smallImageName = "small_" . $originalImageName;
        $smallImageLocation = LOCATION . $smallImageName;
        $smallImageAbsolutePath = ABSOLUTE_PATH . $smallImageLocation;
        switch($image['type']) {
            case 'image/jpeg':
                $result = imagejpeg($smallImage, $smallImageAbsolutePath);
                break;
            case 'image/png':
                $result = imagepng($smallImage, $smallImageAbsolutePath);
                break;
            case 'image/gif':
                $result = imagegif($smallImage, $smallImageAbsolutePath);
                break;
        }
        // small image
        // image upload
        $continue = $result;
    }
}
if($continue) {
    if(isset($_POST['chbChangeImage'])) {
        $result = editPhoneWithImage($id, $name, $price, $os, $brand, $active,
$originalImageName, $smallImageName);
    } else {
        $result = editPhoneWithoutImage($id, $name, $price, $os, $brand, $acti
ve);
    }
    if($result) {
        http_response_code(200);
        $_SESSION['messages'] = ["Phone was successfully edited."];
    } else {
        http_response_code(500);
        $_SESSION['errors'] = ["We encountered an error."];
    }
}

```

```
        } else {
            http_response_code(500);
            $_SESSION['errors'] = ["We encountered an error."];
        }
    }

    header("Location: ../../../../index.php?p=admin&ap=edit-phone&id=$id");
    exit();
} else {
    http_response_code(400);
}

?>
```

## models/admin/os/add-os.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $name = $_POST['name'];

        $errors = [];
        $regExpOSName = "/^[A-Za-z\s]+$"/;

        if(!preg_match($regExpOSName, $name)) {
            array_push($errors, "OS name cannot contain special characters or numbers;");
        }

        if(count($errors)) {
            http_response_code(400);
            $_SESSION['errors'] = $errors;
        } else {
            if(getOS('name', $name)) {
                http_response_code(409);
                $_SESSION['errors'] = ["That OS already exists."];
            } else {
                if(addOS($name)) {
                    http_response_code(200);
                    $_SESSION['messages'] = ["OS '$name' was added successfully."];
                } else {
                    http_response_code(500);
                    $_SESSION['errors'] = ["We encountered an error."];
                }
            }
        }

        header("Location: ../../../../index.php?p=admin&ap=add-os");
    } else {
        http_response_code(400);
    }
}

?>
```

## models/admin/os/edit-os.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

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

        $errors = [];
        $regExpOSName = "/^[A-Za-z\s]+$/";

        if(!preg_match($regExpOSName, $name)) {
            array_push($errors, "OS name cannot contain special characters or numbers;");
        }

        if(count($errors)) {
            http_response_code(400);
            $_SESSION['errors'] = $errors;
        } else {
            $exists = false;
            $osExists = getOS('name', $name);
            if($osExists) {
                if($osExists->id != $id) $exists = true;
            }
            if($exists) {
                http_response_code(409);
                $_SESSION['errors'] = ['That OS already exists.'];
            } else {
                if(editOS($id, $name)) {
                    http_response_code(200);
                    $_SESSION['messages'] = ["OS was edited successfully."];
                } else {
                    http_response_code(500);
                    $_SESSION['errors'] = ["We encountered an error."];
                }
            }
        }
    }
}
```

```
        header("Location: ../../../../index.php?p=admin&ap=edit-os&id=$id");
    } else {
        http_response_code(400);
    }
?>
```



## models/admin/orders/edit-order.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $id = $_POST['id'];
        $name = $_POST['name'];
        $address = $_POST['address'];

        $errors = [];
        $regExpName = "/^[A-Za-z\s]+$/" ;
        $regExpAddress = "/^[A-ZŠŤČŽ][a-zšďčž]{2,}(\s[A-ZŠŤČŽa-zšďčž][a-zšďčž]{2,})*\s\d+[A-Z]?(\s\d+)*$/";

        if(!preg_match($regExpName, $name)) {
            array_push($errors, "A name cannot have less than 3 characters; Has to be at least 2 words; Each word must be capitalized");
        }
        if(!preg_match($regExpAddress, $address)) {
            array_push($errors, "Address example: [Street Name] [Home Number]/[Apartment Number];");
        }

        if(count($errors)) {
            http_response_code(400);
            $_SESSION['errors'] = $errors;
        } else {
            if(editOrder($id, $name, $address)) {
                http_response_code(200);
                $_SESSION['messages'] = ["Order was edited successfully."];
            } else {
                http_response_code(500);
                $_SESSION['errors'] = ["We encountered an error."];
            }
        }

        header("Location: ../../../../index.php?p=admin&ap=edit-order&id=$id");
    } else {
```

```
    http_response_code(400);  
}  
?>
```

## models/admin/brands/add-brand.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

        $name = $_POST['name'];

        $errors = [];
        $regExpBrandName = "/^[A-Za-z\s]+$/" ;

        if(!preg_match($regExpBrandName, $name)) {
            array_push($errors, "Brand name cannot contain special characters or numbers;"
);
        }

        if(count($errors)) {
            http_response_code(400);
            $_SESSION['errors'] = $errors;
        } else {
            if(getBrand('name', $name)) {
                http_response_code(409);
                $_SESSION['errors'] = ['That brand already exists.'];
            } else {
                if(addBrand($name)) {
                    http_response_code(200);
                    $_SESSION['messages'] = ["Brand '$name' was added successfully."];
                } else {
                    http_response_code(500);
                    $_SESSION['errors'] = ["We encountered an error."];
                }
            }
        }

        header("Location: ../../../../index.php?p=admin&ap=add-brand");
    } else {
        http_response_code(400);
    }
?>
```

## models/admin/brands/edit-brand.php

```
<?php
    session_start();

    if(isset($_SESSION['user'])) {
        if($_SESSION['user']->role_name != 'admin') {
            http_response_code(404);
            exit();
        }
    } else {
        http_response_code(404);
        exit();
    }

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

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

        $errors = [];
        $regExpBrandName = "/^[A-Za-z\s]+$/";

        if(!preg_match($regExpBrandName, $name)) {
            array_push($errors, "Brand name cannot contain special characters or numbers;"
);
        }

        if(count($errors)) {
            http_response_code(400);
            $_SESSION['errors'] = $errors;
        } else {
            $exists = false;
            $brandExists = getBrand('name', $name);
            if($brandExists) {
                if($brandExists->id != $id) $exists = true;
            }
            if($exists) {
                http_response_code(409);
                $_SESSION['errors'] = ['That brand already exists.'];
            } else {
                if(editBrand($id, $name)) {
                    http_response_code(200);
                    $_SESSION['messages'] = ["Brand was edited successfully."];
                } else {
                    http_response_code(500);
                    $_SESSION['errors'] = ["We encountered an error."];
                }
            }
        }
    }
}
```

```
        header("Location: ../../../../index.php?p=admin&ap=edit-brand&id=$id");
    } else {
        http_response_code(400);
    }
?>
```

## config/.env

```
SERVER=localhost  
DATABASE=mobishoptpo  
USERNAME=root  
PASSWORD=
```

## config/config.php

```
<?php
// absolute path
define('ABSOLUTE_PATH', $_SERVER['DOCUMENT_ROOT'] . "/mobishoptpo/");

// files
define('ENV_FILE', ABSOLUTE_PATH . 'config/.env');
define('LOG_FILE', ABSOLUTE_PATH . 'data/log.txt');
define('ERROR_LOG', ABSOLUTE_PATH . 'data/error_log.txt');
define('LOGIN_LOG', ABSOLUTE_PATH . 'data/login_log.txt');
define('SEPARATOR', "\t");

// db
define('SERVER', env('SERVER'));
define('DATABASE', env('DATABASE'));
define('USERNAME', env('USERNAME'));
define('PASSWORD', env('PASSWORD'));

function env($keyword) {
    $file = file(ENV_FILE);
    foreach($file as $line) {
        $line = trim($line);
        $lineContent = explode('=', $line);
        if($lineContent[0] == $keyword) {
            return $lineContent[1];
        }
    }
}
?>
```

## config/connection.php

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

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

function getDateAndTime() {
    return date("d-m-Y H:i:s");
}

updateLog();
function updateLog() {
    $file = fopen(LOG_FILE, "a");
    if($file) {
        isset($_SESSION['user']) ? $userId = $_SESSION['user']->id : $userId = NULL;
        $page = $_SERVER['PHP_SELF'];
        $subPage = '';
        if(isset($_GET['p'])) $subPage = $_GET['p'];
        else if(strpos($page, '/index.php') !== false) $subPage = 'home';
        $ipAddress = $_SERVER['REMOTE_ADDR'];
        $text = $userId . SEPARATOR . $page . SEPARATOR . $subPage . SEPARATOR . $ipAd
dress . SEPARATOR . getDateAndTime() . "\n";
        fwrite($file, $text);
        fclose($file);
    }
}

function updateErrorLog($message) {
    $file = fopen(ERROR_LOG, "a");
    if($file) {
        $page = $_SERVER['PHP_SELF'];
        if(isset($_GET['p'])) {
            $page .= "?p=" . $_GET['p'];
            if(isset($_GET['ap'])) {
                $page .= "&ap=" . $_GET['ap'];
            }
        }
        $ipAddress = $_SERVER['REMOTE_ADDR'];
        $text = $page . SEPARATOR . $ipAddress . SEPARATOR . getDateAndTime() . SEPARA
TOR . $message . "\n";
    }
}
```



```
        fwrite($file, $text);
        fclose($file);
    }
}

function getTime($fullDate) {
    $dateAndTime = explode(" ", $fullDate);

    $date = explode("-", $dateAndTime[0]);
    $time = explode(":", $dateAndTime[1]);

    $hour = intval($time[0]);
    $minute = intval($time[1]);
    $second = intval($time[2]);

    $day = intval($date[0]);
    $month = intval($date[1]);
    $year = intval($date[2]);

    return mktime($hour, $minute, $second, $month, $day, $year);
}

?>
```

## 3.2 CSS

### style.css

```
@font-face {
  src: url("../fonts/Roboto-Regular.ttf");
  font-family: "Roboto";
}

@font-face {
  src: url("../fonts/PermanentMarker-Regular.ttf");
  font-family: "PermanentMarker";
}

* {
  margin: 0;
  padding: 0;
  -webkit-box-sizing: border-box;
  box-sizing: border-box;
  color: #16425b;
  font-family: "Roboto";
}

ul {
  list-style: none;
}

a {
  text-decoration: none;
  color: #16425b;
}

body {
  background-color: #fff6ed;
}

.font-xs {
  font-size: 12px;
}

.font-small {
  font-size: 16px;
}

.font-medium {
  font-size: 19px;
}

.font-large {
  font-size: 30px;
}
```

```

.font-xl {
  font-size: 40px;
}

.font-xxl {
  font-size: 60px;
}

.cover {
  width: 100%;
  height: 100%;
  background-color: rgba(0, 0, 0, 0.6);
}

.cover-white {
  width: 100%;
  height: 100%;
  background-color: rgba(255, 255, 255, 0.6);
}

.underline {
  display: block;
  width: 60px;
  margin-left: auto;
  margin-right: auto;
  height: 2px;
  background-color: #16425b;
}

.btn-primary {
  border: none;
  -webkit-box-shadow: none;
  box-shadow: none;
  background-color: #16425b !important;
  color: #fff6ed !important;
  -webkit-transition: background-color .1s;
  transition: background-color .1s;
}

.btn-primary i {
  color: #fff6ed !important;
}

.btn-primary:hover {
  background-color: #2f6690 !important;
}

.btn-primary:focus {
  -webkit-box-shadow: none;
  box-shadow: none;
}

```

```

.btn-primary:active:focus {
  -webkit-box-shadow: none;
  box-shadow: none;
}

#toTop {
  right: 30px;
  bottom: 30px;
  background-color: #5bc0eb;
  display: block;
}

#toTopArrow {
  width: 10px;
  height: 10px;
  border-left: 2px solid #16425b;
  border-top: 2px solid #16425b;
  -webkit-transform: rotate(45deg);
  transform: rotate(45deg);
  top: 3px;
}

#loginbar {
  background-color: #fff6ed;
}

#loginbar a {
  color: #16425b;
}

.right-border {
  border-right: 1px solid #16425b;
}

#navbar {
  background-color: #2f6690;
}

#navbar a, #navbar i {
  color: #fff6ed;
}

#logo a {
  font-family: PermanentMarker;
}

#nav a {
  position: relative;
}

#nav a:after {
  content: '';
}

```

```

position: absolute;
bottom: -2px;
right: 0;
width: 0;
height: 1px;
background-color: #fff6ed;
-webkit-transition: width .2s;
transition: width .2s;
}

#nav a:hover:after {
  left: 0;
  width: 100%;
}

#cartQuantity {
  color: #fff6ed;
  background-color: #826251;
  right: 0;
  bottom: 6px;
  width: 17px;
  height: 17px;
}

#sidenav {
  position: fixed;
  top: 0;
  left: 0;
}

#sidenavContent {
  width: 250px;
  height: 100%;
  background-color: #2f6690;
}

#sidenavContent a {
  color: #fff6ed;
  -webkit-transition: background-color .1s, color .1s;
  transition: background-color .1s, color .1s;
}

#sidenavContent a:hover {
  color: #16425b;
  background-color: #fff6ed;
}

.slider-image {
  background-position: center;
  background-size: cover;
  background-repeat: no-repeat;
}

```

```

#sliderText {
  top: 0;
  left: 0;
}

#sliderText p, #sliderText h2 {
  color: #fff6ed;
}

#sliderText a {
  background-color: #fff6ed;
  color: #16425b;
  -webkit-transition: background-color .1s, color .1s;
  transition: background-color .1s, color .1s;
}

#sliderText a:hover {
  background-color: #16425b;
  color: #fff6ed;
}

#about h3 {
  color: #826251;
}

#aboutUsImage {
  width: 300px;
}

.phoneDiv {
  overflow: hidden;
}

.phoneDiv:hover .phoneOverlay {
  max-height: 100%;
}

.phoneOverlay {
  top: 0;
  left: 0;
  max-height: 0px;
  -webkit-transition: max-height .3s;
  transition: max-height .3s;
  display: -webkit-box;
  display: -ms-flexbox;
  display: flex;
  overflow: hidden;
}

.btn-white {
  background-color: #fff6ed;
}

```

```

    -webkit-transition: .1s;
    transition: .1s;
}

.btn-white:hover {
    background-color: #5bc0eb;
    color: #fff6ed;
}

.btn-white:hover i {
    color: #fff6ed;
}

.activePage {
    background-color: #2f6690 !important;
}

.cartSuccessModal {
    pointer-events: none;
}

.cartSuccessModalContent {
    display: none;
}

.border-correct {
    border: 1px solid #5bc0eb;
}

.error-message {
    display: block;
}

#authorImage:hover #authorSocials {
    max-height: 100%;
}

#authorSocials {
    width: 100%;
    max-height: 0px;
    top: 0;
    left: 0;
    overflow: hidden;
    display: -webkit-box;
    display: -ms-flexbox;
    display: flex;
    -webkit-transition: max-height .3s;
    transition: max-height .3s;
}

#authorSocials ul {
    display: -webkit-box;

```

```

display: -ms-flexbox;
display: flex;
}

#portfolioLink {
  background-color: #2f6690;
  color: #fff6ed;
}

.table {
  min-width: 600px !important;
}

.btnQuantity {
  width: 20px;
  height: 20px;
}

.table-responsive {
  max-height: 500px !important;
}

.colActions {
  min-width: 130px;
}

.messagesDiv {
  max-height: 700px;
}

#footer {
  background-color: #16425b;
  color: #fff6ed !important;
}

#footer a, #footer i, #footer span {
  color: #fff6ed !important;
}

.footerSeparator {
  width: 20px;
  height: 2px;
  background-color: #fff6ed;
}

@media screen and (max-width: 767px) {
  .font-xs {
    font-size: 10px;
  }
  .font-small {
    font-size: 14px;
  }
}

```



```
.font-medium {  
  font-size: 17px;  
}  
.font-large {  
  font-size: 25px;  
}  
.font-xl {  
  font-size: 30px;  
}  
.font-xxl {  
  font-size: 40px;  
}  
}
```

## 3.3 JavaScript/jQuery

### main.js

```
$(document).ready(function() {  
    // SIDENAV  
    $('#sidenav, #sidenavContent').hide();  
  
    $('#hamburgerMenu').click(function(e) {  
        e.preventDefault();  
        openSidenav();  
    });  
  
    $('#sidenav').click(function() {  
        closeSidenav();  
    });  
  
    $('#sidenavContent').click(function(e) {  
        e.stopPropagation();  
    });  
  
    // TO TOP  
    const toTopHeight = 200;  
    const toTopFadeDuration = 200;  
  
    if($(window).scrollTop() <= toTopHeight) {  
        $('#toTop').hide();  
    }  
  
    $('#toTop').click(function(e) {  
        e.preventDefault();  
        $(window).scrollTop(0);  
    });  
  
    $(window).scroll(function() {  
        if($(window).scrollTop() > toTopHeight) {  
            $('#toTop').fadeIn(toTopFadeDuration);  
        } else {  
            $('#toTop').fadeOut(toTopFadeDuration);  
        }  
    });  
  
    // cart quantity  
    printCartQuantity();  
});  
  
// ajax  
function ajaxCallback(url, method, func, data = {}) {  
    $.ajax({  
        url: url,  
        method: method,  
        data: data,
```

```

        dataType: 'json',
        success: func,
        error: function(xhr) {
            console.error(xhr.responseText);
        }
    });
}

// sidenav
const sidenavAnimationDuration = 200;

function openSidenav() {
    $('#sidenav').fadeIn(sidenavAnimationDuration, function() {
        $('#sidenavContent').animate({width: 'toggle'}, sidenavAnimationDuration);
    });
}

function closeSidenav() {
    $('#sidenavContent').animate({width: 'toggle'}, sidenavAnimationDuration, function() {
        $('#sidenav').fadeOut(sidenavAnimationDuration);
    });
}

// cart quantity
function printCartQuantity() {
    ajaxCallback('models/cart/get-cart-quantity.php', 'GET', function(data) {
        $('#cartQuantity').html(data);
    });
}

```

## form-validation.js

```
$(document).ready(function() {
    $('.error-message').hide();
});

let noErrors;

function validateField(field, regExp) {
    if(field.val().match(regExp)) fieldCorrect(field);
    else fieldIncorrect(field);
}

function fieldCorrect(field) {
    $(field).addClass('border-correct');
    $(field).removeClass('border-danger');
    $(field).siblings('.error-message').slideUp();
}

function fieldIncorrect(field) {
    $(field).addClass('border-danger');
    $(field).removeClass('border-correct');
    $(field).siblings('.error-message').slideDown();
    noErrors = false;
}

function fieldNeutral(field) {
    $(field).removeClass('border-danger');
    $(field).removeClass('border-correct');
    $(field).siblings('.error-message').slideUp();
}

function submitForm(checkFunctions, form = false) {
    noErrors = true;
    for(var f of checkFunctions) {
        f();
    }
    if(noErrors) {
        if(form) clearForm(form);
        return true;
    }
    return false;
}
```

## phones.js

```
let page = 1;
let phonesPerPage = 6;

$(document).ready(function() {

    $('.chbBrand, .chbOS, #ddlSortOrder').change(function() {
        getPhones(true);
    });
    $('#tbSearch, #tbMaxPrice').on('input', function() {
        getPhones(true);
    });

    getPhones();
    function getPhones(resetPage = false) {
        if(resetPage) page = 1;

        // brands
        let filteredBrands = [];
        for(let el of $('.chbBrand:checked')) {
            filteredBrands.push(el.value);
        }

        // os
        let filteredOS = [];
        for(let el of $('.chbOS:checked')) {
            filteredOS.push(el.value);
        }

        // max price
        let maxPrice = $('#tbMaxPrice').val();

        // search
        let search = $('#tbSearch').val();

        // sort order
        let sortOrder = $('#ddlSortOrder').val();

        ajaxCallback('models/phones/get-phones.php', 'GET', function(data) {
            printPhones(data);
        }, {
            brand: filteredBrands,
            os: filteredOS,
            maxPrice: maxPrice,
            search: search,
            order: sortOrder,
            page: page,
            phonesPerPage: phonesPerPage
        });
    }
}
```

```

function printPhones(data) {
  // phones
  let phones = data.phones;
  let html = '';
  if(phones.length > 0) {
    for(let p of phones) {
      html += `<div class="mx-auto mx-sm-0 col-9 col-sm-6 col-lg-4 text-
center mb-3">
      <div class="phoneDiv position-relative rounded">
      <div class="phoneImage">
        
      </div>
      <label class="phoneName font-medium">${p.name}</label>
      <br/>
      <label class="phonePrice font-medium fw-bold">${p.price}&euro;</label>
      <div class="phoneOverlay position-absolute cover align-items-
center justify-content-center">
        <div class="phoneButtons d-flex flex-column">
          <a href="index.php?p=details&id=${p.id}" class="btn-white py-1 px-
2 rounded mb-2 d-block font-small">Details</a>
          <a href="#" data-id="${p.id}" class="btn-white btnAddToCart py-
1 px-2 rounded d-block font-small"><i class="fas fa-cart-plus"></i></a>
        </div>
      </div>
    </div>`;
    }
  } else {
    html = "<div class='col-12'><p class='text-center text-md-start font-
medium'>Oops, seems like there are no phones that match your search criteria.</p></div>";
  }
  $('#phones .row').html(html);

  // pagination
  let numberOfPhones = data.numberOfPhones;
  let paginationHtml = '';
  if(numberOfPhones > 0) {
    paginationHtml += `<a href='#' id='btnPrev' class="font-small btn-primary px-
3 py-1 rounded-start">&lt;</a>`;
    for(let i = 0; i < numberOfPhones / phonesPerPage; i++) {
      paginationHtml += `<a href='#' data-page='${i + 1}' class="font-small btn-
primary px-2 py-1 btnPage">${i + 1}</a>`;
    }
    paginationHtml += `<a href='#' id='btnNext' class="font-small btn-primary px-
3 py-1 rounded-end">&gt;</a>`;
  }
  $('#pagination').html(paginationHtml);
  enablePagination();
  enableAddToCart();
}

```

```

function enablePagination() {
    $('#btnPrev').click(function(e) {
        e.preventDefault();
        if(page > 1) {
            page--;
            getPhones();
        }
    });

    $('#btnNext').click(function(e) {
        e.preventDefault();
        if(page < $('#.btnPage').length) {
            page++;
            getPhones();
        }
    });

    $('#.btnPage').click(function(e) {
        e.preventDefault();
        page = $(this).data('page');
        getPhones();
    });

    for(let el of $('#.btnPage')) {
        if($(el).data('page') == page) {
            $(el).addClass('activePage');
            break;
        }
    }
}

function enableAddToCart() {
    $('#.btnAddToCart').click(function(e) {
        e.preventDefault();
        let id = $(this).data('id');
        addToCart(id);
    });
}

$('#btnAddToCart').click(function(e) {
    e.preventDefault();
    let id = $(this).data('id');
    let quantity = parseInt($('#tbAddToCart').val());
    addToCart(id, quantity);
});

function addToCart(phoneId, quantity = 1) {
    let user;
    ajaxCallback('models/users/get-current-user.php', 'GET', function(data) {
        user = data;
        if(user) {
            ajaxCallback('models/cart/add-to-cart.php', 'POST', function(data) {

```

```

        openCartSuccessModal(data);
        printCartQuantity();
    }, {
        userId: user.id,
        phoneId: phoneId,
        quantity: quantity
    });
} else {
    location.href = "index.php?p=login";
}
}, {
    message: "You need to be logged in in order to add items to your cart."
});
}

const modalDuration = 3;
let timer;
function openCartSuccessModal(message) {
    if($('.cartSuccessModal').length > 0) {
        $('.cartSuccessModalContent').html(message);
        clearTimeout(timer);
        timer = setModalCloseTimeout(modalDuration);
    } else {
        let html = `<div class="cartSuccessModal w-100 h-100 position-fixed fixed-
top pt-5">
        <div class="row mt-5">
            <div class="cartSuccessModalContent mx-auto col-8 col-md-3 alert alert-
info text-center">${message}</div>
        </div>
    </div>`;
        $('body').append(html);
        $('.cartSuccessModalContent').fadeIn('fast');
        timer = setModalCloseTimeout(modalDuration);
    }
}

function setModalCloseTimeout(duration) {
    return setTimeout(closeCartSuccessModal, duration * 1000);
}

function closeCartSuccessModal() {
    $('.cartSuccessModalContent').fadeOut('fast', function() {
        $('.cartSuccessModal').remove();
    });
}
});

```



## contact.js

```
$(document).ready(function() {  
    // FORM VALIDATION  
    // NAME  
    let regExpName = /^[A-ZŠĐČĆŽ][a-zšđčćž]{2,}(\s[A-ZŠĐČĆŽ][a-zšđčćž]{2,})*/$;/  
    let $tbName = $('#tbName');  
    $tbName.blur(checkName);  
    function checkName() {  
        validateField($tbName, regExpName);  
    }  
  
    // EMAIL  
    let regExpEmail = /^[a-z][a-z0-9-_\.\.]{2,}@([a-z0-9-_\.\.]{2,}\.)+[a-z]{2,}$/;  
    let $tbEmail = $('#tbEmail');  
    $tbEmail.blur(checkEmail);  
    function checkEmail() {  
        validateField($tbEmail, regExpEmail);  
    }  
  
    // MESSAGE  
    let $tbMessage = $('#tbMessage');  
    function checkMessage() {  
        let numberOfSpaces = tbMessage.value.replace(/^[^\s]/mg, "").length;  
        if(tbMessage.value.length - numberOfSpaces < 20 || tbMessage.value.length > 500) {  
            fieldIncorrect($tbMessage);  
        } else {  
            fieldCorrect($tbMessage);  
        }  
    }  
    $tbMessage.blur(checkMessage);  
  
    // FORM SUBMISSION  
    let checkFunctions = [checkName, checkEmail, checkMessage];  
    $('#btnSend').click(function() {  
        return submitForm(checkFunctions);  
    });  
});
```

## cart.js

```
$(document).ready(function() {
  // FORM VALIDATION
  // NAME
  let regExpName = /^[A-ZŠĐČĆŽ][a-zšđććž]{2,}(\s[A-ZŠĐČĆŽ][a-zšđććž]{2,})+$/;
  let $tbName = $('#tbName');
  $tbName.blur(checkName);
  function checkName() {
    validateField($tbName, regExpName);
  }

  // ADDRESS
  let regExpAddress = /^[A-ZŠĐČĆŽ][a-zšđććž]{2,}(\s[A-ZŠĐČĆŽa-zšđććž][a-zšđććž]{2,})*\s\d+[A-Z]?(\s\d+)*$/;
  let $tbAddress = $('#tbAddress');
  $tbAddress.blur(checkAddress);
  function checkAddress() {
    validateField($tbAddress, regExpAddress);
  }

  // FORM SUBMISSION
  let checkFunctions = [checkName, checkAddress];
  $('#btnCheckout').click(function() {
    return submitForm(checkFunctions);
  });

  // cart table
  getCart();
  function getCart() {
    ajaxCallback('models/cart/get-cart.php', 'GET', function(data) {
      printCart(data);
      printCartQuantity();
    });
  }

  function printCart(data) {
    let totalPrice = 0;
    let html = '';
    for(let el of data) {
      html += `<tr>
        <td class="colDeviceName text-center">${el.name}</td>
        <td class="colPrice text-center">${el.price}&euro;</td>
        <td class="colQuantity d-flex align-items-center justify-content-center"><a href="#" data-id="${el.id}" data-change='decrease' class="btnQuantity btn-primary rounded d-flex align-items-center justify-content-center">-</a><span class="mx-2">${el.quantity}</span><a href="#" data-id="${el.id}" data-change='increase' class="btnQuantity btn-primary rounded d-flex align-items-center justify-content-center">+</a></td>
        <td class="colRemove text-center"><a href="#" data-id="${el.id}" class="btnRemove"><i class="fas fa-minus text-danger"></i></a></td>
      </tr>`;
    }
  }
}
```

```

        totalPrice += el.quantity * el.price;
    }
    $('#cartTable tbody').html(html);
    $('#totalPrice').html(totalPrice);
    enableQuantityChange();
    enableRemoveItem();
}

function enableQuantityChange() {
    $('.btnQuantity').click(function(e) {
        e.preventDefault();
        let id = $(this).data('id');
        let change = $(this).data('change');
        changeQuantity(id, change);
    });
}

function changeQuantity(id, change) {
    ajaxCallback('models/cart/change-quantity.php', 'POST', function() {
        getCart();
    }, {
        id: id,
        change: change
    });
}

function enableRemoveItem() {
    $('.btnRemove').click(function(e) {
        e.preventDefault();
        let id = $(this).data('id');
        removeItem(id);
    });
}

function removeItem(id) {
    ajaxCallback('models/cart/remove-item.php', "POST", function() {
        getCart();
    }, {
        id: id
    });
}
});

```

## login.js

```
$(document).ready(function() {  
    // FORM VALIDATION  
    // EMAIL  
    let regExpEmail = /^[a-z][a-z0-9-_\.\.]{2,}@([a-z0-9-_\.\.]{2,}\.)+[a-z]{2,}$/;  
    let $tbEmail = $('#tbEmail');  
    $tbEmail.blur(checkEmail);  
    function checkEmail() {  
        validateField($tbEmail, regExpEmail);  
    }  
  
    // PASSWORD  
    let regExpPassword = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%\^&\*])(?=.{8,})/;  
    let $tbPassword = $('#tbPassword');  
    $tbPassword.blur(checkPassword);  
    function checkPassword() {  
        validateField($tbPassword, regExpPassword);  
    }  
  
    // FORM SUBMISSION  
    let checkFunctions = [checkEmail, checkPassword];  
    $('#btnLogIn').click(function() {  
        return submitForm(checkFunctions);  
    });  
});
```

## register.js

```
$(document).ready(function() {  
    // FORM VALIDATION  
    // USERNAME  
    let regExpUsername = /^[a-zA-Z]\w{4,20}$/;  
    let $tbUsername = $('#tbUsername');  
    $tbUsername.blur(checkUsername);  
    function checkUsername() {  
        validateField($tbUsername, regExpUsername);  
    }  
  
    // EMAIL  
    let regExpEmail = /^[a-z][a-z0-9-_\.\.]{2,}@([a-z0-9-_\.\.]{2,}\.)+[a-z]{2,}$/;  
    let $tbEmail = $('#tbEmail');  
    $tbEmail.blur(checkEmail);  
    function checkEmail() {  
        validateField($tbEmail, regExpEmail);  
    }  
  
    // PASSWORD  
    let regExpPassword = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%\^&\*])(?=.{8,})/;  
    let $tbPassword = $('#tbPassword');  
    $tbPassword.blur(function() {  
        checkPassword();  
        if($($tbRepeatPassword).val() != '') {  
            checkRepeatPassword();  
        }  
    });  
    function checkPassword() {  
        validateField($tbPassword, regExpPassword);  
    }  
  
    //REPEAT PASSWORD  
    let $tbRepeatPassword = $('#tbRepeatPassword');  
    $tbRepeatPassword.blur(checkRepeatPassword);  
    function checkRepeatPassword() {  
        if($($tbPassword).hasClass('border-  
correct') && $($tbRepeatPassword).val() == $($tbPassword).val()) {  
            fieldCorrect($tbRepeatPassword);  
        } else {  
            fieldIncorrect($tbRepeatPassword);  
        }  
    }  
  
    // FORM SUBMISSION  
    let checkFunctions = [checkUsername, checkEmail, checkPassword, checkRepeatPassword];  
    $('#btnRegister').click(function() {  
        return submitForm(checkFunctions);  
    });  
});
```

## admin.js

```
$(document).ready(function() {  
    // === brands ===  
    // FORM VALIDATION  
    // NAME  
    let regExpBrandName = /^[A-Za-z\s]+$/;  
    let $tbAddBrandName = $('#tbAddBrandName');  
    $tbAddBrandName.blur(checkAddBrandName);  
    function checkAddBrandName() {  
        validateField($tbAddBrandName, regExpBrandName);  
    }  
  
    let $tbEditBrandName = $('#tbEditBrandName');  
    $tbEditBrandName.blur(checkEditBrandName);  
    function checkEditBrandName() {  
        validateField($tbEditBrandName, regExpBrandName);  
    }  
  
    // FORM SUBMISSION  
    let checkFunctionsAddBrand = [checkAddBrandName];  
    $('#btnAddBrand').click(function() {  
        return submitForm(checkFunctionsAddBrand);  
    });  
  
    let checkFunctionsEditBrand = [checkEditBrandName];  
    $('#btnEditBrand').click(function() {  
        return submitForm(checkFunctionsEditBrand);  
    });  
  
    // === os ===  
    // FORM VALIDATION  
    // NAME  
    let regExpOSName = /^[A-Za-z\s]+$/;  
    let $tbAddOSName = $('#tbAddOSName');  
    $tbAddOSName.blur(checkAddOSName);  
    function checkAddOSName() {  
        validateField($tbAddOSName, regExpOSName);  
    }  
  
    let $tbEditOSName = $('#tbEditOSName');  
    $tbEditOSName.blur(checkEditOSName);  
    function checkEditOSName() {  
        validateField($tbEditOSName, regExpOSName);  
    }  
  
    // FORM SUBMISSION  
    let checkFunctionsAddOS = [checkAddOSName];  
    $('#btnAddOS').click(function() {  
        return submitForm(checkFunctionsAddOS);  
    });  
});
```

```

let checkFunctionsEditOS = [checkEditOSName];
$('#btnEditOS').click(function() {
    return submitForm(checkFunctionsEditOS);
});

// === orders ===
// FORM VALIDATION
// NAME
let regExpOrderName = /^[A-ZŠĐČĆŽ][a-zšđććž]{2,}(\s[A-ZŠĐČĆŽ][a-zšđććž]{2,})+$/;
let $tbEditOrderName = $('#tbEditOrderName');
$tbEditOrderName.blur(checkEditOrderName);
function checkEditOrderName() {
    validateField($tbEditOrderName, regExpOrderName);
}

// ADDRESS
let regExpOrderAddress = /^[A-ZŠĐČĆŽ][a-zšđććž]{2,}(\s[A-ZŠĐČĆŽa-zšđććž][a-zšđććž]{2,})*\s\d+[A-Z]?(\s\d+)*$/;
let $tbEditOrderAddress = $('#tbEditOrderAddress');
$tbEditOrderAddress.blur(checkEditOrderAddress);
function checkEditOrderAddress() {
    validateField($tbEditOrderAddress, regExpOrderAddress);
}

// FORM SUBMISSION
let checkFunctionsEditOrder = [checkEditOrderAddress];
$('#btnEditOrder').click(function() {
    return submitForm(checkFunctionsEditOrder);
});

// === users ===
// FORM VALIDATION
// USERNAME
let regExpUsername = /^[a-zA-Z]\w{4,20}$/;
let $tbAddUserUsername = $('#tbAddUserUsername');
$tbAddUserUsername.blur(checkAddUserUsername);
function checkAddUserUsername() {
    validateField($tbAddUserUsername, regExpUsername);
}

let $tbEditUserUsername = $('#tbEditUserUsername');
$tbEditUserUsername.blur(checkEditUserUsername);
function checkEditUserUsername() {
    validateField($tbEditUserUsername, regExpUsername);
}

// EMAIL
let regExpEmail = /^[a-z][a-z0-9-_\.\s]{2,}@([a-z0-9-_\.\s]{2,}\.)+[a-z]{2,}$/;
let $tbAddUserEmail = $('#tbAddUserEmail');
$tbAddUserEmail.blur(checkAddUserEmail);
function checkAddUserEmail() {
    validateField($tbAddUserEmail, regExpEmail);
}

```

```

}

let $tbEditUserEmail = $('#tbEditUserEmail');
$tbEditUserEmail.blur(checkEditUserEmail);
function checkEditUserEmail() {
    validateField($tbEditUserEmail, regExpEmail);
}

// PASSWORD
let regExpPassword = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*])(?=.{8,})/;
let $tbAddUserPassword = $('#tbAddUserPassword');
$tbAddUserPassword.blur(checkAddUserPassword);
function checkAddUserPassword() {
    validateField($tbAddUserPassword, regExpPassword);
}

let $tbEditUserPassword = $('#tbEditUserPassword');
$tbEditUserPassword.blur(checkEditUserPassword);
function checkEditUserPassword() {
    if(!$('#tbEditUserPassword').prop('disabled')) {
        validateField($tbEditUserPassword, regExpPassword);
    }
}

// FORM SUBMISSION
let checkFunctionsAddUser = [checkAddUserUsername, checkAddUserEmail, checkAddUserPass
word];
$('#btnAddUser').click(function() {
    return submitForm(checkFunctionsAddUser);
});

let checkFunctionsEditUser = [checkEditUserEmail, checkEditUserPassword, checkEditUser
Username];
$('#btnEditUser').click(function() {
    return submitForm(checkFunctionsEditUser);
});

if($('#chbChangePassword').prop('checked')) {
    $('#tbEditUserPassword').prop('disabled', false);
} else {
    $('#tbEditUserPassword').prop('disabled', true);
}
$('#chbChangePassword').change(function() {
    if(this.checked) {
        $('#tbEditUserPassword').prop('disabled', false);
    } else {
        $('#tbEditUserPassword').prop('disabled', true);
        fieldNeutral($tbEditUserPassword);
    }
});

// === phones ===

```



```

// FORM VALIDATION
// NAME
let regExpPhoneName = /^[A-Za-z0-9\-\.\s]+$/;
let $tbAddPhoneName = $('#tbAddPhoneName');
$tbAddPhoneName.blur(checkAddPhoneName);
function checkAddPhoneName() {
    validateField($tbAddPhoneName, regExpPhoneName);
}

let $tbEditPhoneName = $('#tbEditPhoneName');
$tbEditPhoneName.blur(checkEditPhoneName);
function checkEditPhoneName() {
    validateField($tbEditPhoneName, regExpPhoneName);
}

// PRICE
let $tbAddPhonePrice = $('#tbAddPhonePrice');
$tbAddPhonePrice.blur(checkAddPhonePrice);
function checkAddPhonePrice() {
    if($('#tbAddPhonePrice').val() == '') {
        noErrors = false;
        fieldIncorrect($tbAddPhonePrice);
    } else {
        fieldCorrect($tbAddPhonePrice);
    }
}

let $tbEditPhonePrice = $('#tbEditPhonePrice');
$tbEditPhonePrice.blur(checkEditPhonePrice);
function checkEditPhonePrice() {
    if($('#tbEditPhonePrice').val() == '') {
        noErrors = false;
        fieldIncorrect($tbEditPhonePrice);
    } else {
        fieldCorrect($tbEditPhonePrice);
    }
}

// IMAGE
let $fileAddPhoneImage = $('#fileAddPhoneImage');
$fileAddPhoneImage.on('input', checkAddPhoneImage);
function checkAddPhoneImage() {
    if($('#fileAddPhoneImage').val() == '') {
        noErrors = false;
        fieldIncorrect($fileAddPhoneImage);
    } else {
        fieldCorrect($fileAddPhoneImage);
    }
}

let $fileEditPhoneImage = $('#fileEditPhoneImage');
$fileEditPhoneImage.on('input', checkEditPhoneImage);

```

```

function checkEditPhoneImage() {
    if(!$('#fileEditPhoneImage').prop('disabled')) {
        if($('#fileEditPhoneImage').val() == '') {
            noErrors = false;
            fieldIncorrect($fileEditPhoneImage);
        } else {
            fieldCorrect($fileEditPhoneImage);
        }
    }
}

if($('#chbChangeImage').prop('checked')) {
    $('#fileEditPhoneImage').prop('disabled', false);
} else {
    $('#fileEditPhoneImage').prop('disabled', true);
}

$('#chbChangeImage').change(function() {
    if(this.checked) {
        $('#fileEditPhoneImage').prop('disabled', false);
    } else {
        $('#fileEditPhoneImage').prop('disabled', true);
        fieldNeutral($fileEditPhoneImage);
    }
});

// FORM SUBMISSION
let checkFunctionsAddPhone = [checkAddPhoneName, checkAddPhonePrice, checkAddPhoneImage];

$('#btnAddPhone').click(function() {
    return submitForm(checkFunctionsAddPhone);
});

let checkFunctionsEditPhone = [checkEditPhoneName, checkEditPhonePrice, checkEditPhoneImage];

$('#btnEditPhone').click(function() {
    return submitForm(checkFunctionsEditPhone);
});
});

```