

VISOKA ICT SKOLA

Zdravka Celara 16
Beograd

Smer: **Internet tehnologije**
Modul: **Web programiranje**
Predmet: **Praktikum iz web programiranja PHP**

DOKUMENTACIJA

Webio Blog Site

<https://webio-php.000webhostapp.com/index.php?page=index>

Nevena Nikolic, 174/21

SADRZAJ

UVOD	3
➤ Html	3
➤ CSS	3
➤ JavaScript	3
➤ jQuery	3
➤ Ajax	3
➤ XML	3
➤ PHP	3
➤ phpMyAdmin	3
➤ XAMPP	3
➤ Regex	3
➤ PHP Mailer	3
1.3.1. Funkcionalnosti za admina:	5
1.3.2. Funkcionalnosti za korisnika:	5
1.5 Organizacija - schema baze:	6
2.2 ?page=login	11
api/ check_login_attempts:	18
Helpers/mail.php:	19
/models/attempt_logins.php	21

UVOD

1.1 Korisceni programski jezici

Za izradu sajta, korisceni su sledeci jezici:

- **Html**
- **CSS**
- **JavaScript**
- **jQuery**
- **Ajax**
- **XML**
- **PHP**
- **phpMyAdmin**
- **XAMPP**
- **Regex**
- **PHP Mailer**

Svi navedeni jezici, biblioteke i framework-ovi su pisani unutar Visual Studio Code okruzenja.

1.2 Opis funkcionalnosti

- **Dinamičko ispisivanje postova iz baze podataka:** Sistem omogućava prikazivanje blogova na web stranici, pri čemu se sadržaj dinamički učitava iz baze podataka. Korisnici mogu pregledati i čitati različite postove koji su sačuvani u bazi.
- **Sistem registracije i upisivanje korisnika u bazu:** Korisnicima se pruža mogućnost da se registruju na web stranici. Nakon što se registruju, njihovi podaci se upisuju u bazu podataka radi dalje obrade.
- **Logovanje korisnika i prikazivanje sadržaja sajta u skladu s ulogom:** Korisnici mogu se prijaviti na svoje korisničke naloge kako bi imali pristup određenim funkcijama i sadržaju. Nakon logovanja, korisnicima se prikazuje sadržaj stranice koji je specifičan za njihovu ulogu (na primer, korisnik ili admin). Korisnici koji se uloguju mogu lajkovati i komentarisati postove, oni koji nisu ulogovani - ne mogu.
- **Anketa za logovane korisnike:** Kada je korisnik ulogovan, ima pristup stranici "survey.php" na kojoj može popuniti anketu i odabrati odgovor koji pokazuje njegovo zadovoljstvo sajtom. Podaci o korisniku (ID) dobijeni prilikom registracije, kao i ID odabranog odgovora, čuvaju se u bazi podataka. Nakon toga, e-mail korisnika i tekstualni odgovor se ispisuju adminu na stranici "admin/dashboard.php".
- **Zaštita od pristupa admin stranicama:** Ukoliko ulogovani korisnik ili guest pokuša pristupiti bilo kojoj admin stranici putem URL adrese, dobiće obaveštenje o zabrani pristupa i informaciju da nema autorizaciju za tu određenu stranicu. Ovo osigurava da samo admin ima pristup administrativnim funkcijama.

- **Kontaktiranje admina:** Korisnici ili gosti mogu pristupiti stranici "contact.php" na kojoj mogu poslati upit adminu ili ostaviti poruku. Naslov i sadržaj koji korisnik bude upisao, šalje se na e-mail admina.

1.3.1. Funkcionalnosti za admina:

Admin je označen u bazi podataka kolonom "role" sa vrednošću "1" i prilikom logovanja se preusmerava na stranicu "dashboard.php". Na ovoj stranici admin može videti odgovore korisnika koji su učestvovali u anketi i koje odgovore su poslali.

Admin ima pristup stranicama "posts/index.php" (posts/edit.php, posts/create.php), "topics/index.php" (topics/edit.php, topics/create.php) i "users/index.php" (users/edit.php, users/create.php).

Dodatne funkcionalnosti za admina uključuju:

- **Upravljanje postovima:** Admin ima mogućnost pregledanja svih postova na stranici "posts/index.php". Takođe, admin može uređivati ili brisati postojeće postove na stranici "posts/edit.php". Ukoliko želi dodati novi post, može to učiniti na stranici "posts/create.php". Admin može dodatno da regulise postove da li će biti vidljivi (publish) ili trenutno nevidljivi (unpublished).
- **Upravljanje temama:** Admin može pregledati sve dostupne teme na stranici "topics/index.php". Takođe, admin ima mogućnost uređivanja ili brisanja postojećih tema na stranici "topics/edit.php". Za dodavanje novih tema, admin može koristiti stranicu "topics/create.php".
- **Upravljanje korisnicima:** Admin ima pristup stranici "users/index.php" gde može pregledati sve registrovane korisnike. Na stranici "users/edit.php" admin može uređivati korisničke podatke ili čak izbrisati korisničke naloge. Ukoliko želi dodati nove korisnike, admin to može uraditi na stranici "users/create.php".
- **Prikaz statistike i odgovora korisnika:** Admin može videti odgovore korisnika na anketu i prikazati statistiku zadovoljstva korisnika na stranici "admin/dashboard.php". Ovo omogućava adminu uvid u mišljenja korisnika i priliku da poboljša sajt na osnovu njihovih povratnih informacija. Takođe ima uvid i u poslate upite koji su korisnici ili gosti prosledili putem stranice contact.php .

Ove funkcionalnosti omogućavaju adminu kompletno upravljanje sadržajem, korisnicima i anketom na sajtu, pružajući im alate za uređivanje, dodavanje i brisanje podataka u bazi, kao i uvid u aktivnosti korisnika.

1.3.2. Funkcionalnosti za korisnika:

- **Lajkovanje postova:** Nakon što se korisnik uloguje, ima mogućnost lajkovanja određenog posta. Korisnik može kliknuti na gumb "Like" kako bi izrazio svoje oduševljenje ili podršku prema određenom postu. Također, ako je već lajkovao post, korisnik može kliknuti na gumb "Dislike" kako bi uklonio svoj lajk.
- **Komentarisanje postova:** Kada je korisnik ulogovan, ima mogućnost ostavljanja komentara na određeni post. Korisnik može unijeti svoje željeno ime i poruku koju želi priložiti. Komentar će biti objavljen na stranici zajedno s korisnikovim imenom i porukom.
- **Anketa:** Nakon što se korisnik uloguje, može pristupiti anketi na web stranici. Korisnik ima mogućnost dati svoj odgovor na pitanje ankete koje se odnosi na zadovoljstvo sajtom. Na primjer, korisnik može ocijeniti sajt na skali od 1 do 5 ili izabrati opciju koja najbolje opisuje njegovo zadovoljstvo.
- **Kontakt:** Korisnici mogu pristupiti stranici za kontaktiranje i poslati sugestiju ili pitanje administratoru web stranice. Korisnici mogu unijeti svoje ime, e-mail adresu, poruku. Nakon što kliknu na gumb "Pošalji", poruka će biti poslana administratoru putem e-maila.

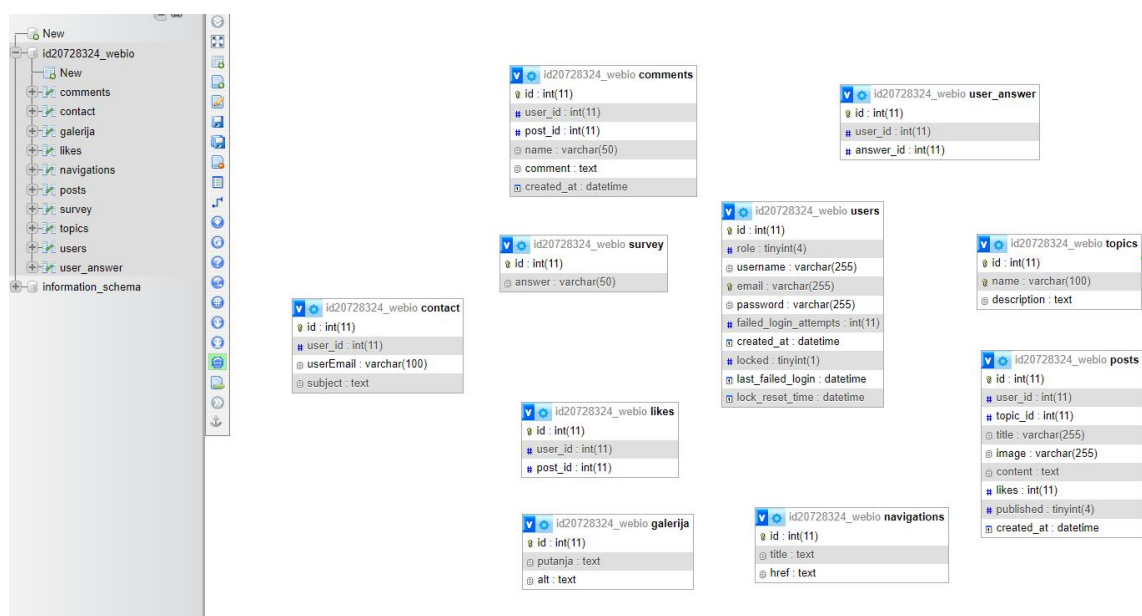
1.4 Template

Projekat je implementiran koristeći **Bootstrap framework** za osiguravanje responsive web stranica na različitim uređajima. Umesto korišćenja gotovog template-a, dizajn je prilagođen specifičnim zahtevima i funkcionalnostima projekta.

Glavni cilj implementacije bilo je stvaranje jednostavnog, ali intuitivnog korisničkog interfejsa koji omogućava korisnicima da lako pristupaju funkcijama sajta. Bootstrap je korišćen za brzo i efikasno kreiranje dizajna, obezbeđujući rešenja za raspoređivanje elemenata, stilizovanje dugmadi, obrazaca, navigacionog menija i drugih interaktivnih elemenata.

5

1.5 Organizacija - schema baze:



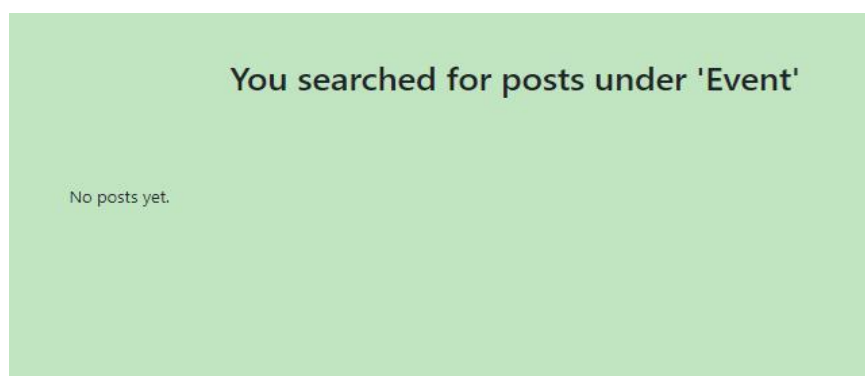
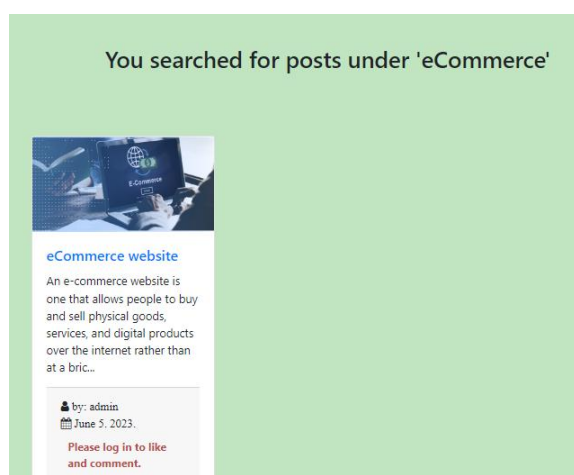
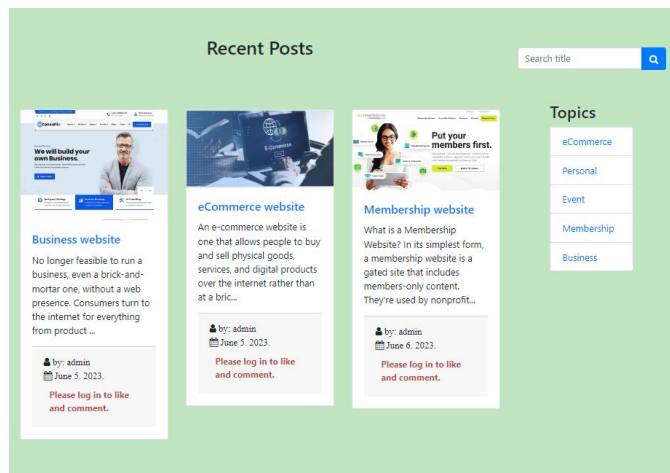
1.6 Organizacija foldera

dashboard.php: Ova stranica predstavlja glavnu administratorsku nadzornu tablu gde administrator ima pregled informacija i funkcionalnosti vezanih za odgovore korisnika na anketu, statistiku i upite korisnika.

- A) **folder api:** Ovaj folder sadrži PHP skripte koje se koriste za pružanje API funkcionalnosti.
 - **check_login_attempts.php:** koristi se za provjeru broja neuspjelih prijava korisnika.
 - **contact.php:** koristi se za obradu kontaktiranja (slanje email admin korisniku).
- B) **folder assets:** Ovaj folder sadrži podfoldere za CSS stilove (css), JavaScript datoteke (js) i slike (images).
- C) **config/database/:** Ovaj folder sadrži connect.php datoteku koja se koristi za uspostavljanje veze s bazom podataka.
- D) **data/:**
 - **login.log:** evidencija prijave korisnika tokom jednog dana
 - **assets.log:** evidencija pristupa svim stranicama
- E) **models/:** Ovaj folder sadrži različite modele, tj. PHP datoteke koje opisuju poslovnu logiku ili manipulaciju podacima
- F) **views/:** Ovaj folder sadrži pristup svim stranicama:
 - **fixed/:** header.php, footer.php, adminSidebar.php
 - **pages/:** author.php, contact.php, create-post.php, create-topic.php, create-user.php, edit-post.php, edit-topic.php, edit-user.php, index-post.php, index-topic.php, index-user.php, dashboard.php, gallery.php, index.php, login.php, logout.php, register.php, more.php, survey.php

1.7 Slike stranica I opis funkcionalnosti

Mogućnost pretrage blogova na osnovu unetog naziva od strane korisnika:

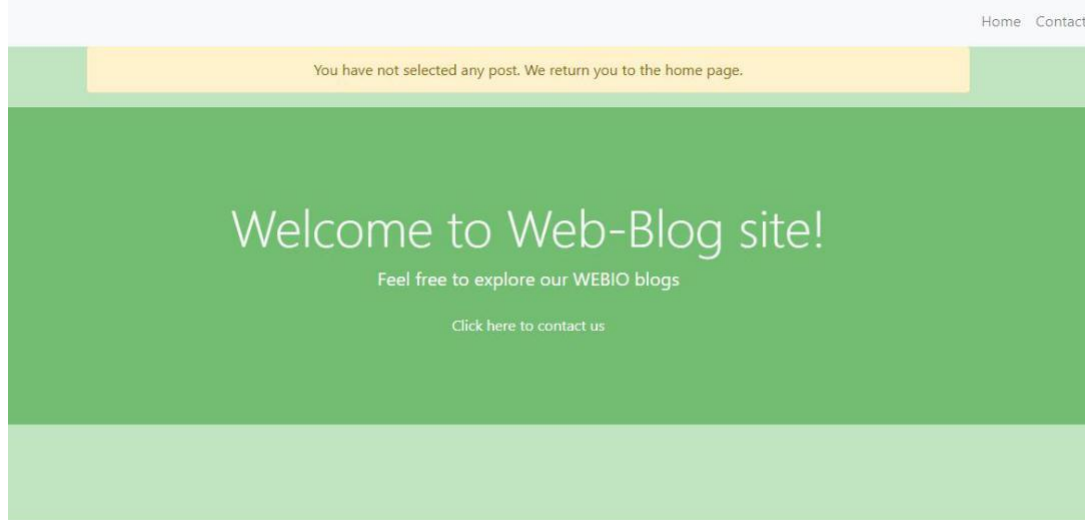


?page=index

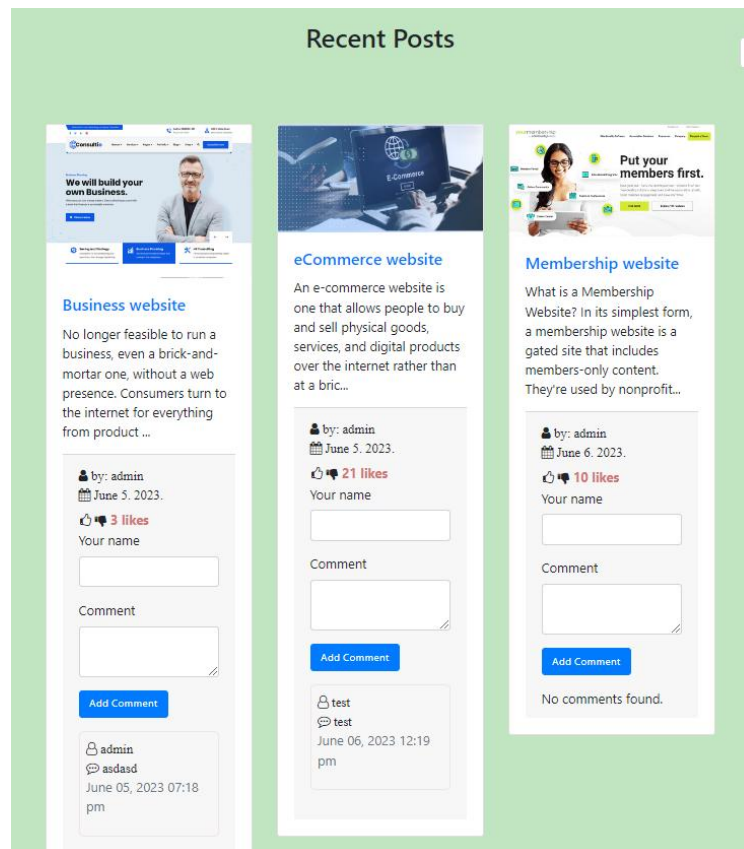
Korisnik ima mogućnost da unese željeni naziv bloga u polje za pretragu ili da koristi listu linkova za filtriranje rezultata.

Kada korisnik unese naziv bloga i pokrene pretragu, sistem pretražuje bazu podataka i pronalazi sve postove koji sadrže taj naziv. Npr. ukoliko korisnik unese "eCommerce" kao naziv bloga, sistem će pronaći sve postove koji imaju "eCommerce" u naslovu ili sadržaju.

Rezultati pretrage se zatim prikazuju korisniku na stranici, gde se prikazuju samo postovi koji zadovoljavaju pretragu. Svaki post je predstavljen sažetim prikazom koji sadrži naslov, kratki opis i datum objave. Korisnik može kliknuti na željeni post kako bi pročitao ceo sadržaj na stranici `more.php`, u suprotnom ako želi da pristupi toj stranici bez da je kliknuo na željeni post, na `index.php` ce mu se pojaviti **warning poruka** da mora prvo nesto izabrati.

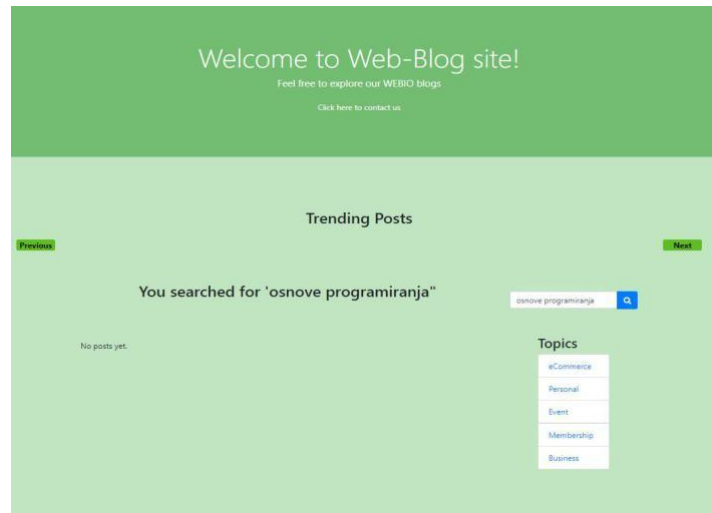


Takodje, korisnik kada je uspesno ulogovan na sajt, sada ima pristup lajkovanju I komentarisanju postova:



Takodje, **prikazuju se samo oni postovi koji imaju publish: active, a to samo Admin moze da menja.**

Ukoliko nema postova koji zadovoljavaju unetu pretragu, na stranici će se prikazati poruka "No posts yet" kako bi korisnik bio obavešten da nema rezultata za uneti naziv bloga.



Ova funkcionalnost omogućava korisnicima da brzo pronađu blogove na osnovu njihovih naziva i dobiju relevantne rezultate. Takođe, pruža intuitivan način pretrage i filtriranja sadržaja na sajtu.

LOGIN
Please enter your username and password!

The email is invalid
Your email

Password

Login

Don't have an account? Sign Up

LOGIN
Please enter your username and password!

The email is required
Your email

The password is required
Password

Login

Don't have an account? Sign Up

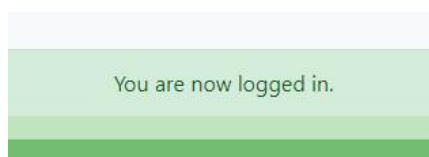
2.2 ?page=login

Prilikom logovanja, kada korisnik koji je već registrovan želi da se prijavi, sistem vrši validaciju unetih podataka kako bi se osigurala njihova ispravnost. Ako korisnik pokuša da prijavi prazna polja klikom na "Login", pojaviće se poruke o greškama ispod svakog polja koja su obavezna za popunjavanje. Na taj način korisnik će biti obavešten da mora uneti podatke u sva obavezna polja.

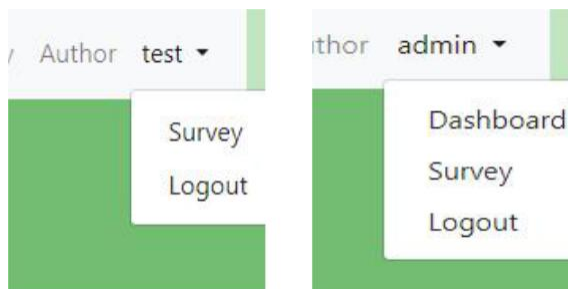
Specifično za polje za unos email-a, validacija se vrši na način da korisnik može uneti samo vrednost sa "@" karakterom. Ako korisnik pokuša da unese email bez toga, pojaviće se poruka o grešci "The email is invalid", obaveštavajući korisnika da unese ispravan email.

U slučaju da korisnik unese pogrešne podatke, tj. pogrešan email ili lozinku, iznad obrasca za prijavu će se prikazati obaveštenje o grešci "Wrong credentials". To će korisnika obavestiti da je došlo do greške prilikom prijavljivanja i da uneti podaci nisu ispravni.

Prilikom logovanja, korisnik ima pravo na tri pokušaja da unese ispravne kredencijale (korisničko ime i email). Ako korisnik tri puta unese pogrešne kredencijale, dugme za logovanje će postati "disabled" narednih 30 sekundi, što znači da će korisnikov nalog biti zaključan u tom periodu.



Kada korisnik uspešno unese sve ispravne podatke za logovanje, na stranici page=index će se prikazati poruka koja obaveštava korisnika o uspešnom logovanju. Ova poruka će biti vidljiva korisniku tokom 3 sekunde, nakon čega će nestati.



U navigacionom meniju, umesto opcija za login i registraciju, pojaviće se padajuća lista sa korisničkim imenom. Korisnik će imati mogućnost izbora između opcija kao što su survey (anketa) i logout (odjavljivanje). Kada admin korisnik izvrši uspešno logovanje, situacija je identična, ali će u padajućoj listi biti dodatno dostupne opcije za dashboard.php (administratorsku kontrolnu tablu) i ?page=survey (anketu).

Kada se admin korisnik loguje, biće automatski preusmeren na ?page=dashboard stranicu, gde će mu se prikazati poruka o uspešnom logovanju.

2.3 Register.php

Prilikom registracije novog korisnika, pored identične provere kao za logovanje, implementirana je provera da li se uneta lozinka i potvrda lozinke podudaraju. Ako se lozinke ne poklapaju, sistem će prikazati grešku koja obaveštava korisnika da se unete lozinke ne podudaraju.

REGISTER
Please enter your username and password!

Email already exists.

Your Name

test@gmail.com
Your Email

Password

Repeat your password

Register

Također, vrši se provjera da li uneti email već postoji u bazi podataka. Ako se uneti email već koristi, pri kliku na dugme "Register" prikazaće se obaveštenje o grešci iznad obrasca sa porukom "Email already exists". Ovo obaveštenje korisnika upozorava da izabere drugi email kako bi se uspešno registrovao.

2.4 Contact stranica

webio-php.000webhostapp.com says
Message could not be sent. Mailer Error: Invalid address: (to):
OK

Your email:
Enter email

Subject:
Enter subject

Paragraph
B I

This is some sample content.

Submit

Stranica za kontakt omogućava korisnicima da pošalju e-mail administratoru sa sugestijama, pitanjima ili bilo kojim drugim relevantnim informacijama. Ukoliko se ne popune navedena polja, prikazace se greska kao na slici. Odradjeno preko PHPMailer-master.

2.5 Admin dashboard.php

The screenshot shows the Admin dashboard with the following components:

- Left Sidebar:** Contains three links: [Manage Posts](#), [Manage Users](#), and [Manage Topics](#).
- Survey answers from users:** A table with columns **Email** and **Answer**.
- Survey Answer Statistics:** A green box containing:
 - Most selected answer:
 - Number of selections: 0
- Page access:** A light green box containing a list of access logs:
 - 2023-06-05 22:31:34 - Page accessed: index
 - 2023-06-05 22:31:38 - Page accessed: dashboard
 - 2023-06-05 22:32:25 - Page accessed: dashboard
 - 2023-06-06 20:19:59 - Page accessed: index
 - 2023-06-06 20:20:01 - Page accessed: contact
 - 2023-06-06 20:20:14 - Page accessed: author
 - 2023-06-06 20:20:15 - Page accessed: gallery
 - 2023-06-06 20:20:16 - Page accessed: index
- Page Access Statistics:** A light green box containing a list of page access percentages:
 - index: 31.67%
 - dashboard: 6.67%
 - contact: 6.67%
 - author: 1.67%
 - gallery: 1.67%
 - login: 23.33%
 - register: 1.67%
 - logout: 3.33%
- The number of users who logged in during the day:** Ukupan broj logovanih korisnika: 15

Nakon uspešne prijave administratora, na ovoj stranici administrator ima sveobuhvatan uvid u podatke i funkcionalnosti sajta.

Prvo, administrator ima pristup odgovorima korisnika na anketu. Može pregledati sve pristigle odgovore i dobiti uvid u statistiku o najčešće izabranim odgovorima. Na taj način administrator ima uvid u trendove i preferencije korisnika.

Administrator ima uvid u broj korisnika koji su se tog dana ulogovali na sajt.

Takođe, administrator ima uvid u statistiku pristupa svim stranicama ovog sajta. Ovde je odradjeno stranicenje (paginacija).

Pored toga, ima mogućnost izvršavanja CRUD operacija nad temama (topics), korisnicima (users) i objavama (posts). Ovo omogućava administratoru da upravlja sadržajem sajta, uređuje postojeće teme, dodaje nove teme, upravlja korisničkim nalogima i moderiše objave korisnika.

2.6 Validirane forme

Name is required.

Name

Description

Add Topic

Manage Topics

Failed to upload image.

Novi post

Title

Neki tekst

Content

Image

Choose File No file chosen

Topic eCommerce

☐ Publish

Add User

Email is required.

Password is required.

Name

admin

Email

Password

Password confirmation

☒ Admin

Edit Users

Passwords is not match.

Email already exists.

admin

Name

admin@admin

Email

Password

Password confirmation

☒ Admin

Add Post
Manage Posts

Edit Post

Post image required.

Title

Business website

Content

It's no longer feasible to run a business, even a brick-and-mortar one, without a web presence. Consumers turn to the internet for everything from product research to location and operating hours. Even just a simple, well-designed website can give you an edge in your field, and if you have products to sell, your site can open up new markets and expand your business cheaply and easily.

Image

Choose File No file chosen

Topic

Business

☒ Publish

Update post

Title is required.
Content is required.

Send us a query!

Title

Content

Send

Add Topic

Manage Topics

Topic updated successfully.

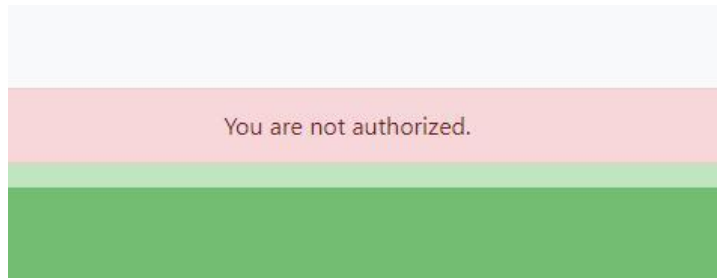
SN	Name	Action
1	eCommerce	<div>Edit</div> <div>Delete</div>
2	Personal	<div>Edit</div> <div>Delete</div>

1
2
3
»

Sve forme koje su dostupne administratoru na stranicama imaju ugrađenu validaciju kako bi se obezbedilo da se ispravni podaci unose na pravilan način. U slučaju da administrator propusti da unese neki od obaveznih podataka ili unese neispravne vrednosti, prikazuju se jasne poruke o greškama na formi.

S druge strane, kada se sve vrednosti uspešno unesu i validiraju, prikazuje se poruka o uspešnoj operaciji. Ova poruka se takođe prikazuje na formi, ali sa pozitivnom povratnom informacijom. Da bi se osiguralo da poruka bude primetna, ona se zadržava na ekranu tokom 3 sekunde pre nego što nestane. Ovo omogućava administratoru da jasno vidi da je operacija uspešno izvršena pre nego što se poruka automatski sakrije.

2.7 Autorizacija



U slučaju da korisnik ili gost pokuša da pristupi bilo kojoj od administratorskih stranica putem URL adrese, sistem će ih automatski preusmeriti na index.php stranicu sa porukom "You are not authorized" (Niste autorizovani). Ova poruka obaveštava korisnika da nema odgovarajuće ovlašćenje za pristup administratorskim stranicama.

3.0 KODOVI

api/ check_login_attempts:

```
<?php
require "../models/users.php";

if ($_SERVER["REQUEST_METHOD"] == 'POST') {
    $username = $_POST['username'];
    $user = getUserByUsername($username);
    if ($user) {
        $failedAttempts = getFailedLoginAttempts($user['id']);
        if ($failedAttempts >= 3) {
            echo json_encode(['locked' => true]);
            exit();
        }
    }
    echo json_encode(['locked' => false]);
}
```

Api/ contact.php

```
<?php
require "../helpers/mail.php";

if ($_SERVER["REQUEST_METHOD"] == 'POST') {
    $userEmail = $_POST["userEmail"];
    $subject = $_POST["subject"];
    $body = $_POST["body"];
    echo Mail::sendMail($userEmail, $subject, $body);
}
?>
```

app/database/connect.php:

```
<?php
$db_host = 'localhost';
$db_name = 'id20728324_webio';
$db_user = 'id20728324_nevena';
$db_password = 'Lozinka123!';

$dsn = "mysql:host=$db_host;dbname=$db_name";
try {
    $conn = new PDO($dsn, $db_user, $db_password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    echo "Connected error: " . $e->getMessage();
}
?>
```

Data/access.log

```
2023-06-05 22:31:34 - Page accessed: index
2023-06-05 22:31:38 - Page accessed: dashboard
2023-06-05 22:32:25 - Page accessed: dashboard
2023-06-06 20:19:59 - Page accessed: index
2023-06-06 20:20:01 - Page accessed: contact
2023-06-06 20:48:13 - Page accessed: index
```

Data/login.log

```
2023-06-06 20:13:57 - User logged in:
2023-06-06 20:14:15 - User logged in:
2023-06-06 20:14:18 - User logged in:
2023-06-06 20:19:18 - User logged in:
2023-06-06 20:20:17 - User logged in:
```

Helpers/mail.php:

```
<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;
use PHPMailer\PHPMailer\SMTP;

require __DIR__ ."/../PHPMailer-master/src/Exception.php";
require __DIR__ ."/../PHPMailer-master/src/PHPMailer.php";
require __DIR__ ."/../PHPMailer-master/src/SMTP.php";

class Mail
{
    public static function sendMail($userEmail, $subject, $body)
    {
        $mail = new PHPMailer(true);
        try {
```

```

$mail->SMTPDebug = SMTP::DEBUG_OFF;
$mail->SMTPOptions = array(
    'ssl' => array(
        'verify_peer' => false,
        'verify_peer_name' => false,
        'allow_self_signed' => true
    )
);
$mail->CharSet = 'utf-8';
$mail->isSMTP();
$mail->Host = 'smtp.gmail.com';
$mail->SMTPAuth = true;
$mail->Username = 'nevena.nikolic.174.21@ict.edu.rs';
$mail->Password = '';
$mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
$mail->Port = 587;
$mail->setFrom('your_email@example.com', 'Nevena Nikolic 174/21');
$mail->addAddress($userEmail);
$mail->isHTML(true);
$mail->Subject = $subject;
$mail->Body =
    "
    $body
    <hr/>
    Reply to email: $userEmail.
    ";
$mail->send();
return 'Message has been sent';
} catch (Exception $e) {
    return "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
}
}
}

```

/models/attempt_logins.php

```
<?php
include_once "db.php";
function trackLoginAttempt($userId) {
    global $conn;

    $stmt = $conn->prepare('INSERT INTO login_attempts (user_id) VALUES (:userId)');
    $stmt->bindParam(':userId', $userId);
    $stmt->execute();
}
function lockAccount($userId) {
    global $conn;

    $stmt = $conn->prepare('UPDATE users SET status = "locked" WHERE id = :userId');
    $stmt->bindParam(':userId', $userId);
    $stmt->execute();
}
?>
```

/models/checkEmailAvailability.php

```
<?php
require_once('db.php');
if(isset($_POST['email'])) {
    $email = $_POST['email'];
    $existingUser = selectOne('users', ['email' => $email]);
    if($existingUser) {
        echo "Email already exists.";
    } else {
        echo "Email available.";
    }
}
?>
```

/models/db.php

```
<?php

include __DIR__ . '/../app/database/connect.php';
include "helpers/mail.php";
function executeQuery($query, $data){
    global $conn;
    $sql = $conn->prepare($query);
    $sql->execute($data);
    return $sql;
}

function selectAll($table, $conditions = []) {
    global $conn;
    $query = "SELECT * FROM $table";
    $params = [];
    if(!empty($conditions)){
        $query .= " WHERE ";
        $i = 0;
        foreach($conditions as $key => $value) {
            if($i > 0) {
                $query .= " AND ";
            }
            $query .= "$key=?";
            $params[] = $value;
            $i++;
        }
    }
    $stmt = $conn->prepare($query);
    $stmt->execute($params);
    $records = $stmt->fetchAll(PDO::FETCH_ASSOC);
    return $records;
}

function selectOne($table, $conditions = []) {
    global $conn;
    $query = "SELECT * FROM $table WHERE ";
    $params = [];
    $i = 0;
    foreach($conditions as $key => $value) {
        if($i > 0) {
            $query .= " AND ";
        }
        $query .= "$key=?";
        $params[] = $value;
        $i++;
    }
    $query .= " LIMIT 1";
    $stmt = $conn->prepare($query);
    $stmt->execute($params);
    $record = $stmt->fetch(PDO::FETCH_ASSOC);
    return $record;
}
```

```

function insertOperation($table, $data) {
    global $conn;
    $columns = array_keys($data);
    $placeholders = array_fill(0, count($data), '?');
    $query = "INSERT INTO $table (" . implode(',', $columns) . ") VALUES (" .
implode(',', $placeholders) . ")";
    $stmt = $conn->prepare($query);
    $stmt->execute(array_values($data));
    $id = $conn->lastInsertId();
    return $id;
}

function updateOperation($table, $id, $data) {
    global $conn;
    $columns = array_keys($data);
    $placeholders = array_fill(0, count($data), '?');
    $data['id'] = $id;
    $setClause = implode('=?,', $columns) . '=?';
    $query = "UPDATE $table SET $setClause WHERE id=?";
    $stmt = $conn->prepare($query);
    $stmt->execute(array_values($data));
    $affectedRows = $stmt->rowCount();
    return $affectedRows;
}

function deleteOperation($table, $id) {
    global $conn;
    $query = "DELETE FROM $table WHERE id=?";
    $stmt = $conn->prepare($query);
    $stmt->execute([$id]);
    $affectedRows = $stmt->rowCount();
    return $affectedRows;
}

function getPublishedPosts() {
    global $conn;
    $query = "SELECT p.*, u.username
        FROM posts AS p
        JOIN users AS u
        ON p.user_id = u.id
        WHERE p.published = ?";
    $stmt = $conn->prepare($query);
    $stmt->execute([1]);
    $records = $stmt->fetchAll(PDO::FETCH_ASSOC);

    return $records;
}

function getPostsbyTopicId($topic_id) {
    global $conn;
    $query = "SELECT p.*, u.username
        FROM posts AS p
        JOIN users AS u
        ON p.user_id = u.id
        WHERE p.published = ?
        AND topic_id = ?";
    $sql = executeQuery($query, [1, $topic_id]);

```

```

    $records = $sql->fetchAll(PDO::FETCH_ASSOC);
    return $records;
}

function searchPosts($words) {
    $match = '%' . $words . '%';
    global $conn;
    $query = "SELECT p.*, u.username
              FROM posts AS p
              JOIN users AS u
              ON p.user_id = u.id
              WHERE p.published = ?
              AND (p.title LIKE ? OR p.content LIKE ?)";

    $sql = executeQuery($query, [1, $match, $match]);

    $records = $sql->fetchAll(PDO::FETCH_ASSOC);
    return $records;
}

function paginate($table, $page, $limit) {
    global $conn;
    $page = intval($page);
    $offset = ($page - 1) * $limit;
    $stmt = $conn->prepare("SELECT COUNT(*) AS total FROM $table");
    $stmt->execute();
    $result = $stmt->fetch();
    $total = $result->total;
    $totalPages = ceil($total / $limit);
    $stmt = $conn->prepare("SELECT * FROM $table LIMIT :offset, :limit");
    $stmt->bindValue(':offset', $offset, PDO::PARAM_INT);
    $stmt->bindValue(':limit', $limit, PDO::PARAM_INT);
    $stmt->execute();
    $data = $stmt->fetchAll(PDO::FETCH_ASSOC);
    return ['data' => $data, 'totalPages' => $totalPages];
}

function getUsernameById($userId) {
    global $conn;
    $sql = "SELECT username FROM users WHERE id = :userId";
    $stmt = $conn->prepare($sql);
    $stmt->bindParam(':userId', $userId, PDO::PARAM_INT);
    $stmt->execute();
    $result = $stmt->fetch(PDO::FETCH_ASSOC);
    return $result['username'] ?? 'Unknown';
}

function getPageAccessStatistics($accessLog) {
    $logContent = file_get_contents($accessLog);
    $logLines = explode("\n", $logContent);
    $totalLines = count($logLines);
    $pageCounts = array();

    foreach ($logLines as $line) {
        $matches = array();
        if (preg_match('/Page accessed: (\w+)/', $line, $matches)) {
            $page = $matches[1];

```



```

        if (isset($pageCounts[$page])) {
            $pageCounts[$page]++;
        } else {
            $pageCounts[$page] = 1;
        }
    }
}

$pageStatistics = array();
foreach ($pageCounts as $page => $count) {
    $percentage = round(($count / $totalLines) * 100, 2);
    $pageStatistics[$page] = $percentage;
}

return $pageStatistics;
}

function getLoggedInUsersCount($loginLog) {
    $logContent = file_get_contents($loginLog);
    $logLines = explode("\n", $logContent);
    $loggedInUsersCount = 0;

    foreach ($logLines as $line) {
        if (strpos($line, "User logged in:") !== false) {
            $loggedInUsersCount++;
        }
    }

    return $loggedInUsersCount;
}

function getFailedLoginAttempts($userId)
{
    global $conn;
    $query = "SELECT failed_login_attempts, last_failed_login, lock_reset_time
FROM users WHERE id = :userId";
    $statement = $conn->prepare($query);
    $statement->bindValue(':userId', $userId);
    $statement->execute();
    $row = $statement->fetch(PDO::FETCH_ASSOC);
    if ($row) {
        $lockResetTime = strtotime($row['lock_reset_time']);
        if ($lockResetTime > time()) {
            return 0;
        }
        return $row['failed_login_attempts'];
    }
    return 0;
}

function incrementFailedLoginAttempts($userId)
{
    global $conn;
    $failedLoginAttempts = getFailedLoginAttempts($userId);

```

```

        if ($failedLoginAttempts >= 2) {
            $lockResetTime = time() + 30;
            updateLockResetTime($userId, $lockResetTime);
            lockAccount($userId);
        } else {
            $query = "UPDATE users SET failed_login_attempts =
failed_login_attempts + 1 WHERE id = :userId";
            $statement = $conn->prepare($query);
            $statement->bindValue(':userId', $userId);
            $statement->execute();
        }
    }
}

function lockAccount($userId)
{
    global $conn;
    $lockResetTime = date('Y-m-d H:i:s', strtotime('+30 seconds'));
    $query = "UPDATE users SET failed_login_attempts = 0, locked = 1,
lock_reset_time = :lockResetTime WHERE id = :userId";
    $statement = $conn->prepare($query);
    $statement->bindValue(':lockResetTime', $lockResetTime);
    $statement->bindValue(':userId', $userId);
    $statement->execute();
}

function sendLockoutEmail($email)
{
    $subject = "Account Lockout Notification";
    $message = "Your account has been locked due to multiple failed login
attempts. Please try again after 30 seconds.";
    try {
        Mail::sendMail($email, $subject, $message);
        return 'Email sent successfully.';
    } catch (Exception $e) {
        return 'Failed to send email: ' . $e->getMessage();
    }
}

function updateLockResetTime($userId, $resetTime) {
    global $conn;

    $formattedResetTime = date('Y-m-d H:i:s', $resetTime);
    $query = "UPDATE users SET lock_reset_time = :resetTime WHERE id
= :userId";
    $statement = $conn->prepare($query);
    $statement->bindValue(':resetTime', $formattedResetTime);
    $statement->bindValue(':userId', $userId);
    $statement->execute();
}

function resetFailedLoginAttempts($userId) {
    global $conn;
    $query = "UPDATE users SET failed_login_attempts = 0 WHERE id = :userId";
    $statement = $conn->prepare($query);
    $statement->bindValue(':userId', $userId);
    $statement->execute();
}

```

```

function startTimer($seconds){
    echo '<script>
        $(document).ready(function() {
            disableLoginButton();
            var timer = setInterval(function() {
                if (' . $seconds . ' <= 0) {
                    clearInterval(timer);
                    resetLoginAttempts();
                    enableLoginButton();
                } else {
                    $("#timer-message").text("Please try again after " + ' .
$seconds . ' + " seconds.");
                    ' . $seconds . '--;
                }
            }, 1000);
        });
    </script>';
}

function getTotalLoggedInUsersCount($logFile) {
    $logContent = file_get_contents($logFile);
    $logLines = explode("\n", $logContent);
    $loggedInUsers = 0;
    foreach ($logLines as $line) {
        if (!empty($line) && strpos($line, "User logged in: ") !== false) {
            $loggedInUsers++;
        }
    }
    return $loggedInUsers;
}

function logLogin($userId, $logFile) {
    $logMessage = date("Y-m-d H:i:s") . " - User logged in: " . $userId . "\n";
    file_put_contents($logFile, $logMessage, FILE_APPEND);
}

```

/models/formErrors.php

```
<?php

if(count($errors) > 0): ?>
    <?php foreach($errors as $error):?>
        <p class="alert alert-danger"><?php echo $error; ?></p>
    <?php endforeach;?>
<?php endif; ?>
```

/models/messages.php

```
<?php if(isset($_SESSION['message'])): ?>
<div class="msg" style="margin: 0px auto; width: 50%; text-align:center;">
    <p class="<?php echo $_SESSION['type']; ?>">
        <?php echo $_SESSION['message'];?>
    </p>
</div>
<?php
    unset($_SESSION['message']);
    unset($_SESSION['type']);
?>

<?php endif;?>
```

/models/posts.php

```
<?php
include_once "db.php";
include_once "validatePosts.php";

$table = 'posts';
$topics = selectAll('topics');
$posts = selectAll($table);
$errors = array();
$id = '';
$title = '';
$content = '';
$topic_id = '';
$published = '';
if(isset($_GET['id'])){

    $post = selectOne($table, ['id' => $_GET['id']]);
    $id = $post['id'];
    $title = $post['title'];
    $content = $post['content'];
    $topic_id = $post['topic_id'];
    $published = $post['published'];
}
//add post
if(isset($_POST['add-post'])){
```

```

$errors = validatePost($_POST);
if(!empty($_FILES['image'])) {
    $image_name = time() . '_' . $_FILES['image']['name'];
    //varDumps($_FILES['image']['name']);
    $image_src = 'assets/images/' . $image_name;
    $result = move_uploaded_file($_FILES['image']['tmp_name'], $image_src);
    if($result) {
        $_POST['image'] = $image_name;
    } else{
        array_push($errors, 'Failed to upload image.');
```

```

        $_POST['content'] = htmlentities($_POST['content']);

        $post_id = updateOperation($table, $id, $_POST);
        $_SESSION['message'] = 'Post updated successfully.';
        $_SESSION['type'] = 'alert alert-success';
        echo '<script>window.location.href = "index.php?page=index-
post";</script>';
    } else{
        $title = $_POST['title'];
        $content = $_POST['content'];
        $topic_id = $_POST['topic_id'];
        $published = isset($_POST['published']) ? 1 : 0;
    }
}
//delete post
if(isset($_GET['delete_id'])){

    $count = deleteOperation($table, $_GET['delete_id']);
    $_SESSION['message'] = 'Post deleted successfully.';
    $_SESSION['type'] = 'alert alert-success';
    echo '<script>window.location.href = "../index.php?page=index-
post";</script>';
    exit();
}
if(isset($_GET['published']) && isset($_GET['p_id'])){

    $published = $_GET['published'];
    $p_id = $_GET['p_id'];
    $count = updateOperation($table, $p_id, ['published' => $published]);
    $_SESSION['message'] = 'Post published state changed.';
    $_SESSION['type'] = 'alert alert-success';
    echo '<script>window.location.href = "../index.php?page=index-
post";</script>';
    exit();
}
?>

```

/models/topics.php

```
<?php
include_once "db.php";
include_once "validateTopics.php";

$table = 'topics';
$topics = selectAll($table);
$errors = array();
$id = '';
$name = '';
$description = '';
if(isset($_POST['add-topic'])) {

    $errors = validateTopic($_POST);
    if(count($errors) === 0){
        unset($_POST['add-topic']);

        $topic_id = insertOperation('topics', $_POST);
        $_SESSION['message'] = 'Topic created successfully.';
        $_SESSION['type'] = 'alert alert-success';
        echo '<script>window.location.href = "index.php?page=index-
topic";</script>';
        exit();
    } else {

    }

}
if(isset($_GET['id'])) {
    $id = $_GET['id'];
    $topic = selectOne($table, ['id' => $id]);
    $id = $topic['id'];
    $name = $topic['name'];
    $description = $topic['description'];
}
if(isset($_POST['update-topic'])) {

    $errors = validateTopic($_POST);
    if(count($errors) === 0){
        $id= $_POST['id'];
        unset($_POST['id'], $_POST['update-topic']);
        $topic_id = updateOperation($table, $id, $_POST);
        $_SESSION['message'] = 'Topic updated successfully.';
        $_SESSION['type'] = 'alert alert-success';
        echo '<script>window.location.href = "index.php?page=index-
topic";</script>';
        exit();
    } else {
        $id = $_POST['id'];
        $name = $_POST['name'];
        $description = $_POST['description'];
    }
}
```

```

}
if(isset($_GET['del_id'])){
    $id = $_GET['del_id'];
    $count = deleteOperation($table, $id);
    $_SESSION['message'] = 'Topic deleted successfully.';
    $_SESSION['type'] = 'alert alert-success';
    echo '<script>>window.location.href = "../index.php?page=index-
topic";</script>';
    exit();
}
?>

```

/models/users.php

```

<?php

include_once "db.php";
include_once "validateUser.php";
$table = 'users';
$admin_users = selectAll($table);
$errors = array();
$id = '';
$username = '';
$role = '';
$email = '';
$password = '';
$passConfirm = '';

function loginUser($user) {

    $_SESSION['id'] = $user['id'];
    $_SESSION['username'] = $user['username'];
    $_SESSION['role'] = $user['role'];
    $_SESSION['message'] = 'You are now logged in.';
    $_SESSION['type'] = 'alert alert-success';
    if($_SESSION['role']){
        echo '<script>>window.location.href = "index.php?page=dashboard";</script>';
    }
    else{
        echo '<script>>window.location.href = "index.php?page=index&u_id=' .
$user['id'] . '";</script>';
    }
    exit();
}

if (isset($_POST['register-btn'])) {

    $errors = validateUser($_POST);
    if (count($errors) === 0) {
        unset($_POST['register-btn'], $_POST['password_confirmation'],
$_POST['add-admin']);
        $_POST['password'] = password_hash($_POST['password'],
PASSWORD_DEFAULT);
        $existingUser = selectOne($table, ['email' => $_POST['email']]);
    }
}

```



```

        if (!$existingUser) {
            $_POST['role'] = 0;
            $user_id = insertOperation($table, $_POST);
            $_SESSION['message'] = "User registered and logged in
successfully.";
            $_SESSION['type'] = 'alert alert-success';
            $user = selectOne($table, ['id' => $user_id]);
            loginUser($user);
        } else {
            $email = $_POST['email'];
            array_push($errors, 'Email already exists.');
```

```

        $id = $_POST['id'];
        unset($_POST['password_confirmation'], $_POST['update-user'],
$_POST['id']);

        $_POST['password'] = password_hash($_POST['password'],
PASSWORD_DEFAULT);
        $_POST['role'] = isset($_POST['role']) ? 1 : 0;
        $count = updateOperation($table, $id, $_POST);
        $_SESSION['message'] = "Admin user updated successfully.";
        $_SESSION['type'] = 'alert alert-success';
        echo "<script>window.location.href = 'index.php?page=index-
user';</script>";
        exit();

    } else{
        $username = $_POST['username'];
        $role = isset($_POST['role'])? 1 : 0;
        $email = $_POST['email'];
        $password = $_POST['password'];
        $passConfim = $_POST['password_confirmation'];
    }
}
if(isset($_GET['id'])) {
    $user = selectOne($table, ['id' => $_GET['id']]);

    $id = $user['id'];
    $username = $user['username'];
    $role = $user['role'];
    $email = $user['email'];
}
if(isset($_POST['login-btn'])) {
    $errors = validateLogin($_POST);
    if(count($errors) === 0){
        $user = selectOne($table, ['email' => $_POST['email']]);
        if($user && password_verify($_POST['password'], $user['password'])) {
            loginUser($user);
        }
        else{
            array_push($errors, 'Wrong credentials.');
```

/models/usersType.php

```
<?php

function adminOnly($redirect = 'index.php?page=index') {
    if(empty($_SESSION['id']) || $_SESSION['role'] != 1) {
        $_SESSION['message'] = 'You are not authorized.';
        $_SESSION['type'] = 'alert alert-danger';
        echo '<script>window.location.href = "' . $redirect . '";</script>';
        exit(0);
    }
}

function guestsOnly($redirect = 'index.php?page=index') {
    if(isset($_SESSION['id'])) {
        echo '<script>window.location.href = "' . $redirect . '";</script>';
        exit(0);
    }
}

function usersOnly($redirect = 'index.php?page=index') {
    if(!isset($_SESSION['id'])) {
        $_SESSION['message'] = 'You are not authorized. Please register or
login first.';
        $_SESSION['type'] = 'alert-danger';
        echo '<script>window.location.href = "' . $redirect . '";</script>';
        exit(0);
    }
}

?>
```

/models/validatePosts.php

```
<?php

function validatePost($post){
    $errors = array();

    if(empty($post['title'])){
        array_push($errors, 'Title is required.');
```

```
    }
    if(empty($post['content'])){
        array_push($errors, 'Content is required.');
```

```
    }
    if(empty($post['topic_id'])){
        array_push($errors, 'Please select a topic.');
```

```
    }

    $existingPost = selectOne('posts', ['title' => $post['title']]);
    if($existingPost){
        if(isset($post['update-post']) && $existingPost['id'] !=
$post['id']){
            array_push($errors, 'Post with that title already exists.');
```

```
        }
    }
}
```

```

        if(isset($post['add-post'])) {
            array_push($errors, 'Post with that title already exists.');
```

```

        }
    }

    return $errors;
}
?>
```

/models/validateTopics.php

```

<?php

function validateTopic($topic){
    $errors = array();

    if(empty($topic['name'])){
        array_push($errors, 'Name is required.');
```

```

    }
    $existingTopic = selectOne('topics', ['name' => $topic['name']]);
    if($existingTopic){
        if(isset($topic['update-topic']) && $existingTopic['id'] !=
$topic['id']){
            array_push($errors, 'Name already exists.');
```

```

        }
        if(isset($topic['add-topic'])) {
            array_push($errors, 'Name already exists.');
```

```

        }
    }

    return $errors;
}
?>
```

/models/validateUser.php

```

<?php

function validateUser($user){
    $errors = array();
    if(empty($user['username'])){
        array_push($errors, 'Username is required.');
```

```

    }
    if(empty($user['email'])){
        array_push($errors, 'Email is required.');
```

```

    }
    if(empty($user['password'])){
        array_push($errors, 'Password is required.');
```

```

    }
    if($user['password_confirmation'] != $user['password']) {
```

```

        array_push($errors, 'Passwords is not match.');
```

```

    }
    $existingUser = selectOne('users', ['email' => $user['email']]);
    if($existingUser){
        if(isset($user['update-user']) && $existingUser['id'] !=
$user['id']){
            array_push($errors, 'Email already exists.');
```

```

        }
        if(isset($user['add-admin'])) {
            array_push($errors, 'Email already exists.');
```

```

        }
    }
}
return $errors;
}
function validateLogin($user){
    $errors = array();

    if(empty($user['email'])){
        array_push($errors, 'Email is required.');
```

```

    }

    if(empty($user['password'])){
        array_push($errors, 'Password is required.');
```

```

    }

    return $errors;
}
?>

```

/views/fixed/adminSidebar.php

```

<div class="left-sidebar">
    <ul>
        <li><a href="index.php?page=index-post">Manage Posts</a></li>
        <li><a href="index.php?page=index-user">Manage Users</a></li>
        <li><a href="index.php?page=index-topic">Manage Topics</a></li>
    </ul>
</div>

```

/views/fixed/footer.php

```
<section class="container-fluid footer_section py-4" style="background-color:
#72bd72;">
  <p class="text-center text-uppercase text-white mb-0">
    &copy; <?php echo date("Y"); ?> All Rights Reserved by: Nevena Nikolic
174/21
    <a href="https://www.ict.edu.rs/" class="text-white">ICT Visoka skola</a>
  </p>
</section>

  <script src="https://code.jquery.com/jquery-3.6.4.min.js" integrity="sha256-
oP6HI9z1XaZNBzJURtCoUT5SUnxFr8s3BzRl+cbzUq8="
crossorigin="anonymous"></script>
  <script type="text/javascript" src="assets/js/bootstrap.js"></script>
  <script src="assets/js/main.js"></script>
</body>
</html>
```

/views/fixed/header.php

```
<?php
$navigacije = selectAll('navigations');
?>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no" />
    <meta name="keywords" content="" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title> WEBIO</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.1.3/assets/owl.carousel.min.css" />
    <link rel="stylesheet" type="text/css" href="assets/css/bootstrap.css" />
    <link
href="https://fonts.googleapis.com/css?family=Baloo+Chettan|Dosis:400,600,700|
Poppins:400,600,700&display=swap" rel="stylesheet" />
    <link href="assets/css/style.css" rel="stylesheet" />
    <link rel="stylesheet" type="text/css" href="assets/css/admin.css" />
  </head>
  <body class="sub_page">
    <header class="header_section">
      <div class="container-fluid">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <a id="logo" class="navbar-brand" href="#">Webio Blog</a>
```

```

        <a id="dokumentacija" class="navbar-brand" href="dokumentacija-
praktikumPHP.pdf">Documentation</a>
        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-
expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
            <ul class="navbar-nav ml-auto">
                <?php foreach($navigacije as $navigacija): ?>
                    <li class="nav-item">
                        <a class="nav-link" href="index.php?page=<?php echo
$navigacija['href'];?>"><?php echo $navigacija['title'];?></a>
                    </li>
                <?php endforeach; ?>
                <?php if(isset($_SESSION['id'])): ?>
                    <li class="nav-item dropdown">
                        <a class="nav-link dropdown-toggle" href="#"
id="navbarDropdownMenuLink" role="button" data-toggle="dropdown" aria-
haspopup="true" aria-expanded="false">
                            <?php echo $_SESSION['username'];?>
                        </a>
                        <div class="dropdown-menu" aria-
labelledby="navbarDropdownMenuLink">
                            <?php if($_SESSION['role']): ?>
                                <a class="dropdown-item"
href="index.php?page=dashboard">Dashboard</a>
                                <?php endif; ?>
                                <a class="dropdown-item"
href="index.php?page=survey&u_id=<?php echo $_SESSION['id']; ?>">Survey</a>
                                <a class="dropdown-item"
href="index.php?page=logout">Logout</a>
                            </div>
                        </li>
                    <?php else: ?>
                        <li class="nav-item">
                            <a class="nav-link" href="index.php?page=login">Login</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="index.php?page=register">Register</a>
                        </li>
                    <?php endif; ?>
                </ul>
            </div>
        </nav>
    </header>

```

/views/pages/author.php

```
<div class="row">
  <div class="col-md-6">
    
  </div>
  <div class="col-md-6">
    <p>&#60;hello&#62;</p>
    <p class="py-2 para-light" style="margin-right:20rem;">
      I am Nevena Nikolić, a 23 year old student that is currently
studying
      Web programming module on the Internet technologies, a part of the
ICT College in Serbia.
      I come from Pancevo, a city near the capital city of Serbia
Belgrade, where i finished
      elementary and high school. By enrolling in ICT my decision to
tinker with computers was
      my way to set a stable future for me, right now in the specter of
Web, and who knows what
      tomorrow. By now i have 3 projects that helped me hone the
knowledge that i currently hold,
      and use on my websites.
    <br><br> &#60;/hello&#62;
  </p>
</div>
</div>
```

/views/pages/contact.php

```
<section class="vh-100 gradient-custom">
  <div class="py-5">
    <div class="row justify-content-center align-items-center h-100">
      <div class="col-md-8 col-lg-6 col-xl-5">
        <div class="card">
          <div class="card-body">
            <h2 class="card-title text-center mb-4">Send us a query!</h2>

            <form validate>
              <div class="form-group">
                <label for="userEmail">Your email:</label>
                <input type="email" class="form-control" id="userEmail"
placeholder="Enter email" name="userEmail">
              </div>
              <div class="form-group">
                <label for="subject">Subject:</label>
                <input type="text" class="form-control" id="subject"
placeholder="Enter subject" name="subject">
              </div>
              <div class="form-group">
                <div id="email-body-editor">This is some sample content.</div>
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
```



```

        <button type="submit" class="btn btn-default" name="send-query"
id="send-query">Submit</button>
    </form>
</div>
</div>
</div>
</div>
</div>
</div>
</section>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script
src="https://cdn.ckeditor.com/ckeditor5/34.0.0/classic/ckeditor.js"></script>
<script>
$(document).ready(function() {
    let editor;
    ClassicEditor
        .create(document.querySelector('#email-body-editor'))
        .then(newEditor => {
            editor = newEditor;
        })
        .catch(error => {
            console.error(error);
        });
    $(document).on("submit", "form", (e) => {
        e.preventDefault();
        $.ajax({
            url: "api/contact.php",
            method: "POST",
            data: {
                userEmail: $("#userEmail").val(),
                subject: $("#subject").val(),
                body: editor.getData()
            },
            success: function(result) {
                alert(result);
            },
            error: function(err) {
                alert("Error!");
                console.log(err);
            }
        });
    });
});
</script>

```



```
<option value="<?php echo $topic['id'] ?>"><?php echo $topic['name'] ?></option>
    <?php endif; ?>
    <?php endforeach; ?>
</select>
</div>
<div class="form-check mb-3">
    <input class="form-check-input" type="checkbox"
id="published" name="published" <?php echo !empty($published) ? 'checked' :
'; ?>>
        <label class="form-check-label"
for="published">Publish</label>
    </div>
    <div class="d-grid gap-2">
        <button class="btn btn-primary btn-lg"
type="submit" name="add-post">Add post</button>
    </div>
</form>
</div>
<div class="card-footer">
    <a href="index.php?page=index-post" class="btn btn-
secondary">Manage Posts</a>
</div>
</div>
</div>
</div>
</section>
```

</views/pages/create-topic.php>

[illegible]

```

        <label for="description" class="form-
label">Description</label>
        <textarea id="description" class="form-control"
name="description" rows="8"><?php echo isset($description) ? $description :
''; ?></textarea>
    </div>
    <div class="d-grid">
        <button class="btn btn-primary btn-lg" id="addTopic"
type="submit" name="add-topic">Add Topic</button>
    </div>
</form>
<div id="rezultat"></div>
</div>
<div class="card-footer">
    <a href="index.php?page=index-topic" class="btn btn-
secondary">Manage Topics</a>
</div>
</div>
</div>
</div>
</div>
</section>

```

/views/pages/create-user.php

```

<?php
include "models/users.php";
adminOnly();
?>
<section class="vh-100 gradient-custom">
    <div class="container py-5">
        <div class="row justify-content-center align-items-center h-100">
            <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                <div class="card bg-light">
                    <div class="card-body">
                        <h2 class="card-title text-center">Add User</h2>

                        <?php include "models/formErrors.php"; ?>
                        <form action="index.php?page=create-user" method="post"
enctype="multipart/form-data" novalidate>
                            <div class="form-group">
                                <label for="username">Name</label>
                                <input type="text" id="username" class="form-control"
name="username" value="<?php echo isset($username) ? $username : ''; ?>">
                            </div>
                            <div class="form-group">
                                <label for="email">Email</label>
                                <input type="email" id="email" class="form-control"
name="email" value="<?php echo isset($email) ? $email : ''; ?>">
                            </div>
                            <div class="form-group">
                                <label for="password">Password</label>

```

```

        <input type="password" id="password" class="form-control"
name="password" value="<?php echo isset($password) ? $password : ''; ?>">
    </div>
    <div class="form-group">
        <label for="password_confirmation">Password
confirmation</label>
        <input type="password" id="password_confirmation" class="form-
control" name="password_confirmation" value="<?php echo isset($passConfim) ?
$passConfim : ''; ?>">
    </div>
    <div class="form-group form-check">
        <?php if (isset($role) && $role == 1): ?>
            <input type="checkbox" class="form-check-input" id="admin"
name="role" checked>
            <label class="form-check-label" for="admin">Admin</label>
        <?php else: ?>
            <input type="checkbox" class="form-check-input" id="admin"
name="role">
            <label class="form-check-label" for="admin">Admin</label>
        <?php endif; ?>
    </div>
    <div class="text-center">
        <button class="btn btn-primary btn-lg" name="add-admin"
type="submit">Add User</button>
    </div>
</form>
<div id="rezultat"></div>
<div class="card-footer">
    <a href="index.php?page=index-user" class="btn btn-
secondary">Manage Users</a>
</div>
</div>
</div>
</div>
</div>
</div>
</section>

```

```
<?php
    adminOnly();

    $sql1 = "SELECT users.email, survey.answer
            FROM user_answer
            INNER JOIN users ON users.id = user_answer.user_id
            INNER JOIN survey ON survey.id = user_answer.answer_id
            ORDER BY user_answer.id DESC";
    $stmt1 = $conn->prepare($sql1);
    $stmt1->execute();
    $results1 = $stmt1->fetchAll(PDO::FETCH_ASSOC);

    $accessLog = "data/access.log";
    if (file_exists($accessLog)) {
        $logContent = file_get_contents($accessLog);
    } else {
        $logContent = "Nema dostupnih podataka o pristupima.";
    }

    // paginacija
    $strana = isset($_GET['strana']) ? intval($_GET['strana']) : 1;
    $limit = 10;
    $logLines = explode("\n", $logContent);
    $totalLines = count($logLines);
    $totalPages = ceil($totalLines / $limit);
    $offset = ($strana - 1) * $limit;
    $logLines = array_slice($logLines, $offset, $limit);
    $paginatedLogContent = implode("\n", $logLines);

    $pageStatistics = getPageAccessStatistics($accessLog);
    $logFile = "data/login.log";
    $totalLoggedInUsers = getTotalLoggedInUsersCount($logFile);
?>
<section class="vh-100 gradient-custom" style="margin-bottom:20%;">
    <div class="py-5">
        <div class="row d-flex justify-content-center align-items-center h-100">
            <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                <div class="tabelaCreate">
                    <h2 class="page-title">Dashboard</h2>
                    <?php include "models/messages.php"; ?>
                    <div class="table-responsive" style="margin-top:10rem;">
                        <h5 class="page-title" style="margin:2rem;">Survey answers from
users:</h5>
                        <table class="table table-striped">
                            <thead>
                                <tr>
                                    <th>Email</th>
                                    <th>Answer</th>
                                </tr>
```

```

        </thead>
        <tbody>
            <?php foreach ($results1 as $result1) : ?>
                <tr>
                    <td><?= $result1['email'] ?></td>
                    <td><?= $result1['answer'] ?></td>
                </tr>
            <?php endforeach; ?>
        </tbody>
    </table>
</div>
<?php
$answerCounts = array();
foreach ($results1 as $result1) {
    $answer = $result1['answer'];
    if (isset($answerCounts[$answer])) {
        $answerCounts[$answer]++;
    } else {
        $answerCounts[$answer] = 1;
    }
}
$maxCount = 0;
$mostSelectedAnswer = '';
foreach ($answerCounts as $answer => $count) {
    if ($count > $maxCount) {
        $maxCount = $count;
        $mostSelectedAnswer = $answer;
    }
}
?>
<div class="statistics" style="background-color: #8ec98e;
        border-radius: 4px;
        padding: 1rem;
        font-weight: bold;">
    <h5 class="page-title" style="margin:2rem;">Survey Answer
Statistics:</h5>
    <p>Most selected answer: <?= $mostSelectedAnswer ?></p>
    <p>Number of selections: <?= $maxCount ?></p>
</div>
</div>

    <div class="statistics" style="background-color: #f5f5f5;
        border-radius: 4px;
        padding: 1rem;
        margin-top: 2rem;">
<h5 class="page-title" style="margin-bottom: 1rem;">Page access:</h5>
<pre><?= $paginatedLogContent ?></pre>
<?php if ($totalPages > 1) : ?>
    <div class="pagination" style="margin-top: 1rem;">
        <?php for ($i = 1; $i <= $totalPages; $i++) : ?>
            <?php if ($i === $strana) : ?>
                <span class="current page-link"><?= $i ?></span>
            <?php else : ?>

```

```

        <a href="index.php?page=dashboard&strana=<?= $i ?>"
class="page-link"><?= $i ?></a>
        <?php endif; ?>
    <?php endfor; ?>
</div>
<?php endif; ?>
<h5 class="page-title" style="margin: 2rem 0;">Page Access
Statistics:</h5>
<div class="statistics">
    <?php foreach ($pageStatistics as $page => $percentage) : ?>
        <p><?= $page ?>: <?= $percentage ?>%</p>
    <?php endforeach; ?>
</div>
<section class="logged-in-users" style="margin-top: 2rem;">
    <h5 class="page-title">The number of users who logged in during the
day:</h5>
    <p><?php echo "Total number of logged in users: " .
$totalLoggedInUsers; ?></p>
</section>
</div>
</div>
</div>
</section>

```

/views/pages/edit-post.php

```

<?php

    include "models/posts.php";

    adminOnly();

?>

<section class="vh-100 gradient-custom">
    <div class="container">
        <div class="row justify-content-center align-items-center h-100">
            <div class="col-md-8 col-lg-6 col-xl-5">
                <div class="card">
                    <div class="card-header bg-success text-white">
                        <div class="btn-group" role="group" aria-label="Basic
example">
                            <a href="index.php?page=create-post" class="btn
btn-success">Add Post</a>
                            <a href="index.php?page=index-post" class="btn btn-
success">Manage Posts</a>
                        </div>
                    </div>
                    <div class="card-body">
                        <h2 class="card-title text-center mb-4">Edit Post</h2>
                        <?php include "models/formErrors.php"; ?>
                    </div>
                </div>
            </div>
        </div>
    </div>
</section>

```



```

        <form action="index.php?page=edit-post" method="post"
enctype="multipart/form-data">
            <input type="hidden" name="id" value="<?php echo
$id ?>" />

            <div class="form-group">
                <label for="title" class="form-
label">Title</label>

                <input type="text" id="title" class="form-
control" name="title" value="<?php echo $title; ?>" />
            </div>
            <div class="form-group">
                <label for="content" class="form-
label">Content</label>

                <textarea class="form-control" name="content"
id="content" rows="8"><?php echo $content; ?></textarea>
            </div>
            <div class="form-group">
                <label for="image" class="form-
label">Image</label>

                <input type="file" id="image" class="form-
control-file" name="image" />
            </div>
            <div class="form-group">
                <label for="topic" class="form-
label">Topic</label>

                <select name="topic_id" class="form-control"
id="topic">

                    <option value=""></option>
                    <?php foreach($topics as $key =>
$topic): ?>

                        <?php if(!empty($topic_id) && $topic_id
== $topic['id']): ?>

                            <option selected value="<?php echo
$topic['id'] ?>"><?php echo $topic['name'] ?></option>
                        <?php else: ?>

                            <option value="<?php echo
$topic['id'] ?>"><?php echo $topic['name'] ?></option>
                        <?php endif; ?>
                    <?php endforeach; ?>
                </select>
            </div>
            <div class="form-check mb-3">
                <?php if(empty($published) && $published ==
0): ?>

                    <input class="form-check-input"
type="checkbox" name="published" id="publish" value="1">
                    <label class="form-check-label"
for="publish">

                        Publish
                    </label>
                <?php else: ?>

                    <input class="form-check-input"
type="checkbox" name="published" id="publish" value="1" checked>

```

```

                                <label class="form-check-label"
for="publish">
                                Publish
                                </label>
                                <?php endif; ?>
                            </div>
                            <button class="btn btn-success btn-block"
name="update-post" type="submit">Update post</button>
                            </form>
                            <div id="rezultat"></div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>

```

/views/pages/edit-topic.php

```

<?php

    include "models/topics.php";

    adminOnly();

?>
<section class="vh-100 gradient-custom">
    <div class="container">
        <div class="row justify-content-center align-items-center h-100">
            <div class="col-md-8 col-lg-6 col-xl-7">
                <div class="card">
                    <div class="card-header bg-success text-white">
                        <div class="btn-group" role="group" aria-label="Basic
example">
                            <a href="index.php?page=create-topic" class="btn
btn-success">Add Topic</a>
                            <a href="index.php?page=index-topic" class="btn
btn-success">Manage Topics</a>
                        </div>
                    </div>
                    <div class="card-body">
                        <h2 class="card-title text-center mb-4">Edit Topic</h2>
                        <?php include "models/formErrors.php"; ?>
                        <form action="index.php?page=edit-topic"
method="post" novalidate>
                            <div class="mb-3">
                                <input type="hidden" name="id" value="<?php echo
$id; ?>" />
                            </div>
                            <div class="mb-3">
                                <label for="name" class="form-label">Name</label>

```

```

        <input type="text" class="form-control" id="name"
name="name" value="<?php echo $name; ?>">
    </div>
    <div class="mb-3">
        <label for="description" class="form-
label">Description</label>
        <textarea class="form-control" id="description"
name="description" rows="8"><?php echo $description; ?></textarea>
    </div>
    <button class="btn btn-success btn-lg px-5"
name="update-topic" type="submit">Update Topic</button>
    </form>
</div>
</div>
<div id="rezultat"></div>
</div>
</div>
</div>
</section>

```

`/views/pages/edit-user.php`

```

<?php
adminOnly();
include "models/users.php";

?>

<section class="vh-100 gradient-custom">
    <div class="container">
        <div class="row justify-content-center align-items-center h-100">
            <div class="col-md-8 col-lg-6 col-xl-7">
                <div class="card">
                    <div class="card-header bg-success text-white">
                        <div class="btn-group" role="group" aria-label="Basic
example">
                            <a href="index.php?page=create-user" class="btn
btn-success">Add User</a>
                            <a href="index.php?page=index-user" class="btn btn-
success">Manage Users</a>
                        </div>
                    </div>
                    <div class="card-body">
                        <h2 class="card-title text-center mb-4">Edit Users</h2>

                        <?php include "models/formErrors.php"; ?>
                        <form action="index.php?page=edit-user" method="post"
enctype="multipart/form-data">
                            <input type="hidden" name="id" value="<?php echo
$id ?>" />

                            <div class="form-floating mb-4">

```

```

        <input type="text" class="form-control"
name="username" value="<?php echo $username; ?>" />
        <label for="username">Name</label>
    </div>
    <div class="form-floating mb-4">
        <input type="email" class="form-control" name="email"
value="<?php echo $email; ?>" />
        <label for="email">Email</label>
    </div>
    <div class="form-floating mb-4">
        <input type="password" class="form-control"
name="password" value="<?php echo $password; ?>" />
        <label for="password">Password</label>
    </div>
    <div class="form-floating mb-4">
        <input type="password" class="form-control"
name="password_confirmation" value="<?php echo $passConfim; ?>" />
        <label for="password_confirmation">Password
confirmation</label>
    </div>
    <div class="form-check mb-4">
        <?php if (isset($role) && $role == 1) : ?>
            <input type="checkbox" name="role" class="form-
check-input" checked />
            <label class="form-check-label"
for="role">Admin</label>
        <?php else : ?>
            <input type="checkbox" name="role" class="form-
check-input" />
            <label class="form-check-label"
for="role">Admin</label>
        <?php endif; ?>
    </div>
    <div class="d-grid">
        <button class="btn btn-success btn-lg px-5"
id="addTopic" name="update-user" type="submit">Update user</button>
    </div>
</form>
</div>
</div>
</div>
</div>
</div>
</section>

```

/views/pages/gallery.php

```
<?php

    $images = selectAll('galerija');
    ?>

<h2 style="text-align: center; padding: 2rem;">Websites gallery</h2>
<div class="row row-cols-1 row-cols-md-3 g-4" style="margin: 10px 20px;">
    <?php foreach($images as $img): ?>
        <div class="col">
            <div class="card" style="border: none;">
                "
                </div>
            </div>
        <?php endforeach; ?>
    </div>
```

/views/pages/index-post.php

```
<?php
adminOnly();
global $conn;

$result = paginate('posts', 1, PHP_INT_MAX);
$posts = $result['data'];
if (isset($_GET['published']) && isset($_GET['p_id'])) {
    $published = $_GET['published'];
    $postId = $_GET['p_id'];
    $sql = "UPDATE posts SET published = :published WHERE id = :postId";
    $stmt = $conn->prepare($sql);
    $stmt->bindParam(':published', $published, PDO::PARAM_INT);
    $stmt->bindParam(':postId', $postId, PDO::PARAM_INT);
    $stmt->execute();
    echo '<script>>window.location.href = "index.php?page=index-
post";</script>';
    exit;
}
?>

<section class="vh-100 gradient-custom">
    <div class="container py-5">
        <div class="row justify-content-center">
            <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                <div class="card" style="width: 40rem;">
                    <div class="card-body">
                        <div class="d-flex justify-content-between mb-3">
                            <a href="index.php?page=create-post" class="btn btn-success">Add
Post</a>
```

```

        <a href="index.php?page=index-post" class="btn btn-
success">Manage Posts</a>
    </div>
    <h2 class="card-title mb-4">Manage Posts</h2>

    <div class="table-responsive">
        <table class="table table-striped">
            <thead>
                <tr>
                    <th scope="col">SN</th>
                    <th scope="col">Title</th>
                    <th scope="col">Author</th>
                    <th scope="col" colspan="3">Action</th>
                </tr>
            </thead>
            <tbody>
                <?php foreach ($posts as $key => $post) : ?>
                    <tr>
                        <td><?php echo ($key + 1); ?></td>
                        <td><?php echo $post['title']; ?></td>
                        <td><?php echo getUsernameById($post['user_id']); ?></td>
                        <td>
                            <a href="index.php?page=edit-post&id=<?php echo
$post['id']; ?>" class="btn btn-sm btn-primary">Edit</a>
                        </td>
                        <td>
                            <a href="#" onclick="confirmDelete(<?php echo
$post['id']; ?>);" class="btn btn-danger btn-sm">Delete</a>
                        </td>
                        <td>
                            <?php if ($post['published']) : ?>
                                <a href="index.php?page=index-
post&published=0&p_id=<?php echo $post['id'] ?>" class="btn btn-sm btn-
warning">Unpublish</a>
                            </td>
                        <?php else : ?>
                            <td>
                                <a href="index.php?page=index-
post&published=1&p_id=<?php echo $post['id'] ?>" class="btn btn-sm btn-
success">Publish</a>
                            </td>
                        <?php endif; ?>
                    </tr>
                <?php endforeach; ?>
            </tbody>
        </table>
    </div>

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

```

```

</section>
<script>
    function confirmDelete(postId) {
        var result = confirm('Da li ste sigurni da želite obrisati ovu stavku?');
        if (result) {
            window.location.href = 'models/posts.php?delete_id=' + postId + '&confirm_delete=true';
        }
    }
</script>

```

</views/pages/index-topic.php>

```

<?php
adminOnly();
global $conn;

$result = paginate('topics', 1, PHP_INT_MAX);
$topics = $result['data'];
?>
<section class="vh-100 gradient-custom">
    <div class="container py-5">
        <div class="row d-flex justify-content-center align-items-center h-100">
            <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                <div class="card" style="width: 500px;">
                    <div class="card-body">
                        <div class="d-flex justify-content-between">
                            <a href="index.php?page=create-topic" class="btn btn-success mb-3">Add Topic</a>
                            <a href="index.php?page=index-topic" class="btn btn-success mb-3">Manage Topics</a>
                        </div>
                        <h2 class="card-title text-center mb-4">Manage Topics</h2>
                        <table class="table">
                            <thead>
                                <tr>
                                    <th>SN</th>
                                    <th>Name</th>
                                    <th colspan="2">Action</th>
                                </tr>
                            </thead>
                            <tbody>
                                <?php foreach ($topics as $key => $topic): ?>
                                    <tr>
                                        <td><?php echo $key + 1; ?></td>
                                        <td><?php echo $topic['name']; ?></td>
                                        <td>
                                            <a href="index.php?page=edit-topic&id=<?php echo $topic['id']; ?>" class="btn btn-primary btn-sm mr-2">Edit</a>

```

```

        <a href="#" onclick="confirmDelete(<?php echo
$topic['id']; ?>);" class="btn btn-danger btn-sm">Delete</a>
    </td>
</tr>
<?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</section>
<script>
    function confirmDelete(topicId) {
        var result = confirm('Da li ste sigurni da želite obrisati ovu
stavku?');
        if (result) {
            window.location.href = 'models/topics.php?del_id=' + topicId +
'&confirm_delete=true';
        }
    }
</script>

```

</views/pages/index-user.php>

[illegible]


```

        <th>Username</th>
        <th>Email</th>
        <th>Action</th>
    </tr>
</thead>
<tbody>
    <?php foreach($users as $key => $user): ?>
        <tr>
            <td><?php echo $key + 1; ?></td>
            <td><?php echo $user['username']; ?></td>
            <td><?php echo $user['email']; ?></td>
            <td>
                <a href="index.php?page=edit-user&id=<?php echo
$user['id'];?>" class="btn btn-primary btn-sm mr-2">Edit</a>
                <a href="#" onclick="confirmDelete(<?php echo
$user['id']; ?>);" class="btn btn-danger btn-sm">Delete</a>
            </td>
        </tr>
    <?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
</section>
<script>
    function confirmDelete(userId) {
        var result = confirm('Da li ste sigurni da želite obrisati ovu
stavku?');
        if (result) {
            window.location.href = 'models/users.php?delete_id=' + userId +
'&confirm_delete=true';
        }
    }
</script>

```

```
<?php
include "models/messages.php";
$db_host = 'localhost';
$db_name = 'id20728324_webio';
$db_user = 'id20728324_nevena';
$db_password = 'Lozinka123!';
$dsn = "mysql:host=$db_host;dbname=$db_name";
$conn = new PDO($dsn, $db_user, $db_password);
if (!isset($conn) || !$conn) {
    echo "Connection error.";
    exit();
}
if (isset($_SESSION['id'])) {
    $userId = $_SESSION['id'];
    $loggedIn = true;
} else {
    $loggedIn = false;
}
if (isset($_POST['liked'])) {
    $postId = $_POST['postid'];
    try {
        $stmt2 = $conn->prepare("SELECT * FROM posts WHERE id=:postId");
        $stmt2->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt2->execute();
        $row = $stmt2->fetch(PDO::FETCH_ASSOC);
        $n = $row['likes'];
        $stmt3 = $conn->prepare("INSERT INTO likes (user_id, post_id) VALUES
(1, :postId)");
        $stmt3->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt3->execute();
        $nPlusOne = $n + 1;
        $stmt4 = $conn->prepare("UPDATE posts SET likes=:likes WHERE
id=:postId");
        $stmt4->bindParam(':likes', $nPlusOne, PDO::PARAM_INT);
        $stmt4->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt4->execute();
        echo $n + 1;
        exit();
    } catch (Exception $e) {
        echo "Query error.";
        exit();
    }
}
if (isset($_POST['unliked'])) {
    $postId = $_POST['postid'];
    try {
        $stmt2 = $conn->prepare("SELECT * FROM posts WHERE id=:postId");
        $stmt2->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt2->execute();
        $row = $stmt2->fetch(PDO::FETCH_ASSOC);
```

```

        $n = $row['likes'];
        $stmt3 = $conn->prepare("DELETE FROM likes WHERE post_id=:postId AND
user_id=1");
        $stmt3->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt3->execute();
        $nMinusOne = $n - 1;
        $stmt4 = $conn->prepare("UPDATE posts SET likes=:likes WHERE
id=:postId");
        $stmt4->bindParam(':likes', $nMinusOne, PDO::PARAM_INT);
        $stmt4->bindParam(':postId', $postId, PDO::PARAM_INT);
        $stmt4->execute();
        echo $n - 1;
        exit();
    } catch (Exception $e) {
        echo "Query error.";
        exit();
    }
}
try {
    $stmt2 = $conn->query("SELECT * FROM posts");
    $posts = $stmt2->fetchAll(PDO::FETCH_ASSOC);
    $postsTitle = 'Recent Posts';
    if (isset($_GET['t_id'])) {
        $posts = getPostsbyTopicId($_GET['t_id']);
        $postsTitle = "You searched for posts under '" . $_GET['name'] . "'";
    } else if (isset($_POST['search'])) {
        $postsTitle = "You searched for '" . $_POST['search'] . "'";
        $posts = searchPosts($_POST['search']);
    } else {
        $posts = getPublishedPosts();
    }
} catch (Exception $e) {
    echo "Query error.";
    exit();
}
if (isset($_POST["post_comment"])) {
    $name = $_POST["name"];
    $comment = $_POST["comment"];
    $post_id = $_POST['post_id'];
    $stmt = $conn->prepare("SELECT * FROM comments WHERE name = :name AND
comment = :comment AND post_id = :post_id");
    $stmt->bindParam(':name', $name, PDO::PARAM_STR);
    $stmt->bindParam(':comment', $comment, PDO::PARAM_STR);
    $stmt->bindParam(':post_id', $post_id, PDO::PARAM_INT);
    $stmt->execute();
    $existingComment = $stmt->fetch(PDO::FETCH_ASSOC);
    if ($existingComment) {
        echo "<p>Comment already exists.</p>";
    } else {
        try {
            $user_id = $_SESSION['id'];
            $stmt = $conn->prepare("INSERT INTO comments (name, comment, post_id,
user_id) VALUES (:name, :comment, :post_id, :user_id)");

```

[illegible]

```

                <h5 class="card-title"><a
href="index.php?page=more&id=<?php echo $post['id'] ?>"><?php echo
$post['title']; ?></a></h5>
                <p class="card-text"><i class="fa fa-user me-
2"></i><?php echo $post['username']; ?></p>
                <p class="card-text"><i class="fa fa-calendar
me-2"></i><?php echo date('F j. Y.', strtotime($post['created_at'])); ?></p>
            </div>
        </div>
    </div>
    <?php if ($i % 3 == 2 || $i == count($posts) - 1) : ?>
    </div>
    </div>
    <?php endif; ?>
    <?php $i++;
endforeach; ?>
</div>
<button class="prev" type="button" data-bs-target="#myCarousel"
data-bs-slide="prev">
    <span class="prev-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Previous</span>
</button>
<button class="next" type="button" data-bs-target="#myCarousel"
data-bs-slide="next">
    <span class="next-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Next</span>
</button>
</div>
</div>
</section>
<section>
    <div class="container" style="margin: 5rem auto;">
        <div class="row">
            <div class="col-md-9">
                <div class="d-flex justify-content-center">
                    <h2 style="padding-bottom: 5rem"><?php echo
$post['title']; ?></h2>
                </div>
                <?php if (empty($posts)) : ?>
                    <p>No posts yet.</p>
                <?php else : ?>
                    <div class="row">
                        <?php foreach ($posts as $post) : ?>
                            <div class="col-md-4" style="margin-bottom: 2rem;">
                                <div class="card">
                                    
                                    <div class="card-body">
                                        <h5 class="card-title">
                                            <a
href="index.php?page=more.php&id=<?php echo $post['id'] ?>"><?php echo
$post['title']; ?></a>
                                        </h5>

```

```

        <p class="card-text">
            <?php echo substr($post['content'],
0, 150) . '...'; ?>

        </p>
        <div class="card-footer">
            <i class="fa fa-user" style="margin-
right: 4px;"> by: <?php echo $post['username']; ?> </i>
            <i class="fa fa-calendar"> <?php
echo date('F j. Y.', strtotime($post['created_at'])); ?> </i>
            <?php if ($loggedIn) : ?>
                <div class="post">

                    <div style="padding: 2px;
margin-top: 5px;">

                        <?php
                            $stmtLikes = $conn-
>prepare("SELECT * FROM likes WHERE user_id=:userId AND post_id=:postId");
                            $stmtLikes-
>bindParam(':userId', $userId, PDO::PARAM_INT);
                            $stmtLikes-
>bindParam(':postId', $post['id'], PDO::PARAM_INT);
                            $stmtLikes->execute();

                            if ($stmtLikes->rowCount()
== 1) :
                                ?>
                                <span class="unlike fa
fa-thumbs-down" data-id="<?php echo $post['id']; ?>"></span>
                                <span class="like hide
fa fa-thumbs-o-up" data-id="<?php echo $post['id']; ?>"></span>
                                <?php else : ?>
                                <span class="like fa fa-
thumbs-o-up" data-id="<?php echo $post['id']; ?>"></span>
                                <span class="unlike fa
fa-thumbs-down" data-id="<?php echo $post['id']; ?>"></span>
                                <?php endif ?>

                                <span class="likes_count"
style="color: #d27676;
font-weight: bold;"> <?php
echo $post['likes']; ?> likes</span>

                            </div>
                        </div>
                    </form>
                    <div class="form-control" style="border: 1px solid #ccc; padding: 5px;">
                        <input type="hidden"
name="post_id" value="<?php echo $post['id']; ?>" required>
                        <div class="mb-3">
                            <label for="" class="form-
label">Your name</label>
                            <input type="text"
name="name" class="form-control" required>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

        <div class="mb-3">
            <label for="" class="form-
label">Comment</label>

            <textarea name="comment"
class="form-control" required></textarea>

        </div>
        <button type="submit"
class="btn btn-primary" name="post_comment" style="margin-bottom: 1rem;
font-size: 13px;">
            Add Comment
        </button>
    </form>
    <?php
    $stmt = $conn->prepare("SELECT *
FROM comments WHERE post_id = :post_id");
    $stmt->bindParam(':post_id',
$post['id'], PDO::PARAM_INT);

    $stmt->execute();
    if ($stmt->errorCode() !==
"00000") {

        $errorMsg = $stmt-
>errorInfo();

        echo "Error executing query:
" . $errorMsg[2];

        exit();
    }
    $comments = $stmt-
>fetchAll(PDO::FETCH_OBJ);

    <div class="list-unstyled">';

    foreach ($comments as
$comment) {

        <div class="mb-4"
style="border: 2px solid #dea7a73d;
border-radius: 4%;
padding: 8px;">';

        <div class="fa fa-
user-o" aria-hidden="true"> ' . $comment->name . '</i>';

        <div class="fa
fa-commenting-o" aria-hidden="true"> ' . $comment->comment . '</i>';

        <div class="text-
muted">' . date("F d, Y h:i a", strtotime($comment->created_at)) . '</p>';

        </div>';

    }
    </div>';
    } else {
        echo 'No comments
found.';
    }
    ?>

```

```

        <?php else : ?>
            <div class="login-warning"
style=" color: #b84b4b;

    font-weight: bold;

    font-size: 15px;

    margin: 10px;">

                Please log in to like and
comment.

            </div>
        <?php endif; ?>
    </div>
</div>
</div>
</div>
<?php endforeach; ?>
</div>
<?php endif; ?>
</div>
<div class="col-md-3">
    <div class="search" style="position: relative; top: 20px;">
        <form action="index.php" method="post">
            <div class="input-group">
                <input type="text" name="search" class="form-
control" placeholder="Search title">
                <div class="input-group-append">
                    <button class="btn btn-primary" type="submit">
                        <i class="fa fa-search"></i>
                    </button>
                </div>
            </div>
        </form>
        <div id="topics">
            <h3>Topics</h3>
            <ul class="list-group">
                <?php foreach ($topics as $topic) : ?>
                    <li class="list-group-item">
                        <a class="text-decoration-none"
href="index.php?page=index&t_id=<?php echo $topic['id']; ?>&name=<?php echo
$topic['name']; ?>">

                            <?php echo $topic['name']; ?>
                        </a>
                    </li>
                <?php endforeach; ?>
            </ul>
        </div>
    </div>
</div>
</div>
</div>
</div>
</section>

```


/views/pages/login.php

```
<?php
include "models/users.php";
error_reporting(0);
$accessLog = "data/access.log";
$logMessage = date("Y-m-d H:i:s") . " - Page accessed: " .
$_SERVER['REQUEST_URI'] . "\n";
file_put_contents($accessLog, $logMessage, FILE_APPEND);

$logFile = "data/login.log";
if (!file_exists($logFile)) {
    fopen($logFile, 'w');
}

$userId = $_SESSION['id'];
logLogin($userId, $logFile);
if (isset($_POST['login-btn'])) {
    $errors = validateLogin($_POST);
    if (count($errors) === 0) {
        $user = selectOne($table, ['email' => $_POST['email']]);
        if ($user && password_verify($_POST['password'], $user['password'])) {
            loginUser($user);
            logLogin($user['id'], $logFile);
            $logMessage = date("Y-m-d H:i:s") . " - User logged in: " .
            $user['id'] . "\n";
            file_put_contents($logFile, $logMessage, FILE_APPEND);
        } else {
            array_push($errors, 'Wrong credentials. ');
            if ($user) {
                $failedLoginAttempts = getFailedLoginAttempts($user['id']);
                $maxFailedAttempts = 3;
                if ($failedLoginAttempts >= $maxFailedAttempts) {
                    try {
                        lockAccount($user['id']);
                        sendLockoutEmail($user['email']);
                        $errors = 'Please try again after 30 seconds.';
                        $lockedUntil = date('Y-m-d H:i:s', time() + 30);
                        updateLockResetTime($user['id'], $lockedUntil);
                        disableLoginButton();
                        startTimer(30);
                    } catch (Exception $e) {
                        $errors = $e->getMessage();
                    }
                } else {
                    incrementFailedLoginAttempts($user['id']);
                }
            }
        }
    }
}

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

```

}
?>
<section class="vh-100 gradient-custom" style="height: 100% !important;">
  <div class="container py-5 h-100">
    <div class="row d-flex justify-content-center align-items-center h-100">
      <div class="col-12 col-md-8 col-lg-6 col-xl-5">
        <div class="card bg-dark text-white" style="border-radius: 1rem;">
          <div class="card-body p-5 text-center">
            <div class="mb-md-5 mt-md-4 pb-5">
              <h2 class="fw-bold mb-2 text-uppercase">Login</h2>
              <p class="text-white-50 mb-5">Please enter your username and password!</p>
              <?php if(count($errors) > 0): ?>
                <div class="alert alert-danger">
                  <?php foreach($errors as $error): ?>
                    <p><?php echo $error; ?></p>
                  <?php endforeach; ?>
                </div>
              <?php endif; ?>
              <form role="form" action="index.php?page=login" method="post" onsubmit="return validateLogin()" name="myForm" novalidate>
                <input type="hidden" name="id"/>
                <span id="check-username"></span>
                <div class="form-outline mb-4">
                  <input type="email" id="email" class="form-control form-control-lg" value="<?php echo $email; ?>" name="email" />
                  <label class="form-label" for="form3Example1cg" name="email">Your email</label>
                </div>
                <div class="form-outline form-white mb-4">
                  <input type="password" id="password" class="form-control form-control-lg" value="<?php echo $password; ?>" name="password" />
                  <label class="form-label" for="typePasswordX" name="password">Password</label>
                </div>
                <button id="login-btn" class="btn btn-outline-light btn-lg px-5" name="login-btn" type="submit">Login</button>
              </div>
              <div>
                <p class="mb-0">Don't have an account? <a href="index.php?page=register" class="text-white-50 fw-bold">Sign Up</a> </p>
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>

```

```

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
$(document).ready(function() {
    var isLocked = <?php echo ($user['locked'] == 1 ? 'true' : 'false'); ?>;
    if (isLocked) {
        disableLoginButton();
        startTimer(30);
    }
});
function disableLoginButton() {
    $('#login-btn').prop('disabled', true);
}
function enableLoginButton() {
    $('#login-btn').prop('disabled', false);
}
function startTimer(seconds) {
    var timer = setInterval(function() {
        if (seconds <= 0) {
            clearInterval(timer);
            resetFailedLoginAttempts();
            enableLoginButton();
        } else {
            seconds--;
        }
    }, 1000);
}
function resetFailedLoginAttempts() {
    <?php
    if ($user['failed_login_attempts'] >= $maxFailedAttempts) {
        $query = "UPDATE users SET failed_login_attempts = 0 WHERE id
= :userId";
        $statement = $conn->prepare($query);
        $statement->bindValue(':userId', $user['id']);
        $statement->execute();
    }
    ?>
}
</script>

```

</views/pages/logout.php>

```
<?php

unset($_SESSION['id']);
unset($_SESSION['username']);
unset($_SESSION['role']);
unset($_SESSION['message']);
unset($_SESSION['type']);
session_destroy();
echo '<script>>window.location.href = "index.php?page=index";</script>';
exit();
?>
```

[/views/pages/more.php](#)[illegible]


```

        <div class="mb-md-5 mt-md-4 pb-5">

            <h2 class="fw-bold mb-2 text-uppercase">Register</h2>
            <p class="text-white-50 mb-5">Please enter your username
and password!</p>

            <div class="error-message" style="<?php echo
(count($errors) > 0) ? 'display:block' : 'display:none'; ?>">
                <?php echo implode('<br>', $errors); ?>
            </div>

            <form role="form" action="index.php?page=register"
method="post" onsubmit="return validateRegistration()" name="regForm"
novalidate>

                <div class="form-outline mb-4">
                    <input type="text" id="reg-username" class="form-
control form-control-lg" value="<?php echo $username;?>" name="username" />
                    <label class="form-label" >Your Name</label>
                </div>
                <div class="form-outline mb-4">
                    <input type="email" id="reg-email" class="form-
control form-control-lg"
                        value="<?php echo $email;?>" name="email"
onblur="checkEmailAvailability()" />
                    <label class="form-label">Your Email</label>
                    <span id="email-availability-status"
style="display:none;"></span>
                </div>
                <div class="form-outline mb-4">
                    <input type="password" id="reg-password"
class="form-control form-control-lg" value="<?php echo $password;?>"
name="password" />
                    <label class="form-label">Password</label>
                </div>
                <div class="form-outline mb-4">
                    <input type="password" id="reg-password-
confirmation" class="form-control form-control-lg" value="<?php echo
$passConfim;?>" name="password_confirmation" />
                    <label class="form-label">Repeat your
password</label>
                </div>

                <div class="d-flex justify-content-center">
                    <button type="submit" class="btn btn-success btn-
block btn-lg gradient-custom-4 text-body" id="submit-reg" name="register-
btn">Register</button>
                </div>
            </form>
        </div>
    </div>
</div>
</div>

```

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

/views/pages/survey.php

```
<?php

usersOnly();
if (!isset($_GET['u_id'])) {
    header("Location: index.php?page=index");
    exit;
}
$user_id = $_GET['u_id'];
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $answer_id = $_POST["answer"];
    $data = array('user_id' => $user_id, 'answer_id' => $answer_id);
    $id = insertOperation("user_answer", $data);
    if ($id) {
        $_SESSION['message'] = 'You have successfully sent a reply!';
        $_SESSION['type'] = 'alert alert-success';
        header("Location: index.php?page=index");
        exit();
    } else {
        $_SESSION['message'] = 'An error occurred while sending the response.
Please try again later.';
        $_SESSION['type'] = 'alert alert-danger';
    }
}
$answers = selectAll("survey");
?>
<div class="container my-5">
    <div class="card">
        <div class="card-body">
            <div class="text-center">
                <i class="far fa-file-alt fa-4x mb-3 text-primary"></i>
                <p class="fs-4">
                    <strong>Your opinion matters</strong>
                </p>
                <p class="fs-5">
                    Have some ideas how to improve our product?
                    <strong>Give us your feedback.</strong>
                </p>
            </div>
            <hr />
            <form method="POST" action="index.php?page=survey&u_id=<?php echo
$user_id; ?>">
                <p class="text-center fs-5"><strong>Your rating:</strong></p>
                <?php foreach ($answers as $answer) : ?>
                    <div class="form-check mb-2">
                        <input class="form-check-input" type="radio" name="answer"
id="answer<?= $answer['id'] ?>" value="<?= $answer['id'] ?>" />
```

```

                <label class="form-check-label" for="answer<?=
$answer['id'] ?>">
                <?= $answer['answer'] ?>
            </label>
        </div>
    <?php endforeach; ?>
    <div class="card-footer text-end">
        <button type="submit" class="btn btn-primary">Submit</button>
    </div>
</form>
</div>
</div>
</div>

```

/root/sitemap.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=index</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>daily</changefreq>
        <priority>0.8</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=contact</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>daily</changefreq>
        <priority>1</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=more</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>daily</changefreq>
        <priority>0.9</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=gallery</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.7</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=author</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=login</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>yearly</changefreq>
    </url>

```



```

        <priority>0.5</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=register</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
    <url>
        <loc>https://webio-
php.000webhostapp.com/index.php?page=dashboard</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
    <url>
        <loc>https://webio-php.000webhostapp.com/index.php?page=survey</loc>
        <lastmod>2023-05-10</lastmod>
        <changefreq>yearly</changefreq>
        <priority>0.5</priority>
    </url>
</urlset>

```

/assets/css/style.css

```

body {
    background-color: rgb(193, 229, 193);
}

.error-message {
    border-radius: 4px;
    color:red;
    font-weight: bold;
    background-color: rgb(246, 208, 208);
    width: 90%;
    padding: 5px;
    margin: 15px;
    text-align: center;
}

.error {
    color:red;
}

#topics{
    margin: 3.5rem;
}

.card-img-top-fixed {
    width: 100%;
    height: 300px;
    object-fit: cover;
}

.prev, .next {

```

```

width: 7%;
background-color: rgb(95, 188, 33);
border-radius: 4px;
border: none;
position: absolute;
top: 50%;
transform: translateY(-50%);
font-weight: bold;
}
.prev {
  left: -8rem;
}
.next {
  right: -8rem;
}
.images {
  width: 100%;
  height: 400px;
  object-fit: cover;
  padding: 2rem;
  background-color: #c1e5c1;
}

```

/assets/js/main.js

```

$(document).ready(function() {
  $('.msg').animate({
    opacity: 1
  }, 500);

  setTimeout(function() {
    $('.msg').animate({
      opacity: 0
    }, 500, function() {
      $(this).remove();
    });
  }, 2000);

  $('.prev').click(function() {
    $('#myCarousel').carousel('prev');
  });

  $('.next').click(function() {
    $('#myCarousel').carousel('next');
  });
});

function validateField(input, fieldName, regex = null) {
  const value = input.value.trim();
  if (!value) {
    showError(input, `The ${fieldName} is required`);
    return false;
  }
}

```

```

    if (regex && !regex.test(value)) {
        showError(input, `The ${fieldName} is invalid`);
        return false;
    }
    return true;
}

function validateLogin() {
    const emailInput = document.getElementById("email");
    const passwordInput = document.getElementById("password");
    const emailRegx = /^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;
    document.querySelectorAll(".error").forEach((element) => element.remove());
    const isEmailValid = validateField(emailInput, "email", emailRegx);
    const isPasswordValid = validateField(passwordInput, "password");
    return isEmailValid && isPasswordValid;
}

function validateRegistration() {
    const usernameInput = document.getElementById("reg-username");
    const emailInput = document.getElementById("reg-email");
    const passwordInput = document.getElementById("reg-password");
    const passwordConfirmInput = document.getElementById("reg-password-confirmation");
    const usernameRegx = /^[a-zA-Z][a-zA-Z0-9]{3,15}$/;
    const emailRegx = /^\S+@\S+\.\S+$/;
    document.querySelectorAll(".error").forEach((element) => element.remove());
    const isUsernameValid = validateField(usernameInput, "username", usernameRegx);
    const isEmailValid = validateField(emailInput, "email", emailRegx);
    const isPasswordValid = validateField(passwordInput, "password");
    const isPasswordConfirmValid = validateField(passwordConfirmInput, "password confirmation");
    if (isPasswordConfirmValid && passwordInput.value.trim() !== passwordConfirmInput.value.trim()) {
        showError(passwordConfirmInput, "The password and password confirmation does not match");
        return false;
    }
    return isUsernameValid && isEmailValid && isPasswordValid && isPasswordConfirmValid;
}

function showError(inputElement, errorMessage) {
    const errorDiv = document.createElement("div");
    inputElement.classList.add("error-border");
    errorDiv.innerText = errorMessage;
    errorDiv.classList.add("error");
    inputElement.parentNode.insertBefore(errorDiv, inputElement.nextElementSibling);
    inputElement.setAttribute("data-error", errorMessage);

    inputElement.addEventListener('input', function() {
        const errorDiv = inputElement.parentNode.querySelector('.error');
        if (errorDiv) errorDiv.remove();
        inputElement.classList.remove('error-border');
    });
}

```

```

        inputElement.removeAttribute("data-error");
    });
}
function checkEmailAvailability() {
    var email = $("#reg-email").val();
    $.ajax({
        type: "POST",
        url: "checkEmailAvailability.php",
        data: {email: email},
        success: function(response){
            if(response=="Email available.") {
                $('#email-availability-status').html('<span class="text-success">'+response+'</span>').show();
            } else {
                $('#email-availability-status').html('<span class="text-danger">'+response+'</span>').show();
            }
        },
        error: function(jqXHR, textStatus, errorThrown) {
            console.log("Error: " + textStatus);
        }
    });
}
$(document).ready(function(){
    $('.like').on('click', function(){
        var postid = $(this).data('id');
        $post = $(this);
        $.ajax({
            url: 'index.php',
            type: 'post',
            data: {
                'liked': 1,
                'postid': postid
            },
            success: function(response){
                $post.parent().find('span.likes_count').text(response + " likes");
                $post.addClass('hide');
                $post.siblings().removeClass('hide');
            }
        });
    });
    $('.unlike').on('click', function(){
        var postid = $(this).data('id');
        $post = $(this);
        $.ajax({
            url: 'index.php',
            type: 'post',
            data: {
                'unliked': 1,
                'postid': postid
            },
            success: function(response){
                $post.parent().find('span.likes_count').text(response + " likes");
            }
        });
    });
});

```

```

        $post.addClass('hide');
        $post.siblings().removeClass('hide');
    }
    });
});
});
function showReplyForm(self) {
    var commentId = self.getAttribute("data-id");
    var form = document.getElementById("form-" + commentId);
    if (form.style.display === "block") {
        form.style.display = "none";
    } else {
        form.style.display = "block";
    }
}
function showReplyForReplyForm(element) {
    var commentId = element.getAttribute("data-id");
    var name = element.getAttribute("data-name");
    var form = document.getElementById("form-" + commentId);
    if (form.style.display === "none") {
        form.style.display = "block";
    } else {
        form.style.display = "none";
    }
    var textarea = document.querySelector("#form-" + commentId + "
textarea[name=comment]");
    textarea.value = "@" + name;
    form.scrollIntoView();
}

```

[/index.php](#)

```

<?php
session_start();

include 'app/database/connect.php';
include_once "models/db.php";
include "models/usersType.php";
$logFile = "data/login.log";
if (!file_exists($logFile)) {
    fopen($logFile, 'w');
}
function logAccess($page) {
    $accessLog = "data/access.log";
    $logMessage = date("Y-m-d H:i:s") . " - Page accessed: " . $page . "\n";
    file_put_contents($accessLog, $logMessage, FILE_APPEND);
}
if (isset($_GET['id'])) {
    $userId = $_GET['id'];
}

if (isset($_GET['page'])) {
    $currentPage = $_GET['page'];
}

```

```
logAccess($currentPage);
switch ($currentPage) {
    case 'dashboard':
        include "views/fixed/header.php";
        include "views/fixed/adminSidebar.php";
        include "views/pages/dashboard.php";
        break;
    case 'login':
        include "views/fixed/header.php";
        include "views/pages/login.php";
        break;
    case 'logout':
        include "views/fixed/header.php";
        include "views/pages/logout.php";
        break;
    case 'register':
        include "views/fixed/header.php";
        include "views/pages/register.php";
        break;
    case 'upload-slike':
        include "views/fixed/header.php";
        include "views/pages/upload-slike.php";
        break;
    case 'contact':
        include "views/fixed/header.php";
        include "views/pages/contact.php";
        break;
    case 'index-post':
        include "views/fixed/header.php";
        include "views/pages/index-post.php";
        break;
    case 'edit-post':
        include "views/fixed/header.php";
        include "views/pages/edit-post.php";
        break;
    case 'create-post':
        include "views/fixed/header.php";
        include "views/pages/create-post.php";
        break;
    case 'index-topic':
        include "views/fixed/header.php";
        include "views/pages/index-topic.php";
        break;
    case 'edit-topic':
        include "views/fixed/header.php";
        include "views/pages/edit-topic.php";
        break;
    case 'create-topic':
        include "views/fixed/header.php";
        include "views/pages/create-topic.php";
        break;
    case 'create-user':
```

```
        include "views/fixed/header.php";
        include "views/pages/create-user.php";
        break;
    case 'edit-user':
        include "views/fixed/header.php";
        include "views/pages/edit-user.php";
        break;
    case 'index-user':
        include "views/fixed/header.php";
        include "views/pages/index-user.php";
        break;
    case 'more':
        include "views/fixed/header.php";
        include "views/pages/more.php";
        break;
    case 'gallery':
        include "views/fixed/header.php";
        include "views/pages/gallery.php";
        break;
    case 'author':
        include "views/fixed/header.php";
        include "views/pages/author.php";
        break;
    case 'survey':
        include "views/fixed/header.php";
        include "views/pages/survey.php";
        break;
    default:
        include "views/fixed/header.php";
        include "views/pages/index.php";
    }
} else {
    logAccess('index');
    include "views/pages/index.php";
}
include "views/fixed/footer.php";
?>
```