Visoka škola strukovnih studija za informacione i komunikacione tehnologije



# Dokumentacija

<http://25h-kitchen.epizy.com/>

Nikola Nedeljković 2/17

# Sadržaj

[Dokumentacija 1](#_Toc11419161)

[Sadržaj 2](#_Toc11419162)

[1. UVOD 6](#_Toc11419163)

[1.1 Korišćeni programski / markap jezici 6](#_Toc11419164)

[1.2 Opis funkcionalnosti 6](#_Toc11419165)

[1.3 Izgled stranica 11](#_Toc11419166)

[2. ORGANIZACIJA 12](#_Toc11419167)

[2.1 Organizacija sajta 12](#_Toc11419168)

[2.2 Organizacija baze 13](#_Toc11419172)

[2.3 Sitemap.xml 14](#_Toc11419177)

[2.4 Slike stranica sa funkcionalnostima 15](#_Toc11419183)

[3. KODOVI 21](#_Toc11419184)

[3.1 index.php 21](#_Toc11419185)

[3.2. views/login\_template.php 22](#_Toc11419190)

[3.3. views/header\_template.php 23](#_Toc11419191)

[3.4 views/footer\_template.php 23](#_Toc11419198)

[3.5. views/food\_post\_template.php 24](#_Toc11419200)

[3.6 views/contact\_template.php 25](#_Toc11419209)

[3.7 views/admin\_template.php 26](#_Toc11419219)

[3.8. views/add\_post\_template.php 27](#_Toc11419230)

[3.9 views/404.php 27](#_Toc11419242)

[3.10 models/add\_post.php 28](#_Toc11419255)

[3.11. models/comment.php 31](#_Toc11419269)

[3.12 models/comments.php 32](#_Toc11419284)

[3.13. models/contact.php 32](#_Toc11419300)

[3.14. models/export/excel.php 33](#_Toc11419317)

[3.15. models/forgot\_password.php 34](#_Toc11419335)

[3.16. models/logged\_users.php 34](#_Toc11419354)

[3.17. models/login.php 34](#_Toc11419357)

[3.18. models/logout.php 35](#_Toc11419358)

[3.19. models/navigation.php 36](#_Toc11419380)

[3.20. models/page\_access\_info.php 36](#_Toc11419384)

[3.21. models/posts.php 36](#_Toc11419408)

[3.22. models/random\_quote.php 37](#_Toc11419433)

[3.23. models/register.php 37](#_Toc11419459)

[3.24. models/save\_word.php 39](#_Toc11419486)

[3.25. models/single\_post.php 39](#_Toc11419514)

[3.26. models/subscribe.php 40](#_Toc11419543)

[3.27. models/tags.php 41](#_Toc11419573)

[3.28. models/top\_posts.php 41](#_Toc11419604)

[3.29. views/types.php 41](#_Toc11419636)

[3.30. views/write\_error.php 41](#_Toc11419669)

[3.31. models/composer/contact\_mail.php 42](#_Toc11419703)

[3.32. models/admin/comments.php 43](#_Toc11419738)

[3.33. models/admin/delete\_comment.php 43](#_Toc11419774)

[3.34. models/admin/delete\_newsletter.php 44](#_Toc11419811)

[3.35. models/admin/delete\_picture.php 44](#_Toc11419849)

[3.36. models/admin/delete\_post\_tag.php 45](#_Toc11419888)

[3.37. models/admin/delete\_post.php 45](#_Toc11419928)

[3.38. models/admin/delete\_tag.php 47](#_Toc11419969)

[3.39. models/admin/delete\_user.php 48](#_Toc11420011)

[3.40. models/admin/edit\_post.php 48](#_Toc11420054)

[3.41. models/admin/edit\_tag.php 49](#_Toc11420098)

[3.42. models/admin/edit\_user.php 50](#_Toc11420143)

[3.43. models/admin/insert\_picture.php 51](#_Toc11420189)

[3.44. models/admin/insert\_post\_tag.php 53](#_Toc11420236)

[3.45. models/admin/insert\_tag.php 54](#_Toc11420284)

[3.46. models/admin/insert\_user.php 55](#_Toc11420333)

[3.47. models/admin/newsletter.php 56](#_Toc11420383)

[3.48. models/admin/pictures.php 56](#_Toc11420434)

[3.49. models/admin/post\_tag.php 57](#_Toc11420486)

[3.50. models/admin/posts.php 57](#_Toc11420488)

[3.51. models/admin/tags.php 57](#_Toc11420490)

[3.52. models/admin/users.php 57](#_Toc11420495)

[3.53. assets/lang/eng.json 57](#_Toc11420498)

[3.54. assets/lang/srb.json 62](#_Toc11420501)

[3.55. assets/js/add\_post\_template.js 67](#_Toc11420503)

[3.56. assets/js/admin\_template.js 69](#_Toc11420505)

[3.57. assets/js/contact\_template.js 97](#_Toc11420510)

[3.58. assets/js/food\_post\_template.js 100](#_Toc11420513)

[3.59. models/admin/post\_tag.php 109](#_Toc11420516)

[3.60. assets/js/footer\_template.js 109](#_Toc11420518)

[3.61. assets/js/header\_template.js 110](#_Toc11420520)

[3.62. assets/js/login\_template.js 111](#_Toc11420525)

[3.63. assets/js/main\_template.js 116](#_Toc11420526)

[3.64. assets/js/main.js 117](#_Toc11420530)

[3.65. assets/css/style.css 124](#_Toc11420534)

[3.66. assets/css/food\_post\_template.css 125](#_Toc11420536)

[3.67. assets/css/footer\_template.css 130](#_Toc11420538)

[3.68. assets/css/header\_template.css 132](#_Toc11420543)

[3.69. assets/css/login\_template.css 134](#_Toc11420544)

[3.70. assets/css/main\_template.css 136](#_Toc11420545)

[3.71. assets/css/add\_post\_template.css 136](#_Toc11420549)

[3.72. assets/css/admin\_template.css 137](#_Toc11420551)

[3.73. assets/css/contact\_template.css 140](#_Toc11420553)

[3.74. assets/css/page\_404.css 141](#_Toc11420555)

[3.75. data/errors.txt (PRIMER) 141](#_Toc11420556)

[3.76. data/log.txt (PRIMER) 141](#_Toc11420566)

[3.77. data/rss-eng.xml 144](#_Toc11420570)

[3.78. data/rss-srb.xml 146](#_Toc11420572)

[3.79. config/.env 148](#_Toc11420574)

[3.80. config/config.php 148](#_Toc11420579)

[6.47. models/admin/newsletter.php 148](#_Toc11420629)

[3.81. config/connection.php 149](#_Toc11420632)

[3.82. robots.txt 149](#_Toc11420635)

[3.83. .htaccess 149](#_Toc11420637)

[4. HIJERARHIJSKI PREGLED 150](#_Toc11420638)

# UVOD

## Korišćeni programski / markap jezici

* Za izradu ovog sajta korišćene su sledeće tehnologije:
* Mark-up jezici: HTML 5, XML
* CSS3
* Skripting-programski jezici: JavaScript, PHP
* JQuery – JavaScript framework
* JSON, AJAX
* PHP biblioteke: PHPmailer, PHPWord i PHPStylesheet
* Sajt je napisan u Atom-u

## Opis funkcionalnosti

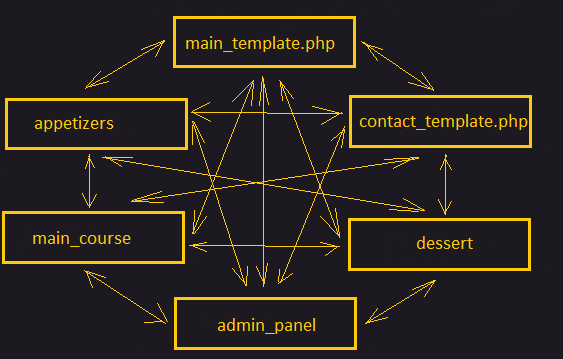
* **index.php**
* Sadržaj svih stranica i dinamičko otvaranje istih preko GET metoda.
* **main\_template.php**
* Landing page sa gif animacijom
* **food\_post\_template.php**
* Sadržaj svih stranica i dinamičko otvaranje istih preko GET metoda.
* Sadržaj menija sa strane
* **login\_template.php**
* Sadržaj svih stranica i dinamičko otvaranje istih preko GET metoda.
* Login, Logout, Register, Forgot Password
* **header\_template.php**
* Navigacija i odabir jezika
* **footer\_template.php**
* Navigacija, najnovije objave i tagovi
* **contact\_template.php**
* Sadržaj kontakt stranice
* O autoru
* Download word o autoru
* **admin\_template.php**
* Forma za admina
* Download dokumentacije
* Export to excel
* Broj logovanih korisnika
* Broj poseta posebnim stranicama
* **add\_post\_template.php**
* Forma za dodavanje nove objave
* **404.php**
* Stranica za grešku
* **add\_post.php**
* Dodavnje nove objave
* Korišćenje transakcija
* Rad sa slikama
* Upis u srpski i engleski RSS
* Provera podataka u PHP-u i vraćanje statusnih kodova
* **comment.php**
* Dodavanje komentara na neku objavu
* Provera podataka u PHP-u i vraćanje statusnih kodova
* **comments.php**
* Dohvatanje svih komentara za određenu objavu
* **contact.php**
* Slanje email-a administratoru korišćenjem PHP mailer-a
* **export\_excel.php**
* Prebacivanje podataka u Excel i automatsko čuvanje na Desktop korišćenjem PHPSpreadsheets
* **forgot\_password.php**
* Slanje mail-a administratorima za zaboravljenu šifru
* **logged\_users.php**
* Dohvatanje svih logovanih korisnika
* **login.php**
* Logovanje korisnika
* Slanje mail-a u slučaju neuspešnog logovanja
* **logout.php**
* Odjava korisnika
* **navigation.php**
* Dohvatanje navigacije
* **voting.php**
* Provera da li je logovani korisnik već glasao
* **page\_access\_info.php**
* Dodavanje informacija o logovanju korisnika
* **add\_navigation.php**
* Dohvatanje svih objava
* **random\_quote.php**
* Dohvatanje nasumičnog statusa
* **add\_user.php**
* Dodavanje novog korisnika
* **register.php**
* Registracija korisnika
* Šifrovanje podataka
* Provera podataka u PHP-u i vraćanje statusnih kodova
* **save\_word.php**
* Sačuvljavanje podataka o autoru u Word-u na korisnički desktop putem COM klase
* **single\_post.php**
* Dohvatanje pojedinačne objave
* **subscribe.php**
* Prijava korisnika na bilten
* Provera podataka u PHP-u i vraćanje statusnih kodova
* **tags.php**
* Dohvatanje svih objava korisnika
* **top\_posts.php**
* Dohvatanje najnovijih objava
* **types.php**
* Dohvatanje svih tipova objava
* **write\_error.php**
* Upisivanje grešaka u poseban fajl
* **contact\_mail.php**
* Funkcije za slanje mejlova korisnicima i administratoru
* **comments.php**
* Dohvatanje svih komentara
* **delete\_comment.php**
* Brisanje komentara
* **delete\_newsletter.php**
* Brisanje pretplate
* **delete\_picture.php**
* Brisanje slike i veze sa objavom
* **delete\_post\_tag.php**
* Brisanje veze između objave i taga
* **delete\_post.php**
* Brisanje objave i uklanjanje slika i veza za objavu
* Uklanjanje iz RSS-a sve podatke o objavi
* **delete\_tag.php**
* Brisanje taga
* **delete\_user.php**
* Brisanje korisnika
* **edit\_post.php**
* Izmena objave i izmena RSS-a vezanog za tu objavu
* **edit\_tag.php**
* Izmena taga
* **edit\_user.php**
* Izmena korisnika
* **insert\_picture.php**
* Dodavanje slike
* **insert\_post\_tag.php**
* Dodavanje veze između post-a i taga
* **insert\_tag.php**
* Dodavanje taga
* **insert\_user.php**
* Dodavanje korisnika
* **newsletter.php**
* Dohvatanje svih korisnika projavljenih na bilten
* **pictures.php**
* Dohvatanje svih slika
* **post\_tag.php**
* Dohvatanje svih veza između objave i tagova
* **posts.php**
* Dohvatanje svih objava
* **tags.php**
* Dohvatanje svih tagova
* **users.php**
* Dohvatanje svih korisnika

## Izgled stranica



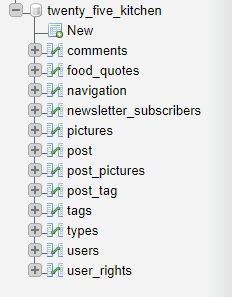
# ORGANIZACIJA

## Organizacija sajta





## Organizacija baze





## Sitemap.xml

<?xml version="1.0"?>

<urlset xlmns="http://www.sitemaps.org/schemas/sitemap/0.9">

<url>

<loc>http://25h-kitchen.epizy.com/index.php</loc>

<lastmod>2019-11-06</lastmod>

<changefreq>daily</changefreq>

<priority>1</priority>

</url>

</urlset>

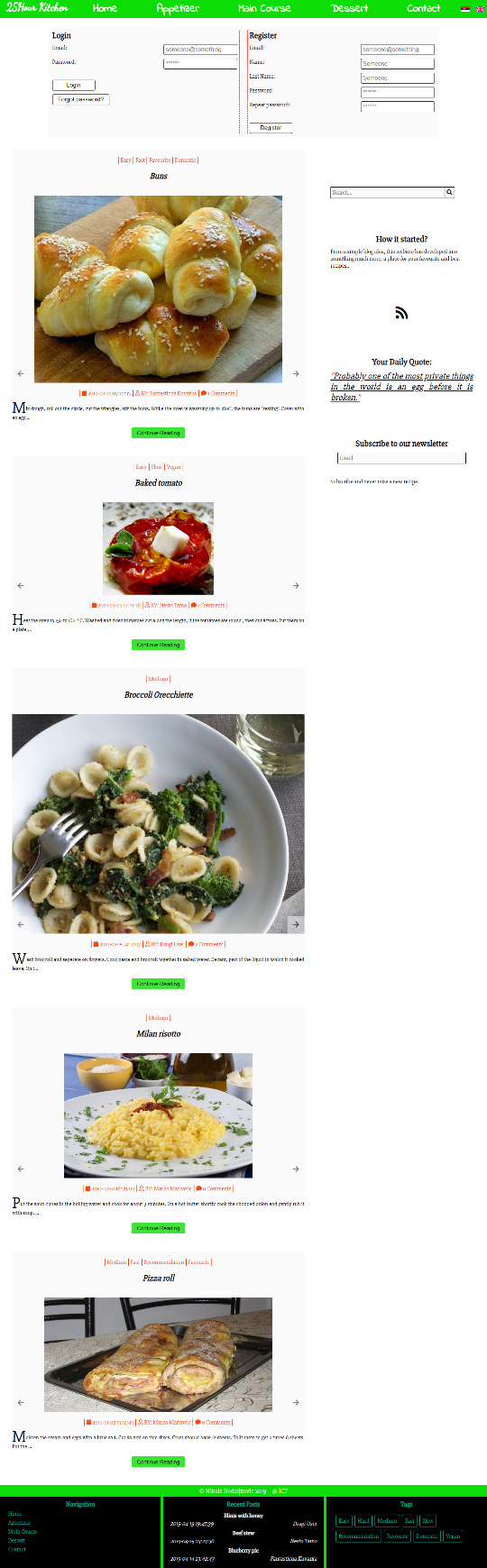


## Slike stranica sa funkcionalnostima



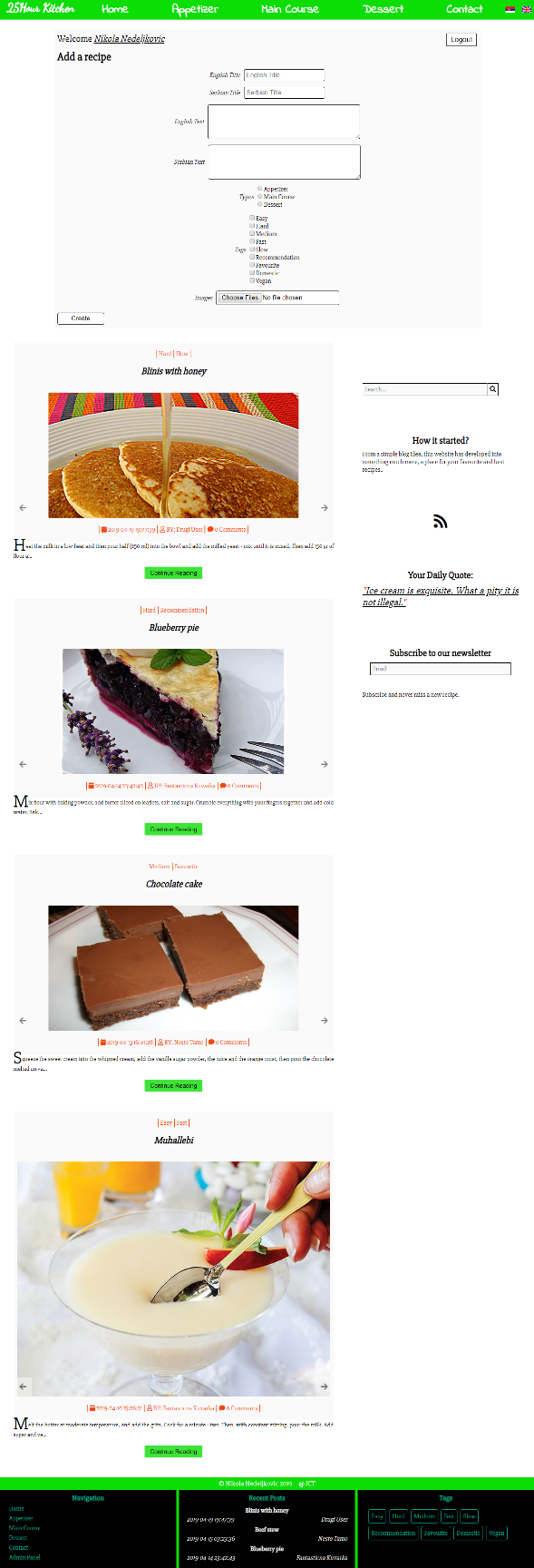
main\_template.php

* Animirani gif, navigacija, najnovije objave, tagovi, promena jezika



food\_post\_template.php

* Prikazivanje svih objava za tu odabranu objavu, vidi više, login, logout, register, forgot password, pretraga, prijava na bilten, kopiranje RSS-a **U ZAVISNOSTI** od izabranog jezika



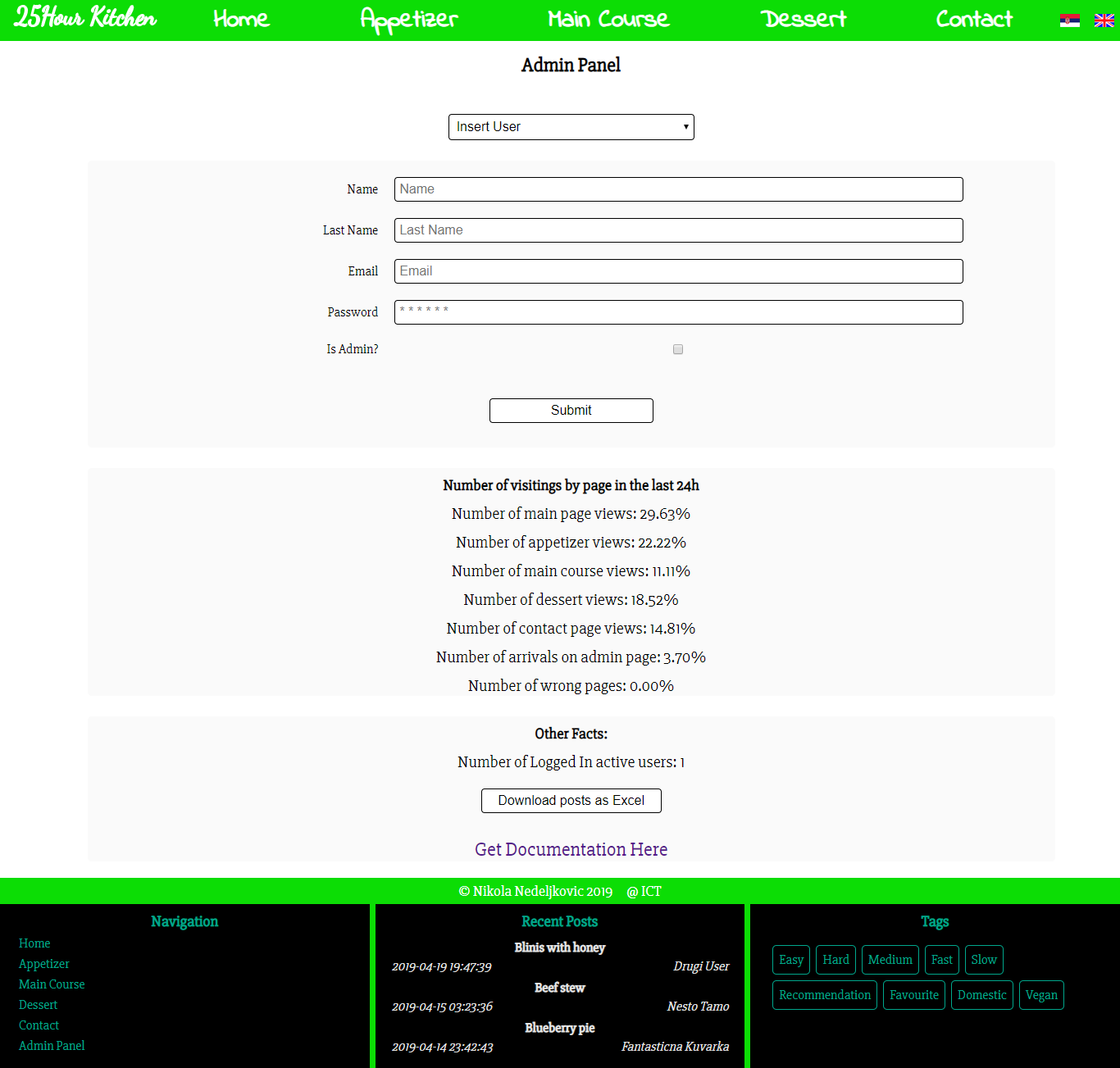
food\_post\_template.php

* Dodavanje novih objava, i odjavljivanje uz ono prethodno



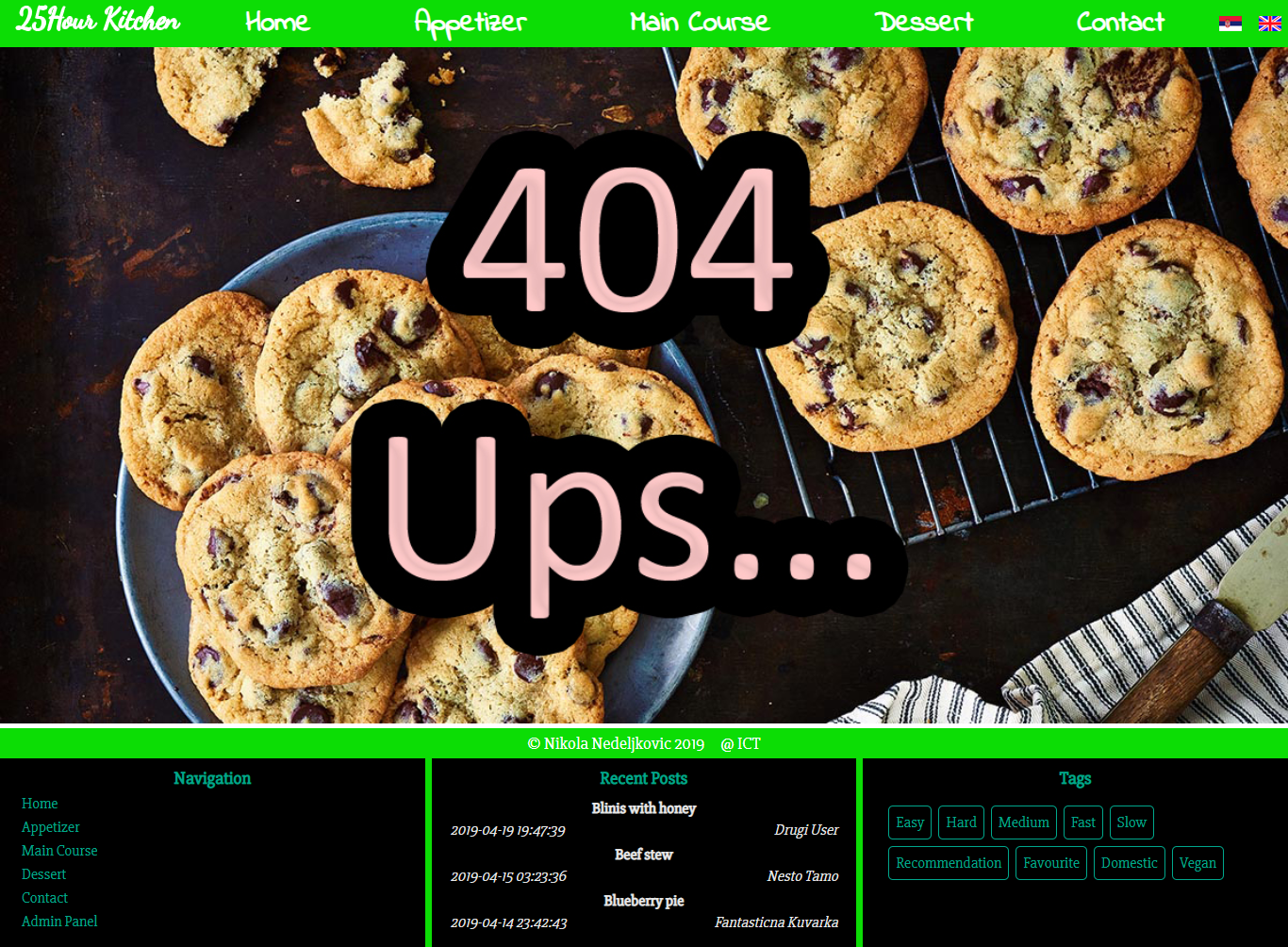
contact\_template.php

* Kontakt forma za administratore, o autoru, download u word



admin\_template.php

* Radnja svih admin funkcija, pregld posete stranicama, broj logovanih korisnika, export u excel



page\_404.php

* Stranica greške

# KODOVI

## index.php

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>

<meta name="author" content="Nikola Nedeljkovic"/>

<meta name="copyright" content="RarOks 1008 Entertainment @ 2019"/>

<meta name="robots" content="index,follow"/>

<link rel="shortcut icon" href="favicon.ico"/>

<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.1/css/all.css" integrity="sha384-50oBUHEmvpQ+1lW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28anvf" crossorigin="anonymous"/>

<link rel="stylesheet" href="assets/css/style.css"/>

<link rel="stylesheet" href="assets/css/main\_template.css"/>

<link rel="stylesheet" href="assets/css/header\_template.css"/>

<link rel="stylesheet" href="assets/css/footer\_template.css"/>

<link rel="stylesheet" href="assets/css/food\_post\_template.css"/>

<link rel="stylesheet" href="assets/css/add\_post\_template.css"/>

<link rel="stylesheet" href="assets/css/login\_template.css"/>

<link rel="stylesheet" href="assets/css/contact\_template.css"/>

<link rel="stylesheet" href="assets/css/admin\_template.css"/>

<link rel="stylesheet" href="assets/css/page\_404.css"/>

</head>

<body>

<?php

require\_once "config/connection.php";

session\_start();

include("views/header\_template.php");

echo "<div id=\"main\_panel\">";

if (isset($\_GET["page"])) {

switch($\_GET["page"]) {

case "main\_template":

zabeleziPristupStranici($\_GET["page"]);

include "views\main\_template.php";

break;

case "appetizer":

case "main\_course":

case "dessert":

zabeleziPristupStranici($\_GET["page"]);

include "views/food\_post\_template.php";

break;

case "contact\_template":

zabeleziPristupStranici($\_GET["page"]);

include "views/contact\_template.php";

break;

case "admin\_template":

zabeleziPristupStranici($\_GET["page"]);

include "views/admin\_template.php";

break;

default:

zabeleziPristupStranici("page\_404");

include "views/404.php";

break;

}

} else {

include "views\main\_template.php";

}

echo "</div>";

include("views/footer\_template.php");

?>

<div class="notification">

<span></span>

<span></span>

</div>

<h1></h1>

</body>

<script type="text/javascript" src="assets/js/3rd/jquery.min.js"></script>

<script type="text/javascript" src="assets/js/main.js"></script>

<script type="text/javascript" src="assets/js/header\_template.js"></script>

<script type="text/javascript" src="assets/js/footer\_template.js"></script>

<script type="text/javascript" src="assets/js/main\_template.js"></script>

<script type="text/javascript" src="assets/js/food\_post\_template.js"></script>

<script type="text/javascript" src="assets/js/add\_post\_template.js"></script>

<script type="text/javascript" src="assets/js/login\_template.js"></script>

<script type="text/javascript" src="assets/js/contact\_template.js"></script>

<script type="text/javascript" src="assets/js/admin\_template.js"></script>

</html>



## views/login\_template.php

<div class="landing\_screen">

<h2></h2>

<img src="assets/images/baking.gif" alt=""/>

</div>

## views/header\_template.php

<div class="header">

<div class="logo\_and\_nav">

<div class="logo"><img src="assets/images/logo.png" alt="Logo"/></div>

<span class="fas fa-bars" id="menu\_click"></span>

</div>

<ul id="header\_content"></ul>

<div class="language">

<img src="assets/images/serbia\_flag.png" alt="" data-img="srb"/>

<img src="assets/images/UK\_flag.jpg" alt="" data-img="eng"/>

</div>

</div>



## views/footer\_template.php

<div class="footer">

<p class="copyright">&copy; Nikola Nedeljkovic 2019 &nbsp;&nbsp;&nbsp; @ ICT</p>

<div class="footer\_holder">

<div class="footer\_part footer\_navigation">

<h3>Navigation</h3>

<ul id="navigation\_footer">

<?php

if (isset($\_SESSION["user"])) {

if ($\_SESSION["user"]->ID == 1) {

echo '<li><a href="index.php?page=admin\_template">Admin Panel</a></li>';

}

}

?>

</ul>

</div>

<div class="footer\_part footer\_recents">

<h3>Recent Posts</h3>

<div id="recent\_posts\_footer"></div>

</div>

<div class="footer\_part footer\_tags">

<h3>Tags</h3>

<div id="tags\_footer"></div>

</div>

</div>

</div>



## views/food\_post\_template.php

<div id="new\_post\_area">

<?php

if (isset($\_SESSION['user'])) {

echo "<div class='add\_post\_area'>";

echo "<div class='welcome\_user'><div><span></span><span>".$\_SESSION['user']->name." ". $\_SESSION['user']->last\_name."</span></div><button id='logout'></button></div>";

echo "<div id='add\_post'>";

include("add\_post\_template.php");

echo "</div>";

echo "</div>";

} else {

include("login\_template.php");

}

?>

</div>

<div id="main\_panel\_load\_food"></div>

<div id="main\_panel\_load\_edit">

<div class="search\_field">

<input type="text" name="search" id="search" placeholder=""/>

<button><span class="fa fa-search" aria-hidden="true"></span></button>

</div>

<div class="how\_it\_started">

<h2></h2>

<p></p>

</div>

<div class="rss\_feed">

<button><span class="fas fa-rss"></span></button>

</div>

<div class="quote\_field">

<h2></h2>

<p></p>

</div>

<div class="subscribe\_newsletter">

<h2></h2>

<input type="text" name="subscribe" id="subscribe" placeholder=""/>

<p id="subscribe\_error"></p>

<p id="subscribe\_text"></p>

</div>

</div>



## views/contact\_template.php

<div id="contact\_template">

<div class="contact\_area">

<h2></h2>

<input type="email" name="message\_mail" id="message\_mail" placeholder=""/>

<textarea rows="8" cols="40" name="send\_message" id="send\_message"></textarea>

<p id="message\_error"></p>

<button name="send\_contact" id="sent\_contact"></button>

</div>

<div class="author">

<h2></h2>

<img src="assets/images/author.jpg" alt=""/>

<div class="author\_name author\_data">

<p></p>

<p>Nikola Nedeljković</p>

</div>

<div class="author\_birth author\_data">

<p></p>

<p>06.05.1996</p>

</div>

<div class="author\_mail author\_data">

<p></p>

<p>nedeljkovic.nikola.ks&#64;gmail.com</p>

</div>

<div class="author\_number author\_data">

<p></p>

<p>064/45-71-380</p>

</div>

<div class="author\_work author\_data">

<p></p>

<p>Junior JS Developer @ Summa Labs d.o.o</p>

</div>

<button name="download\_author\_data" id="download\_author\_data"></button>

</div>

</div>



## views/admin\_template.php

<div id="admin\_panel">

<h2></h2>

<select name="admin\_option" id="admin\_option"></select>

<div class="admin\_controls">

<select name="admin\_extra\_option" id="admin\_extra\_option"></select>

<div id="admin\_extra\_space"></div>

</div>

<div class="admin\_controls file\_controls">

<h3></h3>

<div class="admin\_file\_holder visiting\_main">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_appetizer">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_main\_course">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_dessert">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_contact">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_admin">

<span></span>&nbsp;<span></span>

</div>

<div class="admin\_file\_holder visiting\_404">

<span></span>&nbsp;<span></span>

</div>

</div>

<div class="admin\_controls other\_controls">

<h3></h3>

<div class="admin\_file\_holder number\_logged\_users">

<span></span>&nbsp;<span></span>

</div>

<button name="export\_to\_excel" id="export\_to\_excel"></button>

<a href="documentation/documentation.pdf"></a>

</div>

</div>



## views/add\_post\_template.php

<form onsubmit="return add\_post\_param\_check()" method="POST" action="models/add\_post.php" class="add\_post\_form" name="add\_post\_form" enctype="multipart/form-data">

<h2></h2>

<div class="add\_post\_param">

<label for="add\_post\_title"></label>

<input type="text" name="add\_post\_title" id="add\_post\_title" placeholder=""/>

</div>

<div class="add\_post\_param">

<label for="add\_post\_srb\_title"></label>

<input type="text" name="add\_post\_srb\_title" id="add\_post\_srb\_title" placeholder=""/>

</div>

<div class="add\_post\_param">

<label for="add\_post\_text"></label>

<textarea rows="4" cols="40" name="add\_post\_text" id="add\_post\_text"></textarea>

</div>

<div class="add\_post\_param">

<label for="add\_post\_srb\_text"></label>

<textarea rows="4" cols="40" name="add\_post\_srb\_text" id="add\_post\_srb\_text"></textarea>

</div>

<div class="add\_post\_param">

<label for="add\_post\_type"></label>

<div class="add\_post\_param\_holder add\_post\_type\_holder"></div>

</div>

<div class="add\_post\_param">

<label for="add\_post\_tag"></label>

<div class="add\_post\_param\_holder add\_post\_tag\_holder"></div>

</div>

<div class="add\_post\_param">

<label for="add\_post\_image"></label>

<input type="file" name="add\_post\_image[]" id="add\_post\_image" multiple accept=".gif, .jpeg, .jpg, .png"/>

</div>

<p id="add\_post\_error"></p>

<input type="submit" name="add\_post\_button" id="add\_post\_button"/>

</form>



## views/404.php

<div class="page\_404">

<img src="assets/images/404.png" alt="404"/>

</div>



## models/add\_post.php

<?php

include("../config/connection.php");

include("composer/contact\_mail.php");

session\_start();

$error = "";

if (isset($\_POST['add\_post\_button'])) {

$title = $\_POST['add\_post\_title'];

$srbTitle = $\_POST['add\_post\_srb\_title'];

$text = $\_POST['add\_post\_text'];

$srbText = $\_POST['add\_post\_srb\_text'];

$type = $\_POST['add\_post\_type'];

$tags = $\_POST['add\_post\_tag'];

$images = $\_FILES['add\_post\_image'];

$title\_patern = "/^[A-ZŠĐŽČĆ][a-z0-9ščćđž]+$/";

$date = date("Y-m-d H:i:s", time());

$user\_id = $\_SESSION["user"]->ID;

$number\_of\_images = count($images["name"]);

if ($title == "" || $srbTitle == "" || $text == "" || $srbText == "") { $error = "Fields are empty"; }

if (!$type) { $error = "Type not selected"; }

if (!preg\_match($title\_patern, $title) || !preg\_match($title\_patern, $srbTitle)) { $error = "Titles are not valid"; }

if (count($tags) <= 0 || $number\_of\_images <= 0) { $error = "Tags or Images not selected"; }

try {

$conn->beginTransaction();

$query = "INSERT INTO post (type\_id, user\_id, title, SRBtitle, text, SRBtext, date) VALUES (:type\_id, :user\_id, :title, :SRBtitle, :text, :SRBtext, :date)";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":type\_id", $type);

$priprema->bindParam(":user\_id", $user\_id);

$priprema->bindParam(":title", $title);

$priprema->bindParam(":SRBtitle", $srbTitle);

$priprema->bindParam(":text", $text);

$priprema->bindParam(":SRBtext", $srbText);

$priprema->bindParam(":date", $date);

$rezultat = $priprema->execute();

if (!$rezultat) { $error = "Error adding post"; }

$last\_post\_id = $conn->lastInsertId();

$tag\_query = "INSERT INTO post\_tag (ID\_post, ID\_tag) VALUES";

foreach ($tags as $tag) {

$tag\_query .= " (".$last\_post\_id.",".$tag."),";

}

$tag\_query = rtrim($tag\_query,',');

$tag\_priprema = $conn -> prepare($tag\_query);

$tag\_rezultat = $tag\_priprema->execute();

if (!$tag\_rezultat) { $error = "Error adding tags"; }

for ($i = 0; $i < $number\_of\_images; $i++) {

$image\_name = $images["name"][$i];

$image\_size = $images["size"][$i];

$image\_tmp = $images["tmp\_name"][$i];

$imageFileType = strtolower(pathinfo($image\_name,PATHINFO\_EXTENSION));

if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg" && $imageFileType != "gif" ) { $error = "Not valid image"; }

$alt = $title . " " . ($i + 1);

$srbAlt = $srbTitle . " " . ($i + 1);

$upload\_dir = '../assets/images/recipes/';

$src = 'assets/images/recipes/'.$image\_name;

$add\_picture = "INSERT INTO pictures (src, alt, SRBalt) VALUES (:src, :alt, :srbAlt)";

$priprema\_pic = $conn -> prepare($add\_picture);

$priprema\_pic->bindParam(":src", $src);

$priprema\_pic->bindParam(":alt", $alt);

$priprema\_pic->bindParam(":srbAlt", $srbAlt);

$rezultat\_pic = $priprema\_pic->execute();

if (!$tag\_rezultat) { $error = "Error adding image"; }

$last\_picture\_id = $conn->lastInsertId();

$image\_path = $upload\_dir."food\_".$last\_picture\_id.".".$imageFileType;

list($sirina, $visina) = getimagesize($image\_tmp);

$novaVisina = 200;

$novaSirina = ($novaVisina/$visina) \* $sirina;

$novaSlika = imagecreatetruecolor($novaSirina, $novaVisina);

$postojecaSlika = null;

switch ($imageFileType) {

case 'jpeg':

case 'jpg':

$postojecaSlika = imagecreatefromjpeg($image\_tmp);

break;

case 'png':

$postojecaSlika = imagecreatefrompng($image\_tmp);

break;

case 'gif':

$postojecaSlika = imagecreatefromgif($image\_tmp);

break;

}

imagecopyresampled($novaSlika, $postojecaSlika, 0, 0, 0, 0, $novaSirina, $novaVisina, $sirina, $visina);

switch ($imageFileType) {

case 'jpeg':

case 'jpg':

imagejpeg($novaSlika, $image\_path, 75);

break;

case 'png':

imagepng($novaSlika, $image\_path);

break;

case 'gif':

imagegif($novaSlika, $image\_path);

break;

}

$new\_src = 'assets/images/recipes/food\_'.$last\_picture\_id.".".$imageFileType;

$change\_pic = "UPDATE pictures SET src = :src WHERE ID = :id";

$priprema\_change = $conn -> prepare($change\_pic);

$priprema\_change->bindParam(":src", $new\_src);

$priprema\_change->bindParam(":id", $last\_picture\_id);

$rezultat\_change = $priprema\_change->execute();

if (!$rezultat\_change) { $error = "Error changing name in database"; }

$post\_picture\_query = "INSERT INTO post\_pictures (ID\_post, ID\_picture) VALUES (:post\_id, :picture\_id)";

$priprema\_pp = $conn -> prepare($post\_picture\_query);

$priprema\_pp->bindParam(":post\_id", $last\_post\_id);

$priprema\_pp->bindParam(":picture\_id", $last\_picture\_id);

$rezultat\_pp = $priprema\_pp->execute();

if (!$rezultat\_pp) { $error = "Error adding to table"; }

$srb\_xml = simplexml\_load\_file("../data/rss-srb.xml") or die("Error: Cannot create object");

$srb\_channel = $srb\_xml->channel;

$item = $srb\_channel->addChild('item');

$item->addChild('title', $srbTitle);

$item->addChild('link', 'http://25h-kitchen.epizy.com/index.php');

$item->addChild('description', $\_SESSION["user"]->name.' '.$\_SESSION["user"]->last\_name);

$srb\_xml->asXML("../data/rss-srb.xml");

$eng\_xml = simplexml\_load\_file("../data/rss-eng.xml") or die("Error: Cannot create object");

$eng\_channel = $eng\_xml->channel;

$item = $eng\_channel->addChild('item');

$item->addChild('title', $title);

$item->addChild('link', 'http://25h-kitchen.epizy.com/index.php');

$item->addChild('description', $\_SESSION["user"]->name.' '.$\_SESSION["user"]->last\_name);

$eng\_xml->asXML("../data/rss-eng.xml");

}

if ($error != "") {

echo $error;

$conn->rollback();

} else {

$conn->commit();

$query = "SELECT \* FROM newsletter\_subscribers";

$subscribers = executeQuery($query);

foreach ($subscribers as $subscriber) {

$result = sendOthersMail($subscriber->email, "New post: ".$title);

}

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

die();

}

}catch (PDOException $e) {

echo $e->getMessage();

$conn->rollback();

}

} else {

$error = "Must send by form!";

echo $error;

}

?>



## models/comment.php

<?php

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

session\_start();

include("../config/connection.php");

$code = 500;

$error = "";

$post\_id = $\_POST["post\_id"];

$comment\_text = $\_POST["comment\_text"];

if (strlen($comment\_text) < 5) { $error = "empty\_comment"; }

if (!isset($\_SESSION["user"])) { $error = "log\_to\_comment"; }

if ($error == "") {

$user\_id = $\_SESSION["user"]->ID;

$date = date("Y-m-d H:i:s", time());

$query = "INSERT INTO comments (user\_id, post\_id, text, date) VALUES (:user\_id, :post\_id, :text, :date)";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":user\_id", $user\_id);

$priprema->bindParam(":post\_id", $post\_id);

$priprema->bindParam(":text", $comment\_text);

$priprema->bindParam(":date", $date);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 200;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/comments.php

<?php

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

include("../config/connection.php");

$code = 500;

$postId = $\_GET["postId"];

$query = "SELECT c.text AS commentText, c.date AS commentDate, u.name AS userName, u.last\_name AS userLastName FROM comments c INNER JOIN users u ON c.user\_id = u.ID WHERE c.post\_id = :postId";

$priprema = $conn->prepare($query);

$priprema->bindParam(":postId", $postId);

$rezultat = $priprema->execute();

if ($rezultat) {

$posts = $priprema->FetchAll();

$code = 200;

echo json\_encode($posts);

} else {

$code = 504;

echo json\_encode("Update failed");

}

http\_response\_code($code);

?>



## models/contact.php

<?php

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

include("composer/contact\_mail.php");

$mail = $\_POST["mail"];

$text = $\_POST["text"];

$code = 500;

$error = "";

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

if ($mail == "" || $text == "") { $error = "empty\_field"; }

if (!preg\_match($email\_pattern, $mail)) { $error = "mail\_invalid"; }

if ($error == "") {

$result = sendContactMail($mail, $text);

if ($result) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/export/excel.php

<?php

http\_response\_code(200);

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

require\_once($\_SERVER["DOCUMENT\_ROOT"]."/config/connection.php");

require\_once ABSOLUTE\_PATH.'/models/composer/vendor/autoload.php';

$save\_loc = exec('echo %SystemDrive%') . '\\Users\\' . get\_current\_user() . '\\Desktop';

$posts = executeQuery("SELECT p.ID AS id, title, SRBtitle, text, SRBtext, name, last\_name FROM post p INNER JOIN users u ON p.user\_id = u.ID");

$spreadsheet = new Spreadsheet();

$sheet = $spreadsheet->getActiveSheet();

$sheet->setCellValue('A1', 'Post ID:');

$sheet->setCellValue('B1', 'Title:');

$sheet->setCellValue('C1', 'Naslov:');

$sheet->setCellValue('D1', 'Text:');

$sheet->setCellValue('E1', 'Tekst:');

$sheet->setCellValue('F1', 'Author/Autor:');

$i = 0;

foreach ($posts as $post) {

$sheet->setCellValue('A'.($i + 2), $post->id);

$sheet->setCellValue('B'.($i + 2), $post->title);

$sheet->setCellValue('C'.($i + 2), $post->SRBtitle);

$sheet->setCellValue('D'.($i + 2), $post->text);

$sheet->setCellValue('E'.($i + 2), $post->SRBtext);

$sheet->setCellValue('F'.($i + 2), $post->name." ".$post->last\_name);

$i += 1;

}

$writer = new Xlsx($spreadsheet);

$writer->save($save\_loc.'\\posts.xlsx');

echo "OK";

?>



## models/forgot\_password.php

<?php

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

include("composer/contact\_mail.php");

$mail = $\_POST["email"];

$code = 500;

$error = "";

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

if ($mail == "") { $error = "empty\_field"; }

if (!preg\_match($email\_pattern, $mail)) { $error = "mail\_invalid"; }

if ($error == "") {

$result = sendContactMail($mail, "User: ".$mail. " requests password change.");

if ($result) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/logged\_users.php

<?php

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

include("../config/connection.php");

$logged = executeQuery("SELECT COUNT(\*) AS logged\_users FROM users WHERE LastTimeSeen > DATE\_SUB(NOW(), INTERVAL 5 MINUTE)");

echo json\_encode($logged);

?>



## models/login.php

<?php

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

session\_start();

include("../config/connection.php");

include("composer/contact\_mail.php");

$code = 500;

$error = "";

$email = $\_POST["email"];

$password = $\_POST["password"];

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

if ($email == "" || $password == "") { $error = "empty\_field"; }

if (!preg\_match($email\_pattern, $email)) { $error = "not\_valid\_email"; }

if (strlen($password) < 6) { $error = "password\_short"; }

if ($error == "") {

$query = "SELECT \* FROM users WHERE email=:email AND password=:password";

$password = md5($password);

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$priprema->bindParam(":password", $password);

$rezultat = $priprema->execute();

if ($rezultat) {

if ($priprema -> rowCount() == 1) {

$user = $priprema -> Fetch();

$\_SESSION["user"] = $user;

$code = 200;

echo json\_encode("Successful");

} else if ($priprema -> rowCount() == 0) {

$code = 203;

$result = sendOthersMail($email, "Somebody tried to access your account on ".date('d-m-Y H:i:s'));

echo json\_encode("wrong\_params");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>

## models/logout.php

<?php

session\_start();

if(isset($\_SESSION['user'])) {

unset($\_SESSION['user']);

session\_destroy();

}

echo true;

?>



## models/navigation.php

<?php

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

include("../config/connection.php");

$query = "SELECT \* FROM navigation";

$navigation = executeQuery($query);

echo json\_encode($navigation);

?>



## models/page\_access\_info.php

<?php

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

include("../config/connection.php");

$podaci = file(LOG\_FAJL);

$objekat = array("main\_template" => 0,

"appetizer" => 0,

"main\_course" => 0,

"dessert" => 0,

"contact\_template" => 0,

"admin\_template" => 0,

"page\_404" => 0);

foreach ($podaci as $key => $value) {

$vrednost = explode("\t", $value);

$date\_from\_file = strtotime($vrednost[1]);

if (time() - $date\_from\_file <= 24 \* 60 \* 60) {

$objekat[$vrednost[0]] += 1;

}

}

echo json\_encode($objekat);

?>



## models/posts.php

<?php

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

include("../config/connection.php");

$code = 500;

$type\_id = $\_GET["type\_id"];

$selected\_tags = $\_GET["selected\_tags"];

$tag\_count = count($selected\_tags);

$query = "SELECT p.ID AS id, p.title AS title, p.SRBtitle AS SRBtitle, p.text AS postText, p.SRBtext AS SRBtext, p.date AS postDate, u.name AS name, u.last\_name AS last\_name, pic.src AS picsrc, pic.ID AS picid, pic.alt AS picalt, pic.SRBalt AS picSRBalt, t.name AS tagname, t.ID AS tagid, t.SRBname AS tagSRBname, (SELECT COUNT(\*) FROM comments c WHERE c.post\_id = p.ID) AS commentNumber FROM post p INNER JOIN users u ON p.user\_id = u.ID INNER JOIN post\_pictures pp ON p.ID = pp.ID\_post INNER JOIN pictures pic ON pp.ID\_picture = pic.ID INNER JOIN post\_tag pt ON p.ID = pt.ID\_post INNER JOIN tags t ON t.ID = pt.ID\_tag WHERE type\_id = :type\_id";

if ($tag\_count >= 1 && $selected\_tags[0] != "") {

$query .= " AND pt.ID\_tag IN (".implode(',',$selected\_tags).")";

}

if (isset($\_GET["text"])) {

$text = $\_GET["text"];

$text = strtolower($text);

$query .= " AND LOWER(p.title) LIKE '%".$text."%'";

}

$query .= " ORDER BY p.date DESC";

$priprema = $conn->prepare($query);

$priprema->bindParam(":type\_id", $type\_id);

$rezultat = $priprema->execute();

if ($rezultat) {

$posts = $priprema->FetchAll();

$code = 200;

echo json\_encode($posts);

} else {

$code = 504;

echo json\_encode("Update failed");

}

http\_response\_code($code);

?>



## models/random\_quote.php

<?php

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

include("../config/connection.php");

$query = "SELECT COUNT(\*) AS number FROM food\_quotes";

$max\_number = executeQuery($query);

$random\_number = rand(0, $max\_number[0]->number - 1);

$query = "SELECT \* FROM food\_quotes";

$food\_quotes = executeQuery($query);

echo json\_encode($food\_quotes[$random\_number]);

?>



## models/register.php

<?php

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

include("../config/connection.php");

$code = 500;

$error = "";

$email = $\_POST["email"];

$name = $\_POST["name"];

$lastname = $\_POST["lastname"];

$password = $\_POST["password"];

$password\_repeat = $\_POST["password\_repeat"];

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

$name\_pat = "/^[A-Z][a-z]+$/";

$lastname\_pat = "/^([A-Z][a-z]+)+$/";

if ($email == "" || $name == "" || $lastname == "" || $password == "" || $password\_repeat == "") { $error = "empty\_field"; }

if (strlen($password) < 6) { $error = "password\_short"; }

if ($password != $password\_repeat) { $error = "passwords\_not\_match"; }

if (!preg\_match($email\_pattern, $email)) { $error = "not\_valid\_email"; }

if (!preg\_match($name\_pat, $name) || !preg\_match($lastname\_pat, $lastname)) { $error = "name\_invalid"; }

$query = "SELECT COUNT(email) AS rowCount FROM users WHERE email = :email";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "email\_taken"; }

if ($error == "") {

$query = "INSERT INTO users (email, password, rights\_id, name, last\_name) VALUES (:email, :password, 2, :name, :lastname)";

$password = md5($password);

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$priprema->bindParam(":password", $password);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":lastname", $lastname);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/save\_word.php

<?php

@ ob\_start();

header("Content-Type: application/octet-stream");

header("Content-Disposition: attachment; filename=nikola\_nedeljkovic.doc");

http\_response\_code(200);

$save\_loc = exec('echo %SystemDrive%') . '\\Users\\' . get\_current\_user() . '\\Desktop';

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

$word->Visible = 1;

$word->Documents->Add();

$word->Selection->TypeText($\_POST['author\_name']);

$word->Selection->TypeText($\_POST['author\_name\_val']);

$word->Selection->TypeText("\n");

$word->Selection->TypeText($\_POST['author\_birth']);

$word->Selection->TypeText($\_POST['author\_birth\_val']);

$word->Selection->TypeText("\n");

$word->Selection->TypeText($\_POST['author\_mail']);

$word->Selection->TypeText($\_POST['author\_mail\_val']);

$word->Selection->TypeText("\n");

$word->Selection->TypeText($\_POST['author\_number']);

$word->Selection->TypeText($\_POST['author\_number\_val']);

$word->Selection->TypeText("\n");

$word->Selection->TypeText($\_POST['author\_work']);

$word->Selection->TypeText($\_POST['author\_work\_val']);

$word->Documents[1]->SaveAs($save\_loc.'\\nikola\_nedeljkovic.doc');

$word->Quit();

$word = null;

echo "OK";

?>



## models/single\_post.php

<?php

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

include("../config/connection.php");

$code = 500;

$postId = $\_POST["postId"];

$query = "SELECT p.ID AS id, p.title AS title, p.SRBtitle AS SRBtitle, p.text AS postText, p.SRBtext AS SRBtext, p.date AS postDate, u.name AS name, u.last\_name AS last\_name, pic.src AS picsrc, pic.ID AS picid, pic.alt AS picalt, pic.SRBalt AS picSRBalt, t.name AS tagname, t.ID AS tagid, t.SRBname AS tagSRBname, (SELECT COUNT(\*) FROM comments c WHERE c.post\_id = p.ID) AS commentNumber FROM post p INNER JOIN users u ON p.user\_id = u.ID INNER JOIN post\_pictures pp ON p.ID = pp.ID\_post INNER JOIN pictures pic ON pp.ID\_picture = pic.ID INNER JOIN post\_tag pt ON p.ID = pt.ID\_post INNER JOIN tags t ON t.ID = pt.ID\_tag WHERE p.ID = :postId";

$priprema = $conn->prepare($query);

$priprema->bindParam(":postId", $postId);

$rezultat = $priprema->execute();

if ($rezultat) {

$posts = $priprema->FetchAll();

$code = 200;

echo json\_encode($posts);

} else {

$code = 504;

echo json\_encode("Update failed");

}

http\_response\_code($code);

?>



## models/subscribe.php

<?php

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

include("../config/connection.php");

$code = 500;

$error = "";

$subscription\_mail = $\_POST["subscription\_mail"];

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

if ($subscription\_mail == "") { $error = "subscribe\_empty\_field"; }

if (!preg\_match($email\_pattern, $subscription\_mail)) { $error = "not\_valid\_email"; }

$query = "SELECT COUNT(email) AS rowCount FROM newsletter\_subscribers WHERE email = :subscription\_mail";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":subscription\_mail", $subscription\_mail);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "subscription\_mail\_exists"; }

if ($error == "") {

$query = "INSERT INTO newsletter\_subscribers (email) VALUES (:subscription\_mail)";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":subscription\_mail", $subscription\_mail);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/tags.php

<?php

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

include("../config/connection.php");

$query = "SELECT \* FROM tags";

$tags = executeQuery($query);

echo json\_encode($tags);

?>



## models/top\_posts.php

<?php

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

include("../config/connection.php");

$query = "SELECT p.title AS title, p.SRBtitle AS SRBtitle, p.date AS postDate, u.name AS name, u.last\_name AS lastname FROM post p INNER JOIN users u ON p.user\_id = u.ID ORDER BY p.date DESC LIMIT 3";

$top\_posts = executeQuery($query);

echo json\_encode($top\_posts);

?>



## views/types.php

<?php

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

include("../config/connection.php");

$query = "SELECT \* FROM types";

$types = executeQuery($query);

echo json\_encode($types);

?>



## views/write\_error.php

<?php

include("../config/connection.php");

$text = $\_POST["text"];

$open = fopen(ERROR\_FAJL, "a");

if ($open) {

$date = date('d-m-Y H:i:s');

fwrite($open, "{$content}\t{$date}\t\n");

fclose($open);

}

echo "OK";

?>



## models/composer/contact\_mail.php

<?php

use PHPMailer\PHPMailer\PHPMailer;

use PHPMailer\PHPMailer\Exception;

require\_once($\_SERVER["DOCUMENT\_ROOT"]."/config/connection.php");

require\_once ABSOLUTE\_PATH.'/models/composer/vendor/autoload.php';

function sendContactMail ($to, $content) {

$mail = new PHPMailer(true);

try {

$mail->SMTPDebug = 0;

$mail->isSMTP();

$mail->Host = 'smtp.gmail.com';

$mail->SMTPAuth = true;

$mail->Username = 'twentyfivekitchen@gmail.com';

$mail->Password = '25Kitchen';

$mail->SMTPSecure = 'tls';

$mail->Port = 587;

$mail->setFrom('twentyfivekitchen@gmail.com', '25Hour Kitchen');

$mail->addAddress('twentyfivekitchen@gmail.com');

$mail->addReplyTo($to, 'Information: ');

$mail->isHTML(true);

$mail->Subject = '25Hours Kitchen Contact Form';

$mail->Body = $content;

$mail->send();

return true;

} catch (Exception $e) {

echo $e;

return false;

}

}

function sendOthersMail ($to, $content) {

$mail = new PHPMailer(true);

try {

$mail->SMTPDebug = 0;

$mail->isSMTP();

$mail->Host = 'smtp.gmail.com';

$mail->SMTPAuth = true;

$mail->Username = 'twentyfivekitchen@gmail.com';

$mail->Password = '25Kitchen';

$mail->SMTPSecure = 'tls';

$mail->Port = 587;

$mail->setFrom($to, '25Hour Kitchen');

$mail->addAddress($to);

$mail->addReplyTo($to, 'Information: ');

$mail->isHTML(true);

$mail->Subject = '25Hours Kitchen Contact Form';

$mail->Body = $content;

$mail->send();

return true;

} catch (Exception $e) {

echo $e;

return false;

}

}

?>



## models/admin/comments.php

<?php

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

include("../../config/connection.php");

$query = "SELECT text, ID FROM comments";

$comments = executeQuery($query);

echo json\_encode($comments);

?>



## models/admin/delete\_comment.php

<?php

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

include("../../config/connection.php");

$code = 500;

$comment = $\_POST["comment"];

$query = "DELETE FROM comments WHERE ID = :comment";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":comment", $comment);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_newsletter.php

<?php

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

include("../../config/connection.php");

$code = 500;

$newsletter = $\_POST["newsletter"];

$query = "DELETE FROM newsletter\_subscribers WHERE ID = :newsletter";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":newsletter", $newsletter);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_picture.php

<?php

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

include("../../config/connection.php");

$code = 500;

$picture\_id = $\_POST["picture"];

$query\_select = "SELECT \* FROM pictures WHERE ID = :picture";

$priprema\_select = $conn -> prepare($query\_select);

$priprema\_select->bindParam(":picture", $picture\_id);

$rezultat\_select = $priprema\_select->execute();

if ($rezultat\_select) {

$pictures = $priprema\_select->FetchAll();

foreach ($pictures as $picture) {

$where = '../../'.$picture->src;

unlink($where);

}

$query = "DELETE FROM pictures WHERE ID = :picture";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":picture", $picture\_id);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_post\_tag.php

<?php

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

include("../../config/connection.php");

$code = 500;

$post\_tag = $\_POST["post\_tag"];

$ids = explode("-", $post\_tag);

$query = "DELETE FROM post\_tag WHERE ID\_post = :post\_id AND ID\_tag = :tag\_id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":post\_id", $ids[0]);

$priprema->bindParam(":tag\_id", $ids[1]);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_post.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$post = $\_POST["post"];

$conn->beginTransaction();

$query\_pictures = "SELECT \* FROM post\_pictures WHERE ID\_post = :post";

$priprema\_pictures = $conn -> prepare($query\_pictures);

$priprema\_pictures->bindParam(":post", $post);

$rezultat\_pictures = $priprema\_pictures->execute();

if ($rezultat\_pictures) {

$picture\_ids = $priprema\_pictures->FetchAll();

foreach ($picture\_ids as $picture\_id) {

$query\_pict = "SELECT \* FROM pictures WHERE ID = :pic\_id";

$priprema\_pict = $conn -> prepare($query\_pict);

$priprema\_pict->bindParam(":pic\_id", $picture\_id->ID\_picture);

$rezultat\_pict = $priprema\_pict->execute();

if ($rezultat\_pict) {

$pictures = $priprema\_pict->FetchAll();

foreach($pictures as $picture) {

$where = '../../'.$picture->src;

unlink($where);

$query\_pic = "DELETE FROM pictures WHERE ID = :picture";

$priprema\_pic = $conn -> prepare($query\_pic);

$priprema\_pic->bindParam(":picture", $picture->ID);

$rezultat\_pic = $priprema\_pic->execute();

if (!$rezultat\_pic) {

$error = "database\_error";

}

}

} else {

$error = "database\_error";

}

}

$query\_post = "SELECT \* FROM post WHERE ID = :post";

$priprema\_post = $conn ->prepare($query\_post);

$priprema\_post->bindParam(":post", $post);

$rezultat\_post = $priprema\_post->execute();

if ($rezultat\_post) {

$post\_rez = $priprema\_post -> Fetch();

$srb\_xml = simplexml\_load\_file("../../data/rss-srb.xml") or die("Error: Cannot create object");

$srb\_channel = $srb\_xml->channel;

$srb\_nodes = $srb\_xml->xpath('//channel/item/title[.="'.$post\_rez->SRBtitle.'"]/parent::\*');

$srb\_trazeni\_node = $srb\_nodes[0];

unset($srb\_trazeni\_node);

$srb\_xml->asXML("../../data/rss-srb.xml");

$eng\_xml = simplexml\_load\_file("../../data/rss-eng.xml") or die("Error: Cannot create object");

$eng\_channel = $eng\_xml->channel;

$eng\_nodes = $eng\_xml->xpath('//channel/item/title[.="'.$post\_rez->title.'"]/parent::\*');

$eng\_trazeni\_node = $eng\_nodes[0];

unset($eng\_trazeni\_node);

$eng\_xml->asXML("../../data/rss-eng.xml");

} else {

$error = "database\_error";

}

$query = "DELETE FROM post WHERE ID = :post";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":post", $post);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

if ($error == "") {

echo json\_encode("Successful");

$conn->commit();

} else {

$conn->rollback();

$code = 503;

echo $error;

}

} else {

$code = 503;

$conn->rollback();

echo json\_encode("database\_error");

}

} else {

$code = 503;

$conn->rollback();

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_tag.php

<?php

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

include("../../config/connection.php");

$code = 500;

$tag = $\_POST["tag"];

$query = "DELETE FROM tags WHERE ID = :tag";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":tag", $tag);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/delete\_user.php

<?php

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

include("../../config/connection.php");

$code = 500;

$user = $\_POST["user"];

$query = "DELETE FROM users WHERE ID = :user";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":user", $user);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 202;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

http\_response\_code($code);

?>



## models/admin/edit\_post.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$title = $\_POST["title"];

$srb\_title = $\_POST["srb\_title"];

$text = $\_POST["text"];

$srb\_text = $\_POST["srb\_text"];

$id = $\_POST["id"];

$title\_pat = "/^([A-Z][a-z]+)+$/";

if ($title == "" || $srb\_title == "" || $text == "" || $srb\_text == "") { $error = "empty\_field"; }

if (!preg\_match($title\_pat, $title) || !preg\_match($title\_pat, $srb\_title)) { $error = "title\_name\_invalid"; }

if ($error == "") {

$query\_post = "SELECT \* FROM post WHERE ID = :id";

$priprema\_post = $conn ->prepare($query\_post);

$priprema\_post->bindParam(":id", $id);

$rezultat\_post = $priprema\_post->execute();

if ($rezultat\_post) {

$post\_rez = $priprema\_post -> Fetch();

$srb\_xml = simplexml\_load\_file("../../data/rss-srb.xml") or die("Error: Cannot create object");

$srb\_nodes = $srb\_xml->xpath('//channel/item/title[.="'.$post\_rez->SRBtitle.'"]/parent::\*');

$srb\_trazeni\_node = $srb\_nodes[0];

$srb\_trazeni\_node->title = $srb\_title;

$srb\_xml->asXML("../../data/rss-srb.xml");

$eng\_xml = simplexml\_load\_file("../../data/rss-eng.xml") or die("Error: Cannot create object");

$eng\_nodes = $eng\_xml->xpath('//channel/item/title[.="'.$post\_rez->title.'"]/parent::\*');

$eng\_trazeni\_node = $eng\_nodes[0];

$eng\_trazeni\_node->title = $title;

$eng\_xml->asXML("../../data/rss-eng.xml");

$query = "UPDATE post SET title = :title, SRBtitle = :srb\_title, text = :text, SRBtext = :srb\_text WHERE ID = :id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":title", $title);

$priprema->bindParam(":srb\_title", $srb\_title);

$priprema->bindParam(":text", $text);

$priprema->bindParam(":srb\_text", $srb\_text);

$priprema->bindParam(":id", $id);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/edit\_tag.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$name = $\_POST["name"];

$srb\_name = $\_POST["srb\_name"];

$id = $\_POST["id"];

$name\_pat = "/^[A-Z][a-z]+$/";

if ($name == "" || $srb\_name == "") { $error = "empty\_field"; }

if (!preg\_match($name\_pat, $name) || !preg\_match($name\_pat, $srb\_name)) { $error = "tag\_name\_invalid"; }

$query = "SELECT COUNT(name) AS rowCount FROM tags WHERE (name = :name OR SRBname = :srb\_name) AND ID != :id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":srb\_name", $srb\_name);

$priprema->bindParam(":id", $id);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "tag\_exists"; }

if ($error == "") {

$query = "UPDATE tags SET name = :name, SRBname = :srb\_name WHERE ID = :id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":srb\_name", $srb\_name);

$priprema->bindParam(":id", $id);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/edit\_user.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$email = $\_POST["email"];

$name = $\_POST["name"];

$lastname = $\_POST["lastname"];

$id = $\_POST["id"];

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

$name\_pat = "/^[A-Z][a-z]+$/";

$lastname\_pat = "/^([A-Z][a-z]+)+$/";

if ($email == "" || $name == "" || $lastname == "") { $error = "empty\_field"; }

if (!preg\_match($email\_pattern, $email)) { $error = "not\_valid\_email"; }

if (!preg\_match($name\_pat, $name) || !preg\_match($lastname\_pat, $lastname)) { $error = "name\_invalid"; }

$query = "SELECT COUNT(email) AS rowCount FROM users WHERE email = :email AND id != :id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$priprema->bindParam(":id", $id);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "email\_taken"; }

if ($error == "") {

$query = "UPDATE users SET email = :email, name = :name, last\_name = :lastname WHERE ID = :id";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":lastname", $lastname);

$priprema->bindParam(":id", $id);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/insert\_picture.php

<?php

include("../../config/connection.php");

$error = "";

if (isset($\_POST['admin\_insert\_picture\_submit'])) {

$alt = $\_POST['admin\_insert\_picture\_alt'];

$srb\_alt = $\_POST['admin\_insert\_picture\_srbalt'];

$post\_id = $\_POST['admin\_insert\_picture\_options'];

$image = $\_FILES['admin\_insert\_picture\_picture'];

$name\_pat = "/^[A-Z][a-z]+$/";

if ($alt == "" || $srb\_alt == "") { $error = "Fields are empty"; }

if (!preg\_match($name\_pat, $alt) || !preg\_match($name\_pat, $srb\_alt)) { $error = "Titles are not valid"; }

try {

$conn->beginTransaction();

$image\_name = $image["name"];

$image\_size = $image["size"];

$image\_tmp = $image["tmp\_name"];

$imageFileType = strtolower(pathinfo($image\_name,PATHINFO\_EXTENSION));

if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg" && $imageFileType != "gif" ) { $error = "Not valid image"; }

$alt = $alt . " 1";

$srb\_alt = $srb\_alt . " 1";

$upload\_dir = '../../assets/images/recipes/';

$src = 'assets/images/recipes/'.$image\_name;

$add\_picture = "INSERT INTO pictures (src, alt, SRBalt) VALUES (:src, :alt, :srb\_alt)";

$priprema\_pic = $conn -> prepare($add\_picture);

$priprema\_pic->bindParam(":src", $src);

$priprema\_pic->bindParam(":alt", $alt);

$priprema\_pic->bindParam(":srb\_alt", $srb\_alt);

$rezultat\_pic = $priprema\_pic->execute();

$last\_picture\_id = $conn->lastInsertId();

$image\_path = $upload\_dir."food\_".$last\_picture\_id.".".$imageFileType;

list($sirina, $visina) = getimagesize($image\_tmp);

$novaVisina = 200;

$novaSirina = ($novaVisina/$visina) \* $sirina;

$novaSlika = imagecreatetruecolor($novaSirina, $novaVisina);

$postojecaSlika = null;

switch ($imageFileType) {

case 'jpeg':

case 'jpg':

$postojecaSlika = imagecreatefromjpeg($image\_tmp);

break;

case 'png':

$postojecaSlika = imagecreatefrompng($image\_tmp);

break;

case 'gif':

$postojecaSlika = imagecreatefromgif($image\_tmp);

break;

}

imagecopyresampled($novaSlika, $postojecaSlika, 0, 0, 0, 0, $novaSirina, $novaVisina, $sirina, $visina);

switch ($imageFileType) {

case 'jpeg':

case 'jpg':

imagejpeg($novaSlika, $image\_path, 75);

break;

case 'png':

imagepng($novaSlika, $image\_path);

break;

case 'gif':

imagegif($novaSlika, $image\_path);

break;

}

$new\_src = 'assets/images/recipes/food\_'.$last\_picture\_id.".".$imageFileType;

$change\_pic = "UPDATE pictures SET src = :src WHERE ID = :id";

$priprema\_change = $conn -> prepare($change\_pic);

$priprema\_change->bindParam(":src", $new\_src);

$priprema\_change->bindParam(":id", $last\_picture\_id);

$rezultat\_change = $priprema\_change->execute();

if (!$rezultat\_change) { $error = "Error changing name in database"; }

$post\_picture\_query = "INSERT INTO post\_pictures (ID\_post, ID\_picture) VALUES (:post\_id, :picture\_id)";

$priprema\_pp = $conn -> prepare($post\_picture\_query);

$priprema\_pp->bindParam(":post\_id", $post\_id);

$priprema\_pp->bindParam(":picture\_id", $last\_picture\_id);

$rezultat\_pp = $priprema\_pp->execute();

if (!$rezultat\_pp) { $error = "Error adding to table"; }

if ($error != "") {

echo $error;

$conn->rollback();

} else {

$conn->commit();

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

die();

}

}catch (PDOException $e) {

echo $e->getMessage();

$conn->rollback();

}

} else {

$error = "Must send by form!";

echo $error;

}

?>



## models/admin/insert\_post\_tag.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$post = $\_POST["post"];

$tag = $\_POST["tag"];

$query = "SELECT COUNT(ID\_post) AS rowCount FROM post\_tag WHERE ID\_post = :post AND ID\_tag = :tag";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":post", $post);

$priprema->bindParam(":tag", $tag);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "post\_tag\_exists"; }

if ($error == "") {

$query = "INSERT INTO post\_tag (ID\_post, ID\_tag) VALUES (:post, :tag)";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":post", $post);

$priprema->bindParam(":tag", $tag);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 200;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/insert\_tag.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$name = $\_POST["name"];

$srb\_name = $\_POST["srb\_name"];

$name\_pat = "/^[A-Z][a-z]+$/";

if ($name == "" || $srb\_name == "") { $error = "empty\_field"; }

if (!preg\_match($name\_pat, $name) || !preg\_match($name\_pat, $srb\_name)) { $error = "tag\_name\_invalid"; }

$query = "SELECT COUNT(name) AS rowCount FROM tags WHERE name = :name OR SRBname = :srb\_name";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":srb\_name", $srb\_name);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "tag\_exists"; }

if ($error == "") {

$query = "INSERT INTO tags (name, SRBname) VALUES (:name, :srb\_name)";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":srb\_name", $srb\_name);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/insert\_user.php

<?php

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

include("../../config/connection.php");

$code = 500;

$error = "";

$email = $\_POST["email"];

$name = $\_POST["name"];

$lastname = $\_POST["lastname"];

$password = $\_POST["password"];

$is\_admin = $\_POST["is\_admin"];

$rights\_id = 2;

iF ($is\_admin == "true") { $rights\_id = 1; }

$email\_pattern = "^[a-zA-Z0-9.!#$%&’\*+/=?\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$^";

$name\_pat = "/^[A-Z][a-z]+$/";

$lastname\_pat = "/^([A-Z][a-z]+)+$/";

if ($email == "" || $name == "" || $lastname == "" || $password == "") { $error = "empty\_field"; }

if (strlen($password) < 6) { $error = "password\_short"; }

if (!preg\_match($email\_pattern, $email)) { $error = "not\_valid\_email"; }

if (!preg\_match($name\_pat, $name) || !preg\_match($lastname\_pat, $lastname)) { $error = "name\_invalid"; }

$query = "SELECT COUNT(email) AS rowCount FROM users WHERE email = :email";

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$rezultat = $priprema->execute();

$redovi = $priprema->FetchAll();

$redovi = $redovi[0]->rowCount;

if ($redovi == 1) { $error = "email\_taken"; }

if ($error == "") {

$query = "INSERT INTO users (email, password, rights\_id, name, last\_name) VALUES (:email, :password, :rights\_id, :name, :lastname)";

$password = md5($password);

$priprema = $conn -> prepare($query);

$priprema->bindParam(":email", $email);

$priprema->bindParam(":password", $password);

$priprema->bindParam(":name", $name);

$priprema->bindParam(":lastname", $lastname);

$priprema->bindParam(":rights\_id", $rights\_id);

$rezultat = $priprema->execute();

if ($rezultat) {

$code = 201;

echo json\_encode("Successful");

} else {

$code = 503;

echo json\_encode("database\_error");

}

} else {

$code = 503;

echo json\_encode($error);

}

http\_response\_code($code);

?>



## models/admin/newsletter.php

<?php

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

include("../../config/connection.php");

$query = "SELECT email, ID FROM newsletter\_subscribers";

$subscribers = executeQuery($query);

echo json\_encode($subscribers);

?>



## models/admin/pictures.php

<?php

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

include("../../config/connection.php");

$query = "SELECT src, ID, alt, SRBalt FROM pictures";

$pictures = executeQuery($query);

echo json\_encode($pictures);

?>



## models/admin/post\_tag.php

<?php

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

include("../../config/connection.php");

$query = "SELECT p.title AS postTitle, p.SRBtitle AS postSRBTitle, p.ID AS postID, t.name AS tagName, t.SRBname AS tagSRBName, t.ID AS tagID FROM post\_tag pt INNER JOIN post p ON pt.ID\_post = p.ID INNER JOIN tags t ON pt.ID\_tag = t.ID";

$posttags = executeQuery($query);

echo json\_encode($posttags);

?>



## models/admin/posts.php

<?php

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

include("../../config/connection.php");

$query = "SELECT title, SRBtitle, ID, text, SRBtext FROM post";

$posts = executeQuery($query);

echo json\_encode($posts);

?>



## models/admin/tags.php

<?php

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

include("../../config/connection.php");

$query = "SELECT name, SRBname, ID FROM tags";

$tags = executeQuery($query);

echo json\_encode($tags);

?>



## models/admin/users.php

<?php

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

include("../../config/connection.php");

$query = "SELECT ID, email, name, last\_name FROM users WHERE rights\_id != 1";

$users = executeQuery($query);

echo json\_encode($users);

?>



## assets/lang/eng.json

{

"page\_title": "25Hour Kitchen",

"header\_template": {

"uk\_flag": "UK Flag",

"serbian\_flag": "Serbian Flag"

},

"footer\_template": {

"navigation": "Navigation",

"recent\_posts": "Recent Posts",

"tags": "Tags"

},

"main\_template": {

"title": "Welcome to 25Hour Kitchen",

"baking\_image": "Baking Cookies"

},

"food\_posts": {

"by": "BY: ",

"comments": "Comments",

"continue\_reading": "Continue Reading",

"no\_posts": "Currently there are no posts by these parameters...",

"search": "Search...",

"how\_it\_started\_title": "How it started?",

"how\_it\_started\_text": "From a simple blog idea, this website has developed into something much more, a place for your favourite and best recipes..",

"subscribe": "Email",

"subscribe\_newsletter\_title": "Subscribe to our newsletter",

"subscribe\_newsletter\_text": "Subscribe and never miss a new recipe.",

"quote\_title": "Your Daily Quote:",

"no\_comment\_show": "No comments to show...",

"add\_comment\_title": "Add a new comment",

"add\_comment\_button": "Comment",

"empty\_comment": "Minimum comment size is 5 characters",

"log\_to\_comment": "Must login to comment",

"other\_error": "Unknown error happened",

"database\_error": "Database Error",

"subscribe\_empty\_field": "Email must be entered",

"not\_valid\_email": "Email not valid",

"subscription\_mail\_exists": "You have already subscribed"

},

"notification": {

"rss\_title": "RSS copied",

"rss\_text": "RSS was copied to ClipBoard",

"login\_title": "Login Successful",

"login\_text": "Login Successful, window will now reload",

"register\_title": "Register Successful",

"register\_text": "Register Successful, you can now login",

"logout\_title": "Logout Successful",

"logout\_text": "Logout Successful, window will now reload",

"comment\_title": "Commented successfully",

"comment\_text": "Commented successfully, window will now reload",

"insert\_user\_title": "User inserted successfully",

"insert\_user\_text": "User inserted successfully, you can now login as the new user",

"insert\_tag\_title": "Tag inserted successfully",

"insert\_tag\_text": "Tag inserted successfully, you can now use it",

"insert\_post\_tag\_title": "Connection inserted succesfully",

"insert\_post\_tag\_text": "Your post now has the new tag",

"update\_user\_title": "User edited successfully",

"update\_user\_text": "User edited successfully, you can now login",

"update\_post\_title": "Post edited successfully",

"update\_post\_text": "Post edited successfully",

"update\_tag\_title": "Tag edited successfully",

"update\_tag\_text": "Tag edited successfully",

"delete\_user\_title": "User deleted successfully",

"delete\_user\_text": "User deleted successfully",

"delete\_post\_title": "Post deleted successfully",

"delete\_post\_text": "Post deleted successfully",

"delete\_tag\_title": "Tag deleted successfully",

"delete\_tag\_text": "Tag deleted successfully",

"delete\_picture\_title": "Picture deleted successfully",

"delete\_picture\_text": "Picture deleted successfully",

"delete\_comment\_title": "Comment deleted successfully",

"delete\_comment\_text": "Comment deleted successfully",

"delete\_newsletter\_title": "Newsletter subscription deleted successfully",

"delete\_newsletter\_text": "Newsletter subscription deleted successfully",

"delete\_post\_tag\_title": "Relationship between post and tag deleted",

"delete\_post\_tag\_text": "Relationship between post and tag deleted",

"subscription\_title": "Subscription successfully",

"subscription\_text": "You have successfully subscribed to our news letter",

"contact\_mail\_sent\_title": "Contact mail sent",

"contact\_mail\_sent\_text": "We will contact you as soon as possible",

"word\_saved\_title": "Author text saved",

"word\_saved\_text": "Author text is saved to your desktop",

"excel\_saved\_title": "Posts saved",

"excel\_saved\_text": "Posts saved to your desktop",

"forgot\_password\_title": "Mail reset successful",

"forgot\_password\_text": "Mail reset successfully sent to admins"

},

"login\_template": {

"register\_email": "Email:",

"register\_email\_placeholder": "someone@something",

"register\_name": "Name:",

"register\_name\_placeholder": "Someone",

"register\_lastname": "Last Name:",

"register\_lastname\_placeholder": "Someone",

"register\_password": "Password:",

"register\_password\_repeat": "Repeat password:",

"register": "Register",

"login": "Login",

"forgot\_password": "Forgot password?",

"not\_valid\_email": "Email not valid",

"empty\_field": "None of the fields should be left empty",

"password\_short": "Password to short (min 6 characters)",

"wrong\_params": "Wrong parameters, try again",

"passwords\_not\_match": "Passwords do not match",

"name\_invalid": "Name and last name must be capitalized and contain only letters",

"other\_error": "Unknown error happened",

"database\_error": "Database Error",

"email\_taken": "Email you have entered is already taken",

"empty\_email": "Email field must not be empty"

},

"add\_post\_template": {

"welcome": "Welcome ",

"logout": "Logout",

"title": "Add a recipe",

"create\_button": "Create",

"add\_post\_title": "English Title",

"add\_post\_srb\_title": "Serbian Title",

"add\_post\_text": "English Text",

"add\_post\_srb\_text": "Serbian Text",

"add\_post\_type": "Types",

"add\_post\_image": "Images",

"add\_post\_tag": "Tags",

"empty\_field": "None of the fields should be left empty",

"type\_not\_selected": "You must select a type",

"title\_invalid": "Titles must be capitalized and contain only letters",

"tag\_unchecked": "You must check at least one tag",

"no\_image\_selected": "At least one image must be chosen"

},

"contact\_template": {

"title": "Contact",

"send\_mail": "someone@something",

"send\_contact": "Send",

"author": "Author",

"author\_name": "Name:",

"author\_birth": "Born:",

"author\_mail": "Email:",

"author\_number": "Telephone:",

"author\_work": "Works as:",

"download\_author\_data": "Download",

"empty\_field": "None of the fields should be left empty",

"mail\_invalid": "Email is in not valid format",

"database\_error": "Database Error",

"other\_error": "Unknown error happened"

},

"admin\_template": {

"title": "Admin Panel",

"visiting\_title": "Number of visitings by page in the last 24h",

"visiting\_main": "Number of main page views: ",

"visiting\_appetizer": "Number of appetizer views: ",

"visiting\_main\_course": "Number of main course views: ",

"visiting\_dessert": "Number of dessert views: ",

"visiting\_contact": "Number of contact page views: ",

"visiting\_admin": "Number of arrivals on admin page: ",

"visiting\_404": "Number of wrong pages: ",

"other\_controls\_title": "Other Facts:",

"number\_logged\_users": "Number of Logged In active users: ",

"export\_to\_excel": "Download posts as Excel",

"get\_documentation": "Get Documentation Here",

"admin\_options": {

"admin\_insert\_user": "Insert User",

"admin\_insert\_tag": "Insert Tag",

"admin\_insert\_picture": "Insert Picture",

"admin\_insert\_post\_tag": "Insert Post-Tag",

"admin\_edit\_user": "Edit User",

"admin\_edit\_post": "Edit Post",

"admin\_edit\_tag": "Edit Tag",

"admin\_delete\_user": "Delete User",

"admin\_delete\_post": "Delete Post",

"admin\_delete\_tag": "Delete Tag",

"admin\_delete\_picture": "Delete Picture",

"admin\_delete\_comment": "Delete Comment",

"admin\_delete\_newsletter": "Delete Newsletter Subscription",

"admin\_delete\_post\_tag": "Delete Post-Tag"

},

"admin\_controls": {

"admin\_insert\_user\_name": "Name",

"admin\_insert\_user\_lastname": "Last Name",

"admin\_insert\_user\_mail": "Email",

"admin\_insert\_user\_password": "Password",

"admin\_insert\_user\_admin": "Is Admin?",

"admin\_insert\_user\_submit": "Submit",

"admin\_insert\_tag\_name": "Tag Name",

"admin\_insert\_tag\_srbname": "Tag Name In Serbian",

"admin\_insert\_tag\_submit": "Submit",

"admin\_insert\_picture\_picture": "Picture",

"admin\_insert\_picture\_alt": "Description",

"admin\_insert\_picture\_srbalt": "Description In Serbian",

"admin\_insert\_picture\_options": "For Post",

"admin\_insert\_picture\_submit": "Submit",

"admin\_insert\_post\_tag\_post\_options": "Select Post",

"admin\_insert\_post\_tag\_tag\_options": "Select Tag",

"admin\_insert\_post\_tag\_tag\_submit": "Submit",

"admin\_edit\_user\_name": "Name",

"admin\_edit\_user\_lastname": "Last Name",

"admin\_edit\_user\_email": "Email",

"admin\_edit\_user\_submit": "Submit",

"admin\_edit\_post\_title": "Title",

"admin\_edit\_post\_srbtitle": "Title In Serbian",

"admin\_edit\_post\_text": "Text",

"admin\_edit\_post\_srbtext": "Text In Serbian",

"admin\_edit\_post\_submit": "Submit",

"admin\_edit\_tag\_name": "Title",

"admin\_edit\_tag\_srbname": "Title In Serbian",

"admin\_edit\_tag\_submit": "Submit",

"admin\_delete\_user\_submit": "Submit",

"admin\_delete\_post\_submit": "Submit",

"admin\_delete\_tag\_submit": "Submit",

"admin\_delete\_picture\_submit": "Submit",

"admin\_delete\_comment\_submit": "Submit",

"admin\_delete\_newsletter\_submit": "Submit",

"admin\_delete\_post\_tag\_submit": "Submit"

},

"admin\_errors": {

"empty\_field": "None of the fields should be left empty",

"password\_short": "Password to short (min 6 characters)",

"wrong\_params": "Wrong parameters, try again",

"name\_invalid": "Name and last name must be capitalized and contain only letters",

"other\_error": "Unknown error happened",

"database\_error": "Database Error",

"email\_taken": "Email you have entered is already taken",

"tag\_name\_invalid": "Tag name must be capitalized and contain only letters",

"tag\_exists": "Tag name already exists",

"image\_name\_invalid": "Image name must be capitalized and contain only letters",

"no\_image\_selected": "Image must be selected",

"post\_tag\_exists": "This post already has this tag",

"title\_name\_invalid": "Title name must be capitalized and contain only letters"

}

}

}



## assets/lang/srb.json

{

"page\_title": "25Sati Kuhinja",

"header\_template": {

"uk\_flag": "Zastava Ujedinjenog Kraljevstva",

"serbian\_flag": "Srpska Zastava"

},

"footer\_template": {

"navigation": "Navigacija",

"recent\_posts": "Najnovije Objave",

"tags": "Tagovi"

},

"main\_template": {

"title": "Dobrodošli na 25Sati Kuhinja",

"baking\_image": "Pečenje kolača"

},

"food\_posts": {

"by": "OD: ",

"comments": "Komentara",

"continue\_reading": "Nastavite čitati",

"no\_posts": "Trenutno nema objava po ovim parametrima...",

"search": "Pretraga...",

"how\_it\_started\_title": "Kako je sve počelo?",

"how\_it\_started\_text": "Od jedne jednostavne ideje o blogu, ovaj web sajt je izrastao u nešto mnogo više, mesto za Vaše omiljene i najbolje recepte..",

"subscribe": "Email",

"subscribe\_newsletter\_title": "Pretplatite se na naš bilten",

"subscribe\_newsletter\_text": "Pretplatite se i nikad ne propustite nov recept.",

"quote\_title": "Vaš Dnevni Citat:",

"no\_comment\_show": "Nema komentara za prikaz...",

"add\_comment\_title": "Dodajte komentar",

"add\_comment\_button": "Komentarišite",

"empty\_comment": "Minimalna dužina komentara je 5 karaktera",

"log\_to\_comment": "Morate biti prijavljeni za komentarisanje",

"other\_error": "Desila se nepoznata greška",

"database\_error": "Greška sa bazom podataka",

"subscribe\_empty\_field": "Mora se uneti email",

"not\_valid\_email": "Email nije ispravan",

"subscription\_mail\_exists": "Već ste se prijavili"

},

"notification": {

"rss\_title": "RSS kopiran",

"rss\_text": "RSS je kopiran na Vašu privremenu memoriju",

"login\_title": "Prijavljivanje uspešno",

"login\_text": "Uspešno ste se prijavili, prozor će se osvežiti",

"register\_title": "Upisivanje uspešno",

"register\_text": "Upisivanje uspešno, sada se možete prijaviti",

"logout\_title": "Odjava uspešna",

"logout\_text": "Uspešno ste se odjavili, prozor će se osvežiti",

"comment\_title": "Komentarisanje uspešno",

"comment\_text": "Uspešno ste komentarisali, prozor će se osvežiti",

"insert\_user\_title": "Korisnik napravljen uspešno",

"insert\_user\_text": "Korisnik napravljen uspešno, sada se možete prijaviti kao nov korisnik",

"insert\_tag\_title": "Tag napravljen uspešno",

"insert\_tag\_text": "Tag napravljen uspešno, sada ga možete koristiti",

"insert\_post\_tag\_title": "Veza uspešno dodata",

"insert\_post\_tag\_text": "Vaša objava sadrži nov tag",

"update\_user\_title": "Korisnik promenjen uspešno",

"update\_user\_text": "Korisnik promenjen uspešno, sada se možete prijaviti",

"update\_post\_title": "Objava promenjena uspešno",

"update\_post\_text": "Objava promenjena uspešno",

"update\_tag\_title": "Tag promenjen uspešno",

"update\_tag\_text": "Tag promenjen uspešno",

"delete\_user\_title": "Korisnik obrisan uspešno",

"delete\_user\_text": "Korisnik obrisan uspešno",

"delete\_post\_title": "Objava obrisana uspešno",

"delete\_post\_text": "Objava obrisana uspešno",

"delete\_tag\_title": "Tag obrisan uspešno",

"delete\_tag\_text": "Tag obrisan uspešno",

"delete\_picture\_title": "Slika obrisana uspešno",

"delete\_picture\_text": "Slika obrisana uspešno",

"delete\_comment\_title": "Komentar obrisan uspešno",

"delete\_comment\_text": "Komentar obrisan uspešno",

"delete\_newsletter\_title": "Pretplata na bilten obrisana uspešno",

"delete\_newsletter\_text": "Pretplata na bilten obrisana uspešno",

"delete\_post\_tag\_title": "Veza između objave i taga obrisana",

"delete\_post\_tag\_text": "Veza između objave i taga obrisana",

"subscription\_title": "Pretplata uspešna",

"subscription\_text": "Uspešno ste se prijavili na naš bilten",

"contact\_mail\_sent\_title": "Kontakt poruka poslata",

"contact\_mail\_sent\_text": "Kontaktiraćemo vaš u najkraćem mogućem roku",

"word\_saved\_title": "Tekst o autoru sačuvan",

"word\_saved\_text": "Tekst o autoru je sačuvan na Vašem desktop-u",

"excel\_saved\_title": "Objave sačuvane",

"excel\_saved\_text": "Objave sačuvane na Vašem desktop-u",

"forgot\_password\_title": "Resetovanje mail-a uspešno",

"forgot\_password\_text": "Uspešno ste kontaktirali admine radi resetovanja email-a"

},

"login\_template": {

"register\_email": "Email:",

"register\_email\_placeholder": "neko@nesto",

"register\_name": "Ime:",

"register\_name\_placeholder": "Neko",

"register\_lastname": "Prezime:",

"register\_lastname\_placeholder": "Neko",

"register\_password": "Šifra:",

"register\_password\_repeat": "Ponovite šifru:",

"register": "Upisivanje",

"login": "Prijaviti se",

"forgot\_password": "Zaboravio šifru?",

"not\_valid\_email": "Email nije u ispravnom formatu",

"empty\_field": "Nijedno od polja ne sme biti prazno",

"password\_short": "Šifra je previše kratka (min 6 karaktera)",

"wrong\_params": "Pogrešni parametri, pobajte ponovo",

"passwords\_not\_match": "Šifre se ne podudaraju",

"name\_invalid": "Ime i prezime moraju počinjati velikim slovom i da sadrže samo slova",

"other\_error": "Desila se nepoznata greška",

"database\_error": "Greška sa bazom podataka",

"email\_taken": "Email koji ste uneli je zauzet",

"empty\_email": "Polje za email ne sme biti prazno"

},

"add\_post\_template": {

"welcome": "Dobrodošli ",

"logout": "Odjavite se",

"title": "Dodajte recept",

"create\_button": "Kreirajte",

"add\_post\_title": "Engleski Naslov",

"add\_post\_srb\_title": "Srpski Naslov",

"add\_post\_text": "Engleski Tekst",

"add\_post\_srb\_text": "Srpski Tekst",

"add\_post\_type": "Tipovi",

"add\_post\_image": "Slike",

"add\_post\_tag": "Tagovi",

"empty\_field": "Nijedno od polja ne sme biti prazno",

"type\_not\_selected": "Morate odabrati tip",

"title\_invalid": "Naslovi moraju počinjati velikim slovom i da sadrže samo slova",

"tag\_unchecked": "Barem jedan tag mora biti odabran",

"no\_image\_selected": "Mora se odabrati barem jedna slika"

},

"contact\_template": {

"title": "Kontakt",

"send\_mail": "neko@nesto",

"send\_contact": "Pošaljite",

"author": "Autor",

"author\_name": "Ime:",

"author\_birth": "Rođen:",

"author\_mail": "Email:",

"author\_number": "Telefon:",

"author\_work": "Radi kao:",

"download\_author\_data": "Preuzmite",

"empty\_field": "Nijedno od polja ne sme biti prazno",

"mail\_invalid": "Email nije u ispravnom formatu",

"database\_error": "Greška sa bazom podataka",

"other\_error": "Desila se nepoznata greška"

},

"admin\_template": {

"title": "Admin Panel",

"visiting\_title": "Broj pregleda po strani u poslednjih 24h",

"visiting\_main": "Broj pregleda početne stranice: ",

"visiting\_appetizer": "Broj pregleda predjela: ",

"visiting\_main\_course": "Broj pregleda glavnih jela: ",

"visiting\_dessert": "Broj pregleda dezerata: ",

"visiting\_contact": "Broj pregleda kontakt stranice: ",

"visiting\_admin": "Broj dolazaka na admin stranicu: ",

"visiting\_404": "Broj pogrešnih stranica: ",

"other\_controls\_title": "Ostali podaci:",

"number\_logged\_users": "Broj prijavljenih aktivnih korisnika: ",

"export\_to\_excel": "Preuzmite objave kao Excel",

"get\_documentation": "Dohvatite Dokumentaciju Ovde",

"admin\_options": {

"admin\_insert\_user": "Dodajte Korisnika",

"admin\_insert\_tag": "Dodajte Tag",

"admin\_insert\_picture": "Dodajte Sliku",

"admin\_insert\_post\_tag": "Dodajte Objavu-Tag",

"admin\_edit\_user": "Promenite Korisnika",

"admin\_edit\_post": "Promenite Objavu",

"admin\_edit\_tag": "Promenite Tag",

"admin\_delete\_user": "Obrišite Korisnika",

"admin\_delete\_post": "Obrišite Objavu",

"admin\_delete\_tag": "Obrišite Tag",

"admin\_delete\_picture": "Obrišite Sliku",

"admin\_delete\_comment": "Obrišite Komentar",

"admin\_delete\_newsletter": "Obrišite Pretplatu",

"admin\_delete\_post\_tag": "Obrišite Objavu-Tag"

},

"admin\_controls": {

"admin\_insert\_user\_name": "Ime",

"admin\_insert\_user\_lastname": "Prezime",

"admin\_insert\_user\_mail": "Email",

"admin\_insert\_user\_password": "Šifra",

"admin\_insert\_user\_admin": "Da li je admin?",

"admin\_insert\_user\_submit": "Pošaljite",

"admin\_insert\_tag\_name": "Ime Taga",

"admin\_insert\_tag\_srbname": "Ime Taga Na Srpskom",

"admin\_insert\_tag\_submit": "Pošaljite",

"admin\_insert\_picture\_picture": "Slika",

"admin\_insert\_picture\_alt": "Opis",

"admin\_insert\_picture\_srbalt": "Opis Na Srpskom",

"admin\_insert\_picture\_options": "Za Objavu",

"admin\_insert\_picture\_submit": "Pošaljite",

"admin\_insert\_post\_tag\_post\_options": "Izaberite Objavu",

"admin\_insert\_post\_tag\_tag\_options": "Izaberite Tag",

"admin\_insert\_post\_tag\_tag\_submit": "Pošaljite",

"admin\_edit\_user\_name": "Ime",

"admin\_edit\_user\_lastname": "Prezime",

"admin\_edit\_user\_email": "Email",

"admin\_edit\_user\_submit": "Pošaljite",

"admin\_edit\_post\_title": "Naslov",

"admin\_edit\_post\_srbtitle": "Naslov Na Srpskom",

"admin\_edit\_post\_text": "Tekst",

"admin\_edit\_post\_srbtext": "Tekst Na Srpskom",

"admin\_edit\_post\_submit": "Pošaljite",

"admin\_edit\_tag\_name": "Naslov",

"admin\_edit\_tag\_srbname": "Naslov Na Srpskom",

"admin\_edit\_tag\_submit": "Pošaljite",

"admin\_delete\_user\_submit": "Pošaljite",

"admin\_delete\_post\_submit": "Pošaljite",

"admin\_delete\_tag\_submit": "Pošaljite",

"admin\_delete\_picture\_submit": "Pošaljite",

"admin\_delete\_comment\_submit": "Pošaljite",

"admin\_delete\_newsletter\_submit": "Pošaljite",

"admin\_delete\_post\_tag\_submit": "Pošaljite"

},

"admin\_errors": {

"not\_valid\_email": "Email nije u ispravnom formatu",

"empty\_field": "Nijedno od polja ne sme biti prazno",

"password\_short": "Šifra je previše kratka (min 6 karaktera)",

"wrong\_params": "Pogrešni parametri, pobajte ponovo",

"name\_invalid": "Ime i prezime moraju počinjati velikim slovom i da sadrže samo slova",

"other\_error": "Desila se nepoznata greška",

"database\_error": "Greška sa bazom podataka",

"email\_taken": "Email koji ste uneli je zauzet",

"tag\_name\_invalid": "Ime taga mora počinjati velikim slovom i da sadrži samo slova",

"tag\_exists": "Ime taga već postoji",

"image\_name\_invalid": "Ime slike mora počinjati velikim slovom i da sadrži samo slova",

"no\_image\_selected": "Mora se odabrati slika",

"post\_tag\_exists": "Ova objava već sadrži ovaj tag",

"title\_name\_invalid": "Ime naslova mora počinjati velikim slovom i da sadrži samo slova"

}

}

}



## assets/js/add\_post\_template.js

function start\_add\_post() {

$(".welcome\_user span:first-child").html(language\_content.add\_post\_template.welcome);

$(".welcome\_user button").html(language\_content.add\_post\_template.logout);

$(".add\_post\_form h2").html(language\_content.add\_post\_template.title);

$(".add\_post\_form #add\_post\_button").val(language\_content.add\_post\_template.create\_button);

$(".add\_post\_param label[for='add\_post\_title']").html(language\_content.add\_post\_template.add\_post\_title);

$(".add\_post\_param #add\_post\_title").attr("placeholder", language\_content.add\_post\_template.add\_post\_title);

$(".add\_post\_param label[for='add\_post\_srb\_title']").html(language\_content.add\_post\_template.add\_post\_srb\_title);

$(".add\_post\_param #add\_post\_srb\_title").attr("placeholder", language\_content.add\_post\_template.add\_post\_srb\_title);

$(".add\_post\_param label[for='add\_post\_text']").html(language\_content.add\_post\_template.add\_post\_text);

$(".add\_post\_param label[for='add\_post\_srb\_text']").html(language\_content.add\_post\_template.add\_post\_srb\_text);

$(".add\_post\_param label[for='add\_post\_type']").html(language\_content.add\_post\_template.add\_post\_type);

$(".add\_post\_param label[for='add\_post\_image']").html(language\_content.add\_post\_template.add\_post\_image);

$(".add\_post\_param label[for='add\_post\_tag']").html(language\_content.add\_post\_template.add\_post\_tag);

post\_get\_types();

post\_get\_tags();

}

function post\_get\_types() {

$.ajax({

url: "models/types.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (t) {

text += `<div><input type="radio" name="add\_post\_type" value="${t.ID}"/> ${lang=="eng"?t.text:(lang=="srb"?t.SRBtext:"Error")}</div>`;

});

$(".add\_post\_param .add\_post\_type\_holder").html(text);

},

error: function (err) {

write\_error\_to\_file(err);

$(".add\_post\_param .add\_post\_type\_holder").html(err.responseText);

}

});

}

function post\_get\_tags() {

$.ajax({

url: "models/tags.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (t) {

text += `<div><input type="checkbox" name="add\_post\_tag[]" value="${t.ID}"/> ${lang=="eng"?t.name:(lang=="srb"?t.SRBname:"Error")}</div>`;

});

$(".add\_post\_param .add\_post\_tag\_holder").html(text);

},

error: function (err) {

write\_error\_to\_file(err);

$(".add\_post\_param .add\_post\_tag\_holder").html(err.responseText);

}

});

}

function add\_post\_param\_check() {

var eng\_title = document.getElementById("add\_post\_title").value,

srb\_title = document.getElementById("add\_post\_srb\_title").value,

eng\_text = document.getElementById("add\_post\_text").value,

srb\_text = document.getElementById("add\_post\_srb\_text").value,

title\_pattern = /^[A-ZŠĐŽČĆ][a-z0-9ščćđž]+$/,

type\_checked = $(".add\_post\_type\_holder input:checked").val(),

tags\_checked = document.getElementsByName("add\_post\_tag[]"),

has\_checked = false,

image\_selected = document.getElementsByName("add\_post\_image[]").length;

if (eng\_title == "" || srb\_title == "" || eng\_text == "" || srb\_text == "") {

document.getElementById("add\_post\_error").innerHTML = language\_content.add\_post\_template.empty\_field;

return false;

}

if (!type\_checked) {

document.getElementById("add\_post\_error").innerHTML = language\_content.add\_post\_template.type\_not\_selected;

return false;

}

if (!title\_pattern.test(eng\_title) || !title\_pattern.test(srb\_title)) {

document.getElementById("add\_post\_error").innerHTML = language\_content.add\_post\_template.title\_invalid;

return false;

}

tags\_checked.forEach(function (t) {

if (t.checked) { has\_checked = true; }

});

if (!has\_checked) {

document.getElementById("add\_post\_error").innerHTML = language\_content.add\_post\_template.tag\_unchecked;

return false;

}

if (image\_selected <= 0) {

document.getElementById("add\_post\_error").innerHTML = language\_content.add\_post\_template.no\_image\_selected;

return false;

}

document.getElementById("add\_post\_error").innerHTML = "";

return true;

}



## assets/js/admin\_template.js

function start\_admin() {

if (!$("#admin\_panel:visible").length) { setTimeout(start\_admin, 50); } else { load\_admin\_data(); }

}

function load\_admin\_data() {

$("#admin\_panel h2").html(language\_content.admin\_template.title);

$(".admin\_controls.file\_controls h3").html(language\_content.admin\_template.visiting\_title);

$(".admin\_controls .visiting\_main span:first-child").html(language\_content.admin\_template.visiting\_main);

$(".admin\_controls .visiting\_appetizer span:first-child").html(language\_content.admin\_template.visiting\_appetizer);

$(".admin\_controls .visiting\_main\_course span:first-child").html(language\_content.admin\_template.visiting\_main\_course);

$(".admin\_controls .visiting\_dessert span:first-child").html(language\_content.admin\_template.visiting\_dessert);

$(".admin\_controls .visiting\_contact span:first-child").html(language\_content.admin\_template.visiting\_contact);

$(".admin\_controls .visiting\_admin span:first-child").html(language\_content.admin\_template.visiting\_admin);

$(".admin\_controls .visiting\_404 span:first-child").html(language\_content.admin\_template.visiting\_404);

$(".admin\_controls.other\_controls h3").html(language\_content.admin\_template.other\_controls\_title);

$(".admin\_controls.other\_controls #export\_to\_excel").html(language\_content.admin\_template.export\_to\_excel);

$(".admin\_controls.other\_controls a").html(language\_content.admin\_template.get\_documentation);

$(".admin\_controls .number\_logged\_users span:first-child").html(language\_content.admin\_template.number\_logged\_users);

var text = "",

el,

first\_option = "";

for (el in language\_content.admin\_template.admin\_options) {

if (!first\_option) { first\_option = el; }

text += `<option value="${el}">${language\_content.admin\_template.admin\_options[el]}</option>`;

}

$("#admin\_panel #admin\_option").html(text);

$('#admin\_panel #admin\_option').on('change', function() {

fill\_admin\_controls($(this).val());

});

$("#export\_to\_excel").on("click", export\_to\_excel);

fill\_admin\_controls(first\_option);

get\_page\_access\_info();

get\_logged\_users();

}

function fill\_admin\_controls(option) {

switch(option) {

case "admin\_insert\_user":

$("#admin\_extra\_option").css("display", "none");

admin\_insert\_user\_show();

break;

case "admin\_insert\_tag":

$("#admin\_extra\_option").css("display", "none");

admin\_insert\_tag\_show();

break;

case "admin\_insert\_picture":

$("#admin\_extra\_option").css("display", "none");

admin\_insert\_picture\_show();

break;

case "admin\_insert\_post\_tag":

$("#admin\_extra\_option").css("display", "none");

admin\_insert\_post\_tag\_show();

break;

case "admin\_edit\_user":

$("#admin\_extra\_option").css("display", "block");

admin\_edit\_user\_show();

break;

case "admin\_edit\_post":

$("#admin\_extra\_option").css("display", "block");

admin\_edit\_post\_show();

break;

case "admin\_edit\_tag":

$("#admin\_extra\_option").css("display", "block");

admin\_edit\_tag\_show();

break;

case "admin\_delete\_user":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_user\_show();

break;

case "admin\_delete\_post":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_post\_show();

break;

case "admin\_delete\_tag":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_tag\_show();

break;

case "admin\_delete\_picture":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_picture\_show();

break;

case "admin\_delete\_comment":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_comment\_show();

break;

case "admin\_delete\_newsletter":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_newsletter\_show();

break;

case "admin\_delete\_post\_tag":

$("#admin\_extra\_option").css("display", "block");

admin\_delete\_post\_tag\_show();

break;

}

}

function admin\_insert\_user\_show() {

var text = `<div class="admin\_option\_holder"><label for="admin\_insert\_user\_name">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_name}</label><input type="text" name="admin\_insert\_user\_name" id="admin\_insert\_user\_name" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_name}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_user\_lastname">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_lastname}</label><input type="text" name="admin\_insert\_user\_lastname" id="admin\_insert\_user\_lastname" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_lastname}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_user\_mail">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_mail}</label><input type="email" name="admin\_insert\_user\_mail" id="admin\_insert\_user\_mail" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_mail}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_user\_password">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_password}</label><input type="password" name="admin\_insert\_user\_password" id="admin\_insert\_user\_password" placeholder="\* \* \* \* \* \*"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_user\_admin">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_admin}</label><input type="checkbox" id="admin\_insert\_user\_admin" name="admin\_insert\_user\_admin"/></div>`;

text += `<p id="admin\_insert\_user\_error"></p>`;

text += `<button id="admin\_insert\_user\_submit">${language\_content.admin\_template.admin\_controls.admin\_insert\_user\_submit}</button>`;

$("#admin\_extra\_space").html(text);

document.getElementById("admin\_insert\_user\_name").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_user\_send(); } });

document.getElementById("admin\_insert\_user\_lastname").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_user\_send(); } });

document.getElementById("admin\_insert\_user\_mail").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_user\_send(); } });

document.getElementById("admin\_insert\_user\_password").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_user\_send(); } });

$("#admin\_insert\_user\_submit").on("click", admin\_insert\_user\_send);

}

function admin\_insert\_user\_send() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

name\_pat = /^[A-Z][a-z]+$/,

lastname\_pat = /^([A-Z][a-z]+)+$/,

email = document.getElementById("admin\_insert\_user\_mail").value,

name = document.getElementById("admin\_insert\_user\_name").value,

lastname = document.getElementById("admin\_insert\_user\_lastname").value,

password = document.getElementById("admin\_insert\_user\_password").value,

is\_admin = document.getElementById("admin\_insert\_user\_admin").checked;

if (email == "" || name == "" || lastname == "" || password == "") {

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

return;

}

if (password.length < 6) {

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.password\_short;

return;

}

if (!email\_pat.test(email)) {

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.not\_valid\_email;

return;

}

if (!name\_pat.test(name) || !lastname\_pat.test(lastname)) {

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.name\_invalid;

return;

}

document.getElementById("admin\_insert\_user\_error").innerHTML = "";

$.ajax({

url: "models/admin/insert\_user.php",

method: "POST",

dataType: "json",

data: {

email: email,

name: name,

lastname: lastname,

password: password,

is\_admin: is\_admin

},

success: function (data) {

if (data == "Successful") {

show\_notification("insert\_user");

document.getElementById("admin\_insert\_user\_mail").value = "";

document.getElementById("admin\_insert\_user\_name").value = "";

document.getElementById("admin\_insert\_user\_lastname").value = "";

document.getElementById("admin\_insert\_user\_password").value = "";

document.getElementById("admin\_insert\_user\_admin").checked = false;

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "empty\_field":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

break;

case "name\_invalid":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.name\_invalid;

break;

case "email\_taken":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.email\_taken;

break;

case "not\_valid\_email":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.not\_valid\_email;

break;

case "password\_short":

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.password\_short;

break;

default:

document.getElementById("admin\_insert\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_insert\_tag\_show() {

var text = `<div class="admin\_option\_holder"><label for="admin\_insert\_tag\_name">${language\_content.admin\_template.admin\_controls.admin\_insert\_tag\_name}</label><input type="text" name="admin\_insert\_tag\_name" id="admin\_insert\_tag\_name" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_tag\_name}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_tag\_srbname">${language\_content.admin\_template.admin\_controls.admin\_insert\_tag\_srbname}</label><input type="text" name="admin\_insert\_tag\_srbname" id="admin\_insert\_tag\_srbname" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_tag\_srbname}"/></div>`;

text += `<p id="admin\_insert\_tag\_error"></p>`;

text += `<button id="admin\_insert\_tag\_submit">${language\_content.admin\_template.admin\_controls.admin\_insert\_tag\_submit}</button>`;

$("#admin\_extra\_space").html(text);

document.getElementById("admin\_insert\_tag\_name").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_tag\_send(); } });

document.getElementById("admin\_insert\_tag\_srbname").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_insert\_tag\_send(); } });

$("#admin\_insert\_tag\_submit").on("click", admin\_insert\_tag\_send);

}

function admin\_insert\_tag\_send() {

var name\_pat = /^[A-Z][a-z]+$/,

name = document.getElementById("admin\_insert\_tag\_name").value,

srb\_name = document.getElementById("admin\_insert\_tag\_srbname").value;

if (name == "" || srb\_name == "") {

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

return;

}

if (!name\_pat.test(name) || !name\_pat.test(srb\_name)) {

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_name\_invalid;

return;

}

document.getElementById("admin\_insert\_tag\_error").innerHTML = "";

$.ajax({

url: "models/admin/insert\_tag.php",

method: "POST",

dataType: "json",

data: {

name: name,

srb\_name: srb\_name

},

success: function (data) {

if (data == "Successful") {

show\_notification("insert\_tag");

document.getElementById("admin\_insert\_tag\_name").value = "";

document.getElementById("admin\_insert\_tag\_srbname").value = "";

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "empty\_field":

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

break;

case "tag\_name\_invalid":

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_name\_invalid;

break;

case "tag\_exists":

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_exists;

break;

default:

document.getElementById("admin\_insert\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_insert\_picture\_show() {

$.ajax({

url: "models/admin/posts.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = `<form onsubmit="return admin\_insert\_picture\_check()" method="POST" action="models/admin/insert\_picture.php" name="admin\_insert\_picture\_form" enctype="multipart/form-data"><div class="admin\_option\_holder"><label for="admin\_insert\_picture\_picture">${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_picture}</label><input type="file" name="admin\_insert\_picture\_picture" id="admin\_insert\_picture\_picture"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_picture\_alt">${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_alt}</label><input type="text" name="admin\_insert\_picture\_alt" id="admin\_insert\_picture\_alt" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_alt}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_picture\_srbalt">${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_srbalt}</label><input type="text" name="admin\_insert\_picture\_srbalt" id="admin\_insert\_picture\_srbalt" placeholder="${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_srbalt}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_picture\_options">${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_options}</label><select name="admin\_insert\_picture\_options" id="admin\_insert\_picture\_options">`;

data.forEach(function (p) {

text += `<option value="${p.ID}">${lang=="eng"?p.title:(lang=="srb"?p.SRBtitle:"Error")}</option>`;

});

text += `</select></div>`;

text += `<p id="admin\_insert\_picture\_error"></p>`;

text += `<button id="admin\_insert\_picture\_submit" name="admin\_insert\_picture\_submit" type="submit">${language\_content.admin\_template.admin\_controls.admin\_insert\_picture\_submit}</button>`;

text += "</form>";

$("#admin\_extra\_space").html(text);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_insert\_picture\_check() {

var name\_pat = /^[A-Z][a-z]+$/,

alt = document.getElementById("admin\_insert\_picture\_alt").value,

srb\_alt = document.getElementById("admin\_insert\_picture\_srbalt").value,

image\_selected = document.getElementsByName("admin\_insert\_picture\_picture").length;

if (alt == "" || srb\_alt == "") {

document.getElementById("admin\_insert\_picture\_error").innerHTML = language\_content.admin\_template.admin\_controls.empty\_field;

return false;

}

if (!name\_pat.test(alt) || !name\_pat.test(srb\_alt)) {

document.getElementById("admin\_insert\_picture\_error").innerHTML = language\_content.admin\_template.admin\_controls.image\_name\_invalid;

return false;

}

if (image\_selected <= 0) {

document.getElementById("admin\_insert\_picture\_error").innerHTML = language\_content.admin\_template.admin\_controls.no\_image\_selected;

return false;

}

document.getElementById("admin\_insert\_picture\_error").innerHTML = "";

return true;

}

function admin\_insert\_post\_tag\_show() {

var posts,

tags;

$.ajax({

url: "models/admin/posts.php",

method: "GET",

dataType: "json",

success: function (data) {

posts = data;

$.ajax({

url: "models/admin/tags.php",

method: "GET",

dataType: "json",

success: function (data) {

tags = data;

var text = `<div class="admin\_option\_holder"><label for="admin\_insert\_post\_tag\_post\_options">${language\_content.admin\_template.admin\_controls.admin\_insert\_post\_tag\_post\_options}</label><select name="admin\_insert\_post\_tag\_post\_options" id="admin\_insert\_post\_tag\_post\_options">`;

posts.forEach(function (p) {

text += `<option value="${p.ID}">${lang=="eng"?p.title:(lang=="srb"?p.SRBtitle:"Error")}</option>`;

});

text += `</select></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_insert\_post\_tag\_tag\_options">${language\_content.admin\_template.admin\_controls.admin\_insert\_post\_tag\_tag\_options}</label><select name="admin\_insert\_post\_tag\_tag\_options" id="admin\_insert\_post\_tag\_tag\_options">`;

tags.forEach(function (t) {

text += `<option value="${t.ID}">${lang=="eng"?t.name:(lang=="srb"?t.SRBname:"Error")}</option>`;

});

text += `</select></div>`;

text += `<p id="admin\_insert\_post\_tag\_tag\_error"></p>`;

text += `<button id="admin\_insert\_post\_tag\_tag\_submit">${language\_content.admin\_template.admin\_controls.admin\_insert\_post\_tag\_tag\_submit}</button>`;

$("#admin\_extra\_space").html(text);

$("#admin\_insert\_post\_tag\_tag\_submit").on("click", admin\_insert\_post\_tag\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_insert\_post\_tag\_send() {

var post = document.getElementById("admin\_insert\_post\_tag\_post\_options").options[document.getElementById("admin\_insert\_post\_tag\_post\_options").selectedIndex].value,

tag = document.getElementById("admin\_insert\_post\_tag\_tag\_options").options[document.getElementById("admin\_insert\_post\_tag\_tag\_options").selectedIndex].value;

document.getElementById("admin\_insert\_post\_tag\_tag\_error").innerHTML = "";

$.ajax({

url: "models/admin/insert\_post\_tag.php",

method: "POST",

dataType: "json",

data: {

post: post,

tag: tag

},

success: function (data) {

if (data == "Successful") {

show\_notification("insert\_post\_tag");

}

},

error: function (err) {

write\_error\_to\_file(err);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_insert\_post\_tag\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "post\_tag\_exists":

document.getElementById("admin\_insert\_post\_tag\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.post\_tag\_exists;

break;

default:

document.getElementById("admin\_insert\_post\_tag\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_edit\_user\_show() {

var users;

$.ajax({

url: "models/admin/users.php",

method: "GET",

dataType: "json",

success: function (data) {

users = data;

var text = "";

users.forEach(function (u) {

text += `<option value="${u.ID}">${u.name} ${u.last\_name}</option>`;

});

$("#admin\_extra\_option").html(text);

admin\_edit\_user\_show\_second(users, users[0].ID);

$("#admin\_extra\_option").on('change', function() {

admin\_edit\_user\_show\_second(users, $(this).val());

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_edit\_user\_show\_second(users, user\_id) {

users.forEach(function (u) {

if (u.ID != user\_id) { return; }

var text = `<div class="admin\_option\_holder"><label for="admin\_edit\_user\_name">${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_name}</label><input type="text" name="admin\_edit\_user\_name" id="admin\_edit\_user\_name" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_name}" value="${u.name}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_user\_lastname">${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_lastname}</label><input type="text" name="admin\_edit\_user\_lastname" id="admin\_edit\_user\_lastname" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_lastname}" value="${u.last\_name}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_user\_email">${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_email}</label><input type="email" name="admin\_edit\_user\_email" id="admin\_edit\_user\_email" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_email}" value="${u.email}"/></div>`;

text += `<input type="hidden" name="admin\_edit\_user\_id" id="admin\_edit\_user\_id" value="${u.ID}"/></div>`;

text += `<p id="admin\_edit\_user\_error"></p>`;

text += `<button id="admin\_edit\_user\_submit">${language\_content.admin\_template.admin\_controls.admin\_edit\_user\_submit}</button>`;

$("#admin\_extra\_space").html(text);

});

document.getElementById("admin\_edit\_user\_name").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_user\_send(); } });

document.getElementById("admin\_edit\_user\_lastname").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_user\_send(); } });

document.getElementById("admin\_edit\_user\_email").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_user\_send(); } });

$("#admin\_edit\_user\_submit").on("click", admin\_edit\_user\_send);

}

function admin\_edit\_user\_send() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

name\_pat = /^[A-Z][a-z]+$/,

lastname\_pat = /^([A-Z][a-z]+)+$/,

email = document.getElementById("admin\_edit\_user\_email").value,

name = document.getElementById("admin\_edit\_user\_name").value,

lastname = document.getElementById("admin\_edit\_user\_lastname").value,

id = document.getElementById("admin\_edit\_user\_id").value;

if (email == "" || name == "" || lastname == "") {

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

return;

}

if (!email\_pat.test(email)) {

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.not\_valid\_email;

return;

}

if (!name\_pat.test(name) || !lastname\_pat.test(lastname)) {

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.name\_invalid;

return;

}

document.getElementById("admin\_edit\_user\_error").innerHTML = "";

$.ajax({

url: "models/admin/edit\_user.php",

method: "POST",

dataType: "json",

data: {

email: email,

name: name,

lastname: lastname,

id: id

},

success: function (data) {

if (data == "Successful") {

show\_notification("update\_user");

document.getElementById("admin\_edit\_user\_email").value = "";

document.getElementById("admin\_edit\_user\_name").value = "";

document.getElementById("admin\_edit\_user\_lastname").value = "";

fill\_admin\_controls("admin\_edit\_user");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "empty\_field":

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

break;

case "name\_invalid":

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.name\_invalid;

break;

case "email\_taken":

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.email\_taken;

break;

case "not\_valid\_email":

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.not\_valid\_email;

break;

default:

document.getElementById("admin\_edit\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_edit\_post\_show() {

var posts;

$.ajax({

url: "models/admin/posts.php",

method: "GET",

dataType: "json",

success: function (data) {

posts = data;

var text = "";

posts.forEach(function (p) {

text += `<option value="${p.ID}">${lang=="eng"?p.title:(lang=="srb"?p.SRBtitle:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

admin\_edit\_post\_show\_second(posts, posts[0].ID);

$("#admin\_extra\_option").on('change', function() {

admin\_edit\_post\_show\_second(posts, $(this).val());

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_edit\_post\_show\_second(posts, post\_id) {

posts.forEach(function (p) {

if (p.ID != post\_id) { return; }

var text = `<div class="admin\_option\_holder"><label for="admin\_edit\_post\_title">${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_title}</label><input type="text" name="admin\_edit\_post\_title" id="admin\_edit\_post\_title" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_title}" value="${p.title}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_post\_srbtitle">${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_srbtitle}</label><input type="text" name="admin\_edit\_post\_srbtitle" id="admin\_edit\_post\_srbtitle" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_srbtitle}" value="${p.SRBtitle}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_post\_text">${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_text}</label><textarea rows="8" cols="40" name="admin\_edit\_post\_text" id="admin\_edit\_post\_text">${p.text}</textarea></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_post\_srbtext">${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_srbtext}</label><textarea rows="8" cols="40" name="admin\_edit\_post\_srbtext" id="admin\_edit\_post\_srbtext">${p.SRBtext}</textarea></div>`;

text += `<input type="hidden" name="admin\_edit\_post\_id" id="admin\_edit\_post\_id" value="${p.ID}"/></div>`;

text += `<p id="admin\_edit\_post\_error"></p>`;

text += `<button id="admin\_edit\_post\_submit">${language\_content.admin\_template.admin\_controls.admin\_edit\_post\_submit}</button>`;

$("#admin\_extra\_space").html(text);

});

document.getElementById("admin\_edit\_post\_title").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_post\_send(); } });

document.getElementById("admin\_edit\_post\_srbtitle").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_post\_send(); } });

document.getElementById("admin\_edit\_post\_text").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_post\_send(); } });

document.getElementById("admin\_edit\_post\_srbtext").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_post\_send(); } });

$("#admin\_edit\_post\_submit").on("click", admin\_edit\_post\_send);

}

function admin\_edit\_post\_send() {

var title\_pat = /^([A-Z][a-z]+)+$/,

title = document.getElementById("admin\_edit\_post\_title").value,

srb\_title = document.getElementById("admin\_edit\_post\_srbtitle").value,

text = document.getElementById("admin\_edit\_post\_text").value,

srb\_text = document.getElementById("admin\_edit\_post\_srbtext").value,

id = document.getElementById("admin\_edit\_post\_id").value;

if (title == "" || srb\_title == "" || text == "" || srb\_text == "") {

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

return;

}

if (!title\_pat.test(title) || !title\_pat.test(srb\_title)) {

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.title\_name\_invalid;

return;

}

document.getElementById("admin\_edit\_post\_error").innerHTML = "";

$.ajax({

url: "models/admin/edit\_post.php",

method: "POST",

dataType: "json",

data: {

title: title,

srb\_title: srb\_title,

text: text,

srb\_text: srb\_text,

id: id

},

success: function (data) {

if (data == "Successful") {

show\_notification("update\_post");

document.getElementById("admin\_edit\_post\_title").value = "";

document.getElementById("admin\_edit\_post\_srbtitle").value = "";

document.getElementById("admin\_edit\_post\_text").value = "";

document.getElementById("admin\_edit\_post\_srbtext").value = "";

fill\_admin\_controls("admin\_edit\_post");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "empty\_field":

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

break;

case "title\_name\_invalid":

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.title\_name\_invalid;

break;

default:

document.getElementById("admin\_edit\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_edit\_tag\_show() {

var tags;

$.ajax({

url: "models/admin/tags.php",

method: "GET",

dataType: "json",

success: function (data) {

tags = data;

var text = "";

tags.forEach(function (t) {

text += `<option value="${t.ID}">${lang=="eng"?t.name:(lang=="srb"?t.SRBname:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

admin\_edit\_tag\_show\_second(tags, tags[0].ID);

$("#admin\_extra\_option").on('change', function() {

admin\_edit\_tag\_show\_second(tags, $(this).val());

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_edit\_tag\_show\_second(tags, tag\_id) {

tags.forEach(function (t) {

if (t.ID != tag\_id) { return; }

var text = `<div class="admin\_option\_holder"><label for="admin\_edit\_tag\_name">${language\_content.admin\_template.admin\_controls.admin\_edit\_tag\_name}</label><input type="text" name="admin\_edit\_tag\_name" id="admin\_edit\_tag\_name" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_tag\_name}" value="${t.name}"/></div>`;

text += `<div class="admin\_option\_holder"><label for="admin\_edit\_tag\_srbname">${language\_content.admin\_template.admin\_controls.admin\_edit\_tag\_srbname}</label><input type="text" name="admin\_edit\_tag\_srbname" id="admin\_edit\_tag\_srbname" placeholder="${language\_content.admin\_template.admin\_controls.admin\_edit\_tag\_srbname}" value="${t.SRBname}"/></div>`;

text += `<input type="hidden" name="admin\_edit\_tag\_id" id="admin\_edit\_tag\_id" value="${t.ID}"/></div>`;

text += `<p id="admin\_edit\_tag\_error"></p>`;

text += `<button id="admin\_edit\_tag\_submit">${language\_content.admin\_template.admin\_controls.admin\_edit\_tag\_submit}</button>`;

$("#admin\_extra\_space").html(text);

});

document.getElementById("admin\_edit\_tag\_name").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_tag\_send(); } });

document.getElementById("admin\_edit\_tag\_srbname").addEventListener("keyup", function (e) { if (e.key == "Enter") { admin\_edit\_tag\_send(); } });

$("#admin\_edit\_tag\_submit").on("click", admin\_edit\_tag\_send);

}

function admin\_edit\_tag\_send() {

var name\_pat = /^[A-Z][a-z]+$/,

name = document.getElementById("admin\_edit\_tag\_name").value,

srb\_name = document.getElementById("admin\_edit\_tag\_srbname").value,

id = document.getElementById("admin\_edit\_tag\_id").value;

if (name == "" || srb\_name == "") {

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

return;

}

if (!name\_pat.test(name) || !name\_pat.test(srb\_name)) {

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_name\_invalid;

return;

}

document.getElementById("admin\_edit\_tag\_error").innerHTML = "";

$.ajax({

url: "models/admin/edit\_tag.php",

method: "POST",

dataType: "json",

data: {

name: name,

srb\_name: srb\_name,

id: id

},

success: function (data) {

if (data == "Successful") {

show\_notification("update\_tag");

document.getElementById("admin\_edit\_tag\_name").value = "";

document.getElementById("admin\_edit\_tag\_srbname").value = "";

fill\_admin\_controls("admin\_edit\_tag");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

case "empty\_field":

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.empty\_field;

break;

case "tag\_name\_invalid":

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_name\_invalid;

break;

case "tag\_exists":

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.tag\_exists;

break;

default:

document.getElementById("admin\_edit\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_user\_show() {

$.ajax({

url: "models/admin/users.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (u) {

text += `<option value="${u.ID}">${u.name} ${u.last\_name}</option>`;

});

$("#admin\_extra\_option").html(text);

var button\_text = `<p id="admin\_delete\_user\_error"></p>`;

button\_text += `<button id="admin\_delete\_user\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_user\_submit}</button>`;

$("#admin\_extra\_space").html(button\_text);

$("#admin\_delete\_user\_submit").on("click", admin\_delete\_user\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_user\_send() {

var user = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_user\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_user.php",

method: "POST",

dataType: "json",

data: {

user: user

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_user");

fill\_admin\_controls("admin\_delete\_user");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_user\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_post\_show() {

$.ajax({

url: "models/admin/posts.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (p) {

text += `<option value="${p.ID}">${lang=="eng"?p.title:(lang=="srb"?p.SRBtitle:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

var button\_text = `<p id="admin\_delete\_post\_error"></p>`;

button\_text += `<button id="admin\_delete\_post\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_post\_submit}</button>`;

$("#admin\_extra\_space").html(button\_text);

$("#admin\_delete\_post\_submit").on("click", admin\_delete\_post\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_post\_send() {

var post = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_post\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_post.php",

method: "POST",

dataType: "json",

data: {

post: post

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_post");

fill\_admin\_controls("admin\_delete\_post");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_post\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_tag\_show() {

$.ajax({

url: "models/admin/tags.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (t) {

text += `<option value="${t.ID}">${lang=="eng"?t.name:(lang=="srb"?t.SRBname:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

var button\_text = `<p id="admin\_delete\_tag\_error"></p>`;

button\_text += `<button id="admin\_delete\_tag\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_tag\_submit}</button>`;

$("#admin\_extra\_space").html(button\_text);

$("#admin\_delete\_tag\_submit").on("click", admin\_delete\_tag\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_tag\_send() {

var tag = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_tag\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_tag.php",

method: "POST",

dataType: "json",

data: {

tag: tag

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_tag");

fill\_admin\_controls("admin\_delete\_tag");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_picture\_show() {

var comments;

$.ajax({

url: "models/admin/pictures.php",

method: "GET",

dataType: "json",

success: function (data) {

pictures = data;

var text = "";

pictures.forEach(function (p) {

text += `<option value="${p.ID}">${lang=="eng"?p.alt:(lang=="srb"?p.SRBalt:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

admin\_delete\_picture\_show\_second(pictures, pictures[0].ID);

$("#admin\_extra\_option").on('change', function() {

admin\_delete\_picture\_show\_second(pictures, $(this).val());

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_picture\_show\_second(pictures, picture\_id) {

pictures.forEach(function (p) {

if (p.ID != picture\_id) { return; }

var text = `<img src="${p.src}" alt="${lang=="eng"?p.alt:(lang=="srb"?p.SRBalt:"Error")}"/>`;

text += `<p id="admin\_delete\_picture\_error"></p>`;

text += `<button id="admin\_delete\_picture\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_picture\_submit}</button>`;

$("#admin\_extra\_space").html(text);

});

$("#admin\_delete\_picture\_submit").on("click", admin\_delete\_picture\_send);

}

function admin\_delete\_picture\_send() {

var picture = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_picture\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_picture.php",

method: "POST",

dataType: "json",

data: {

picture: picture

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_picture");

fill\_admin\_controls("admin\_delete\_picture");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_picture\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_picture\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_comment\_show() {

var comments;

$.ajax({

url: "models/admin/comments.php",

method: "GET",

dataType: "json",

success: function (data) {

comments = data;

var text = "";

comments.forEach(function (c) {

text += `<option value="${c.ID}">${c.text}</option>`;

});

$("#admin\_extra\_option").html(text);

admin\_delete\_comment\_show\_second(comments, comments[0].ID);

$("#admin\_extra\_option").on('change', function() {

admin\_delete\_comment\_show\_second(comments, $(this).val());

});

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_comment\_show\_second(comments, comment\_id) {

comments.forEach(function (c) {

if (c.ID != comment\_id) { return; }

var text = `<span>${c.text}</span>`;

text += `<p id="admin\_delete\_comment\_error"></p>`;

text += `<button id="admin\_delete\_comment\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_comment\_submit}</button>`;

$("#admin\_extra\_space").html(text);

});

$("#admin\_delete\_comment\_submit").on("click", admin\_delete\_comment\_send);

}

function admin\_delete\_comment\_send() {

var comment = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_comment\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_comment.php",

method: "POST",

dataType: "json",

data: {

comment: comment

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_comment");

fill\_admin\_controls("admin\_delete\_comment");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_comment\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_comment\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_newsletter\_show() {

$.ajax({

url: "models/admin/newsletter.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (s) {

text += `<option value="${s.ID}">${s.email}</option>`;

});

$("#admin\_extra\_option").html(text);

var button\_text = `<p id="admin\_delete\_newsletter\_error"></p>`;

button\_text += `<button id="admin\_delete\_newsletter\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_newsletter\_submit}</button>`;

$("#admin\_extra\_space").html(button\_text);

$("#admin\_delete\_newsletter\_submit").on("click", admin\_delete\_newsletter\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_newsletter\_send() {

var newsletter = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_newsletter\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_newsletter.php",

method: "POST",

dataType: "json",

data: {

newsletter: newsletter

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_newsletter");

fill\_admin\_controls("admin\_delete\_newsletter");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_newsletter\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_newsletter\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function admin\_delete\_post\_tag\_show() {

$.ajax({

url: "models/admin/post\_tag.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (pt) {

text += `<option value="${pt.postID}-${pt.tagID}">${lang=="eng"?pt.postTitle:(lang=="srb"?pt.postSRBTitle:"Error")} - ${lang=="eng"?pt.tagName:(lang=="srb"?pt.tagSRBName:"Error")}</option>`;

});

$("#admin\_extra\_option").html(text);

var button\_text = `<p id="admin\_delete\_post\_tag\_error"></p>`;

button\_text += `<button id="admin\_delete\_post\_tag\_submit">${language\_content.admin\_template.admin\_controls.admin\_delete\_post\_tag\_submit}</button>`;

$("#admin\_extra\_space").html(button\_text);

$("#admin\_delete\_post\_tag\_submit").on("click", admin\_delete\_post\_tag\_send);

},

error: function (err) {

write\_error\_to\_file(err);

$("#admin\_extra\_space").html(err.responseText);

}

});

}

function admin\_delete\_post\_tag\_send() {

var post\_tag = document.getElementById("admin\_extra\_option").options[document.getElementById("admin\_extra\_option").selectedIndex].value;

document.getElementById("admin\_delete\_post\_tag\_error").innerHTML = "";

$.ajax({

url: "models/admin/delete\_post\_tag.php",

method: "POST",

dataType: "json",

data: {

post\_tag: post\_tag

},

success: function (data) {

if (data == "Successful") {

show\_notification("delete\_post\_tag");

fill\_admin\_controls("admin\_delete\_post\_tag");

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("admin\_delete\_post\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.database\_error;

break;

default:

document.getElementById("admin\_delete\_post\_tag\_error").innerHTML = language\_content.admin\_template.admin\_errors.other\_error;

}

}

});

}

function get\_page\_access\_info() {

$.ajax({

url: "models/page\_access\_info.php",

method: "GET",

dataType: "json",

success: function (data) {

var sum = data.admin\_template + data.appetizer + data.contact\_template + data.dessert + data.main\_course + data.main\_template + data.page\_404;

$(".admin\_controls .visiting\_main span:last-child").html((data.main\_template / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_appetizer span:last-child").html((data.appetizer / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_main\_course span:last-child").html((data.main\_course / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_dessert span:last-child").html((data.dessert / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_contact span:last-child").html((data.contact\_template / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_admin span:last-child").html((data.admin\_template / sum \* 100).toFixed(2) + "%");

$(".admin\_controls .visiting\_404 span:last-child").html((data.page\_404 / sum \* 100).toFixed(2) + "%");

},

error: function (err) {

write\_error\_to\_file(err);

$(".admin\_controls .admin\_file\_holder").html(language\_content.admin\_template.admin\_errors.other\_error);

}

});

}

function get\_logged\_users() {

$.ajax({

url: "models/logged\_users.php",

method: "GET",

dataType: "json",

success: function (data) {

$(".admin\_controls .number\_logged\_users span:last-child").html(data[0].logged\_users);

},

error: function (err) {

write\_error\_to\_file(err);

$(".admin\_controls .admin\_file\_holder").html(language\_content.admin\_template.admin\_errors.other\_error);

}

});

}

function export\_to\_excel() {

$.ajax({

url: "models/export\_excel.php",

method: "GET",

dataType: "json",

success: function (data) {

show\_notification("excel\_saved");

},

error: function (err) {

if (err.status == 200) {

show\_notification("excel\_saved");

} else {

alert(err.responseText);

}

}

});

}



## assets/js/contact\_template.js

function start\_contact() {

setTimeout(start\_second\_contact, 50);

}

function start\_second\_contact() {

if (!$("#contact\_template:visible").length) { setTimeout(start\_second\_contact, 50); } else { load\_contact(); }

}

function load\_contact() {

$("#contact\_template .contact\_area h2").html(language\_content.contact\_template.title);

$("#contact\_template .contact\_area button").html(language\_content.contact\_template.send\_contact);

$("#contact\_template .contact\_area #message\_mail").attr("placeholder", language\_content.contact\_template.send\_mail);

$("#contact\_template .author h2").html(language\_content.contact\_template.author);

$("#contact\_template .author #message\_mail").attr("alt", language\_content.contact\_template.author);

$("#contact\_template .author .author\_name p:first-child").html(language\_content.contact\_template.author\_name);

$("#contact\_template .author .author\_birth p:first-child").html(language\_content.contact\_template.author\_birth);

$("#contact\_template .author .author\_mail p:first-child").html(language\_content.contact\_template.author\_mail);

$("#contact\_template .author .author\_number p:first-child").html(language\_content.contact\_template.author\_number);

$("#contact\_template .author .author\_work p:first-child").html(language\_content.contact\_template.author\_work);

$("#contact\_template .author button").html(language\_content.contact\_template.download\_author\_data);

document.getElementById("message\_mail").addEventListener("keyup", function (e) { if (e.key == "Enter") { mail\_send\_work(); } });

$("#sent\_contact").on("click", mail\_send\_work);

$("#download\_author\_data").on("click", download\_author\_data);

}

function mail\_send\_work() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

mail = document.getElementById("message\_mail").value,

text = document.getElementById("send\_message").value;

if (mail == "" || text == "") {

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.empty\_field;

return;

}

if (!email\_pat.test(mail)) {

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.mail\_invalid;

return;

}

document.getElementById("message\_error").innerHTML = "";

$.ajax({

url: "models/contact.php",

method: "POST",

dataType: "json",

data: {

mail: mail,

text: text

},

success: function (data) {

if (data == "Successful") {

show\_notification("contact\_mail\_sent");

document.getElementById("message\_mail").value = "";

document.getElementById("send\_message").value = "";

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.database\_error;

break;

case "empty\_field":

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.empty\_field;

break;

case "mail\_invalid":

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.mail\_invalid;

break;

default:

document.getElementById("message\_error").innerHTML = language\_content.contact\_template.other\_error;

}

}

});

}

function download\_author\_data() {

$.ajax({

url: "models/save\_word.php",

method: "POST",

dataType: "json",

data: {

author\_name: $("#contact\_template .author .author\_name p:first-child").html(),

author\_name\_val: $("#contact\_template .author .author\_name p:last-child").html(),

author\_birth: $("#contact\_template .author .author\_birth p:first-child").html(),

author\_birth\_val: $("#contact\_template .author .author\_birth p:last-child").html(),

author\_mail: $("#contact\_template .author .author\_mail p:first-child").html(),

author\_mail\_val: $("#contact\_template .author .author\_mail p:last-child").html(),

author\_number: $("#contact\_template .author .author\_number p:first-child").html(),

author\_number\_val: $("#contact\_template .author .author\_number p:last-child").html(),

author\_work: $("#contact\_template .author .author\_work p:first-child").html(),

author\_work\_val: $("#contact\_template .author .author\_work p:last-child").html()

},

success: function (data) {

show\_notification("word\_saved");

},

error: function (err) {

if (err.status == 200) {

show\_notification("word\_saved");

} else {

alert(err.responseText);

}

}

});

}



## assets/js/food\_post\_template.js

function start\_food\_posts() {

if (!$("#main\_panel\_load\_food:visible").length) { setTimeout(start\_food\_posts, 50); } else { load\_food\_posts(); }

}

function load\_food\_posts() {

var sel\_tag = selected\_tags,

type\_id = food\_types.indexOf(selected\_food\_type) + 2;

if (sel\_tag.length == 0) { sel\_tag = [""]; }

$.ajax({

url: "models/posts.php",

method: "GET",

dataType: "json",

data: {

type\_id: type\_id,

selected\_tags: sel\_tag

},

success: function (data) {

post\_to\_array(data);

},

error: function (err) {

write\_error\_to\_file(err);

post\_to\_array(err.responseText);

}

});

}

function post\_to\_array(data, post\_id) {

posts = [];

data.forEach(function (longP) {

var has = false;

posts.forEach(function (post) {

if (post.id == longP.id) { has = true; }

});

if (has) {

posts.forEach(function (post) {

if (post.id != longP.id) { return; }

var hasimage = false,

hastag = false;

post.images.forEach(function (pi) {

if (pi.picid == longP.picid) { hasimage = true; }

});

post.tags.forEach(function (pt) {

if (pt.tagid == longP.tagid) { hastag = true; }

});

if (!hasimage) {

var image = {

picsrc: longP.picsrc,

picSRBalt: longP.picSRBalt,

picalt: longP.picalt,

picid: longP.picid

};

post.images.push(image);

}

if (!hastag) {

var tag = {

tagid: longP.tagid,

tagSRBname: longP.tagSRBname,

tagname: longP.tagname

};

post.tags.push(tag);

}

});

} else {

var el = {

SRBtext: longP.SRBtext,

SRBtitle: longP.SRBtitle,

id: longP.id,

images: [{

picsrc: longP.picsrc,

picSRBalt: longP.picSRBalt,

picalt: longP.picalt,

picid: longP.picid

}],

last\_name: longP.last\_name,

name: longP.name,

postDate: longP.postDate,

postText: longP.postText,

tags: [{

tagid: longP.tagid,

tagSRBname: longP.tagSRBname,

tagname: longP.tagname

}],

title: longP.title,

commentNumber: longP.commentNumber

};

posts.push(el);

}

});

fill\_mid\_posts(posts, post\_id);

}

function fill\_mid\_posts(posts, post\_id) {

var text = "";

posts.forEach(function (post) {

text += `<div class="post\_header"><ul class="post\_tags">`;

post.tags.forEach(function (t) {

text += `<li><span data-id="${t.tagid}">${lang=="eng"?t.tagname:(lang=="srb"?t.tagSRBname:"Error")}</span></li>`;

})

text += "</ul>";

text += `<h2>${lang=="eng"?post.title:(lang=="srb"?post.SRBtitle:"Error")}</h2>`;

text += `<div class="post\_image"><button data-way="back" data-image="${post.images[0].picid}" data-post="${post.id}"><span class="fas fa-arrow-left"></span></button><img src="${post.images[0].picsrc}" alt="${lang=="eng"?post.images[0].picalt:(lang=="srb"?post.images[0].picSRBalt:"Error")}" data-id="${post.images[0].picid}"/><button data-way="forward" data-image="${post.images[0].picid}" data-post="${post.id}"><span class="fas fa-arrow-right"></span></button></div>`;

text += `<ul class="post\_details"><li><span class="fa fa-calendar" aria-hidden="true"></span> ${post.postDate}</li><li><span class="far fa-user"></span> ${language\_content.food\_posts.by} ${post.name} ${post.last\_name}</li><li><span class="fas fa-comment"></span> ${post.commentNumber} ${language\_content.food\_posts.comments}</li></ul>`;

text += "</div>";

if (post\_id) {

text += `<p class="bigger\_letter">${lang=="eng"?post.postText:(lang=="srb"?post.SRBtext:"Error")}</p>`;

} else {

text += `<p class="bigger\_letter">${lang=="eng"?(post.postText.substring(0,146) + "..."):(lang=="srb"?(post.SRBtext.substring(0,146) + "..."):"Error")}</p>`;

text += `<div class="post\_button"><button data-id="${post.id}">${language\_content.food\_posts.continue\_reading}</button></div>`;

}

});

if (posts.length == 0) {

text += `<h2>${language\_content.food\_posts.no\_posts}</h2>`;

}

if (food\_types.indexOf(page\_content) < 0) { return; }

document.getElementById("main\_panel\_load\_food").innerHTML = text;

if (post\_id) { get\_post\_comments(post\_id); }

$(".post\_header .post\_image button").on("click", function(e) {

var div\_tag = $(this).parent(),

way = $(this).attr("data-way"),

post\_id = $(this).attr("data-post"),

image\_id = $(this).attr("data-image"),

image\_tag = $(this).parent().children('img');

e.preventDefault();

posts.forEach(function(p) {

if(p.id != post\_id) { return; }

for (var i = 0; i < p.images.length; i++) {

if (p.images[i].picid == image\_id) {

if (way == "forward") {

var next\_num = i + 1;

if (next\_num == p.images.length) { next\_num = 0; }

image\_tag.attr("alt", lang=="eng"?p.images[next\_num].picalt:(lang=="srb"?pp.images[next\_num].picSRBalt:"Error"));

image\_tag.attr("src", p.images[next\_num].picsrc);

div\_tag.children("button").eq(0).attr("data-image", p.images[next\_num].picid);

div\_tag.children("button").eq(1).attr("data-image", p.images[next\_num].picid);

}

if (way == "back") {

var prev\_num = i - 1;

if (prev\_num < 0) { prev\_num = p.images.length - 1; }

image\_tag.attr("alt", lang=="eng"?p.images[prev\_num].picalt:(lang=="srb"?pp.images[prev\_num].picSRBalt:"Error"));

image\_tag.attr("src", p.images[prev\_num].picsrc);

div\_tag.children("button").eq(0).attr("data-image", p.images[prev\_num].picid);

div\_tag.children("button").eq(1).attr("data-image", p.images[prev\_num].picid);

}

}

}

});

});

$(".post\_button button").on("click", function() {

get\_single\_post($(this).attr("data-id"));

});

finished\_loading\_posts();

}

function finished\_loading\_posts() {

search\_work();

how\_it\_started\_work();

subscribe\_newsletter\_work();

random\_quote\_work();

rss\_click();

login\_work();

add\_post\_work();

$("#logout").on("click", function() {

$.ajax({

url: "models/logout.php",

method: "GET",

dataType: "json",

success: function (data) {

show\_notification("logout", true);

},

error: function (err) {

write\_error\_to\_file(err);

window.location.reload();

}

});

});

}

function search\_work() {

$("#search").attr("placeholder", language\_content.food\_posts.search);

document.getElementById("search").addEventListener("keyup", function (e) {

if (e.key == "Enter") { search\_posts(); }

});

$(".search\_field button").on("click", search\_posts);

}

function search\_posts() {

var sel\_tag = selected\_tags,

text = document.getElementById("search").value,

type\_id = food\_types.indexOf(selected\_food\_type) + 2;

if (sel\_tag.length == 0) { sel\_tag = [""]; }

$.ajax({

url: "models/posts.php",

method: "GET",

dataType: "json",

data: {

type\_id: type\_id,

text: text,

selected\_tags: sel\_tag

},

success: function (data) {

post\_to\_array(data);

},

error: function (err) {

write\_error\_to\_file(err);

post\_to\_array(err.responseText);

}

});

}

function how\_it\_started\_work() {

$("#main\_panel\_load\_edit .how\_it\_started h2").html(language\_content.food\_posts.how\_it\_started\_title);

$("#main\_panel\_load\_edit .how\_it\_started p").html(language\_content.food\_posts.how\_it\_started\_text);

}

function subscribe\_newsletter\_work() {

$("#subscribe").attr("placeholder", language\_content.food\_posts.subscribe);

$("#main\_panel\_load\_edit .subscribe\_newsletter h2").html(language\_content.food\_posts.subscribe\_newsletter\_title);

$("#main\_panel\_load\_edit .subscribe\_newsletter #subscribe\_text").html(language\_content.food\_posts.subscribe\_newsletter\_text);

document.getElementById("subscribe").addEventListener("keyup", function (e) { if (e.key == "Enter") { subscribe\_send(); } });

}

function subscribe\_send() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

subscription\_mail = document.getElementById("subscribe").value;

if (subscription\_mail == "") {

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.subscribe\_empty\_field;

return;

}

if (!email\_pat.test(subscription\_mail)) {

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.not\_valid\_email;

return;

}

document.getElementById("subscribe\_error").innerHTML = "";

$.ajax({

url: "models/subscribe.php",

method: "POST",

dataType: "json",

data: {

subscription\_mail: subscription\_mail

},

success: function (data) {

if (data == "Successful") {

show\_notification("subscription");

document.getElementById("subscribe").value = "";

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.database\_error;

break;

case "subscribe\_empty\_field":

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.subscribe\_empty\_field;

break;

case "subscription\_mail\_exists":

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.subscription\_mail\_exists;

break;

case "not\_valid\_email":

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.not\_valid\_email;

break;

default:

document.getElementById("subscribe\_error").innerHTML = language\_content.food\_posts.other\_error;

}

}

});

}

function random\_quote\_work() {

$("#main\_panel\_load\_edit .quote\_field h2").html(language\_content.food\_posts.quote\_title);

$.ajax({

url: "models/random\_quote.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "<span>\"</span>";

text += lang=="eng"?data.text:(lang=="srb"?data.SRBtext:"Error");

text += "<span>\"</span>";

$("#main\_panel\_load\_edit .quote\_field p").html(text);

},

error: function (err) {

write\_error\_to\_file(err);

$("#main\_panel\_load\_edit .quote\_field p").html(err.responseText);

}

});

}

function get\_single\_post(data) {

var post\_id = data;

$.ajax({

url: "models/single\_post.php",

method: "POST",

dataType: "json",

data: {

postId: data

},

success: function (data) {

post\_to\_array(data, post\_id);

},

error: function (err) {

write\_error\_to\_file(err);

post\_to\_array(err.responseText);

}

});

}

function get\_post\_comments(post\_id) {

$.ajax({

url: "models/comments.php",

method: "GET",

dataType: "json",

data: {

postId: post\_id

},

success: function (data) {

document.getElementById("main\_panel\_load\_food").innerHTML += `<h2 class="comment\_content">${language\_content.food\_posts.comments}:</h2>`;

if (data.length) {

data.forEach(function(c) {

document.getElementById("main\_panel\_load\_food").innerHTML += `<div class="comment\_holder"><p>${c.userName} ${c.userLastName}</p><p>${c.commentText}</p><p>${c.commentDate}</p></div><div id="add\_comment"></div>`;

});

} else {

document.getElementById("main\_panel\_load\_food").innerHTML += `<h3>${language\_content.food\_posts.no\_comment\_show}</h3><div id="add\_comment"></div>`;

}

add\_comment\_view(post\_id);

},

error: function (err) {

write\_error\_to\_file(err);

document.getElementById("main\_panel\_load\_food").innerHTML = err.responseText;

}

});

}

function rss\_click() {

$(".rss\_feed button").on("click", function() {

var location = window.location.href,

index\_location = location.indexOf("index.php"),

new\_location = location.substring(0, index\_location),

rss\_location = new\_location + "data/rss-" + lang + ".xml",

textArea = document.createElement("textarea");

textArea.value = rss\_location;

textArea.className = "always\_hidden";

document.body.appendChild(textArea);

textArea.focus();

textArea.select();

document.execCommand('copy');

document.body.removeChild(textArea);

show\_notification("rss");

});

}

function login\_work() {

if (!$(".login\_register:visible").length) { setTimeout(login\_work, 50); } else { start\_login\_register(); }

}

function add\_post\_work() {

if (!$("#add\_post form:visible").length) { setTimeout(add\_post\_work, 50); } else { start\_add\_post(); }

}

function add\_comment\_view(post\_id) {

document.getElementById("add\_comment").innerHTML = `<h2>${language\_content.food\_posts.add\_comment\_title}</h2><textarea rows="8" cols="60" name="add\_comment\_area" id="add\_comment\_area"></textarea><p id="add\_comment\_error"></p><button id="add\_comment\_button">${language\_content.food\_posts.add\_comment\_button}</button>`;

comment\_work(post\_id);

}

function comment\_work(post\_id) {

document.getElementById("add\_comment\_button").addEventListener("click", function() {

var comment\_text = document.getElementById("add\_comment\_area").value;

if (comment\_text.trim().length < 5) {

document.getElementById("add\_comment\_error").innerHTML = language\_content.food\_posts.empty\_comment;

return;

}

document.getElementById("add\_comment\_error").innerHTML = "";

$.ajax({

url: "models/comment.php",

method: "POST",

dataType: "json",

data: {

post\_id: post\_id,

comment\_text: comment\_text.trim()

},

success: function (data) {

document.getElementById("add\_comment\_area").value = "";

show\_notification("comment", true);

},

error: function (err) {

write\_error\_to\_file(err);

switch(err.responseJSON) {

case "empty\_comment":

document.getElementById("add\_comment\_error").innerHTML = language\_content.food\_posts.empty\_comment;

break;

case "log\_to\_comment":

document.getElementById("add\_comment\_error").innerHTML = language\_content.food\_posts.log\_to\_comment;

break;

case "database\_error":

document.getElementById("add\_comment\_error").innerHTML = language\_content.food\_posts.database\_error;

break;

default:

document.getElementById("add\_comment\_error").innerHTML = language\_content.food\_posts.other\_error;

}

}

});

});

}



## models/admin/post\_tag.php

<?php

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

include("../../config/connection.php");

$query = "SELECT p.title AS postTitle, p.SRBtitle AS postSRBTitle, p.ID AS postID, t.name AS tagName, t.SRBname AS tagSRBName, t.ID AS tagID FROM post\_tag pt INNER JOIN post p ON pt.ID\_post = p.ID INNER JOIN tags t ON pt.ID\_tag = t.ID";

$posttags = executeQuery($query);

echo json\_encode($posttags);

?>



## assets/js/footer\_template.js

var selected\_tags = [];

function fill\_navigation\_footer(text) {

document.getElementById("navigation\_footer").innerHTML = text + document.getElementById("navigation\_footer").innerHTML;

$("#navigation\_footer li a").on("click", function(e) {

if ($.inArray($(this).attr("href"), food\_types) !== -1) {

selected\_food\_type = $(this).attr("href");

start\_food\_posts();

} else {

selected\_food\_type = "";

if ($(this).attr("href") == "index.php?page=main\_template") {

language\_main\_template();

} else if ($(this).attr("href") == "index.php?page=admin\_template") {

start\_admin();

} else {

start\_contact();

}

}

});

}

function fill\_tags\_footer(text) {

document.getElementById("tags\_footer").innerHTML = text;

$("#tags\_footer span").on("click", function(e) {

if(selected\_tags.indexOf($(this).attr("data-id")) >= 0) {

for (var i = 0; i < selected\_tags.length; i++) {

if ( selected\_tags[i] === $(this).attr("data-id")) {

selected\_tags.splice(i, 1);

}

}

$(this).removeClass("selected\_footer\_tag");

} else {

selected\_tags.push($(this).attr("data-id"));

$(this).addClass("selected\_footer\_tag");

}

filter\_tags();

});

}

function filter\_tags() {

load\_food\_posts();

}

function fill\_top\_posts\_footer(text) {

document.getElementById("recent\_posts\_footer").innerHTML = text;

}

function language\_footer\_template() {

$(".footer\_navigation h3").html(language\_content.footer\_template.navigation);

$(".footer\_recents h3").html(language\_content.footer\_template.recent\_posts);

$(".footer\_tags h3").html(language\_content.footer\_template.tags);

}



## assets/js/header\_template.js

function fill\_navigation\_header(text) {

document.getElementById("header\_content").innerHTML = text;

}

$(".header .language img").on("click", function() {

set\_language($(this).attr("data-img"));

});

$("#menu\_click").click(function() {

if ($("#header\_content").hasClass("show\_menu")) {

$("#header\_content").removeClass("show\_menu");

} else {

$("#header\_content").addClass("show\_menu");

}

});

function set\_language(lang) {

localStorage.setItem("language", lang);

location.reload();

}

function language\_header\_template() {

document.getElementsByClassName("language")[0].firstElementChild.alt = language\_content.header\_template.serbian\_flag;

document.getElementsByClassName("language")[0].lastElementChild.alt = language\_content.header\_template.uk\_flag;

}



## assets/js/login\_template.js

function start\_login\_register() {

$(".login\_param label[for='register\_email']").html(language\_content.login\_template.register\_email);

$(".login\_param #register\_email").attr("placeholder", language\_content.login\_template.register\_email\_placeholder);

$(".login\_param label[for='register\_name']").html(language\_content.login\_template.register\_name);

$(".login\_param #register\_name").attr("placeholder", language\_content.login\_template.register\_name\_placeholder);

$(".login\_param label[for='register\_lastname']").html(language\_content.login\_template.register\_lastname);

$(".login\_param #register\_lastname").attr("placeholder", language\_content.login\_template.register\_lastname\_placeholder);

$(".login\_param label[for='register\_password']").html(language\_content.login\_template.register\_password);

$(".login\_param label[for='register\_password\_repeat']").html(language\_content.login\_template.register\_password\_repeat);

$("#register\_button").html(language\_content.login\_template.register);

$("#login\_button").html(language\_content.login\_template.login);

$("#forgot\_password").html(language\_content.login\_template.forgot\_password);

$(".login\_param label[for='login\_email']").html(language\_content.login\_template.register\_email);

$(".login\_param #login\_email").attr("placeholder", language\_content.login\_template.register\_email\_placeholder);

$(".login\_param label[for='login\_password']").html(language\_content.login\_template.register\_password);

$(".login\_register .register\_title").html(language\_content.login\_template.register);

$(".login\_register .login\_title").html(language\_content.login\_template.login);

if ($("#login\_email:visible").length) { login\_register\_events(); }

}

function login\_register\_events() {

document.getElementById("login\_email").addEventListener("keyup", function (e) {

if (e.key == "Enter") { login\_request(); }

});

document.getElementById("login\_password").addEventListener("keyup", function (e) {

if (e.key == "Enter") { login\_request(); }

});

$("#login\_button").on("click", login\_request);

document.getElementById("register\_email").addEventListener("keyup", function (e) {

if (e.key == "Enter") { register\_request(); }

});

document.getElementById("register\_name").addEventListener("keyup", function (e) {

if (e.key == "Enter") { register\_request(); }

});

document.getElementById("register\_lastname").addEventListener("keyup", function (e) {

if (e.key == "Enter") { register\_request(); }

});

document.getElementById("register\_password").addEventListener("keyup", function (e) {

if (e.key == "Enter") { register\_request(); }

});

document.getElementById("register\_password\_repeat").addEventListener("keyup", function (e) {

if (e.key == "Enter") { register\_request(); }

});

$("#register\_button").on("click", register\_request);

$("#forgot\_password").on("click", forgot\_password);

}

function login\_request() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

email = document.getElementById("login\_email").value,

password = document.getElementById("login\_password").value;

if (email == "" || password == "") {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.empty\_field;

return;

}

if (password.length < 6) {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.password\_short;

return;

}

if (!email\_pat.test(email)) {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

return;

}

document.getElementById("login\_error").innerHTML = "";

$.ajax({

url: "models/login.php",

method: "POST",

dataType: "json",

data: {

email: email,

password: password

},

success: function (data) {

if (data == "Successful") {

show\_notification("login", true);

}

if (data == "wrong\_params") {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.wrong\_params;

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.database\_error;

break;

case "empty\_field":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.empty\_field;

break;

case "not\_valid\_email":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

break;

case "password\_short":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.password\_short;

break;

default:

document.getElementById("login\_error").innerHTML = language\_content.login\_template.other\_error;

}

}

});

}

function register\_request() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

name\_pat = /^[A-Z][a-z]+$/,

lastname\_pat = /^([A-Z][a-z]+)+$/,

email = document.getElementById("register\_email").value,

name = document.getElementById("register\_name").value,

lastname = document.getElementById("register\_lastname").value,

password = document.getElementById("register\_password").value,

password\_repeat = document.getElementById("register\_password\_repeat").value;

if (email == "" || name == "" || lastname == "" || password == "" || password\_repeat == "") {

document.getElementById("register\_error").innerHTML = language\_content.login\_template.empty\_field;

return;

}

if (password.length < 6) {

document.getElementById("register\_error").innerHTML = language\_content.login\_template.password\_short;

return;

}

if (password != password\_repeat) {

document.getElementById("register\_error").innerHTML = language\_content.login\_template.passwords\_not\_match;

return;

}

if (!email\_pat.test(email)) {

document.getElementById("register\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

return;

}

if (!name\_pat.test(name) || !lastname\_pat.test(lastname)) {

document.getElementById("register\_error").innerHTML = language\_content.login\_template.name\_invalid;

return;

}

document.getElementById("register\_error").innerHTML = "";

$.ajax({

url: "models/register.php",

method: "POST",

dataType: "json",

data: {

email: email,

name: name,

lastname: lastname,

password: password,

password\_repeat: password\_repeat

},

success: function (data) {

if (data == "Successful") {

show\_notification("register");

document.getElementById("register\_email").value = "";

document.getElementById("register\_name").value = "";

document.getElementById("register\_lastname").value = "";

document.getElementById("register\_password").value = "";

document.getElementById("register\_password\_repeat").value = "";

}

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.database\_error;

break;

case "empty\_field":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.empty\_field;

break;

case "passwords\_not\_match":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.passwords\_not\_match;

break;

case "name\_invalid":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.name\_invalid;

break;

case "email\_taken":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.email\_taken;

break;

case "not\_valid\_email":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

break;

case "password\_short":

document.getElementById("register\_error").innerHTML = language\_content.login\_template.password\_short;

break;

default:

document.getElementById("register\_error").innerHTML = language\_content.login\_template.other\_error;

}

}

});

}

function forgot\_password() {

var email\_pat = /^[a-zA-Z0-9.!#$%&’\*+/=?^\_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)\*$/,

email = document.getElementById("login\_email").value;

if (email == "") {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.empty\_email;

return;

}

if (!email\_pat.test(email)) {

document.getElementById("login\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

return;

}

document.getElementById("login\_error").innerHTML = "";

$.ajax({

url: "models/forgot\_password.php",

method: "POST",

dataType: "json",

data: {

email: email

},

success: function (data) {

show\_notification("forgot\_password");

document.getElementById("login\_email").value = "";

},

error: function (err) {

write\_error\_to\_file(err.responseJSON);

switch(err.responseJSON) {

case "database\_error":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.database\_error;

break;

case "empty\_email":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.empty\_email;

break;

case "not\_valid\_email":

document.getElementById("login\_error").innerHTML = language\_content.login\_template.not\_valid\_email;

break;

default:

document.getElementById("login\_error").innerHTML = language\_content.login\_template.other\_error;

}

}

});

}

## assets/js/main\_template.js

function start\_main\_template() {

if (!$(".landing\_screen") || !$(".landing\_screen h2:visible").length || !$(".landing\_screen img:visible").length) {

setTimeout(start\_main\_template, 50);

} else {

var elArray = document.getElementsByClassName("landing\_screen");

setTimeout(function() {

var firstChild = elArray[0];

fill\_first\_child(firstChild);

}, 10);

}

}

function language\_main\_template() {

setTimeout(start\_main\_template, 50);

}

function fill\_first\_child(firstChild) {

firstChild.firstElementChild.innerHTML = language\_content.main\_template.title;

firstChild.lastElementChild.alt = language\_content.main\_template.baking\_image;

}



## assets/js/main.js

var lang = "eng",

language\_content = "",

food\_types = ['appetizer', 'main\_course', 'dessert'],

selected\_food\_type = '',

posts = [],

page = window.location.href.substr(window.location.href.indexOf("?"), window.location.href.length - 1),

page\_content = page.substr(page.indexOf("=") + 1, page.length - 1);

$(document).ready(function () {

get\_language();

get\_content();

get\_navigation("header");

get\_navigation("footer");

get\_tags("footer");

get\_top\_posts("footer");

});

function get\_language() {

if (window.localStorage.getItem("language")) {

lang = window.localStorage.getItem("language");

} else {

lang = "eng";

}

}

function language\_head() {

if (lang == "eng") {

document.getElementsByTagName("head")[0].innerHTML += `<title>25Hour Kitchen</title>

<meta name="keywords" content="25, hours, kitchen, meal, food"/>

<meta name="description" content="The best blog for your favourite meals and recipes!"/>

<meta name="abstract" content="25Hour Kitchen with RarOks 1008."/>`;

}

if (lang == "srb") {

document.getElementsByTagName("head")[0].innerHTML += `<meta charset="utf-8"/>

<title>25Sati Kuhinja</title>

<meta name="keywords" content="25, sat, kuhinja, obrok, hrana"/>

<meta name="description" content="Najbolji blog za Vašu omiljenu hranu i recepte!"/>

<meta name="abstract" content="25Sati Kuhinja sa RarOks-om 1008."/>`;

}

$("h1").html(language\_content.page\_title);

}

function get\_content() {

$.ajax({

url: "assets/lang/" + lang + ".json",

method: "GET",

dataType: "json",

success: function (data) {

language\_content = data;

language\_head();

language\_header\_template();

language\_footer\_template();

selected\_food\_type = "";

switch(page\_content) {

case "main\_template":

language\_main\_template();

break;

case "appetizer":

case "main\_course":

case "dessert":

selected\_food\_type = page\_content;

start\_food\_posts();

break;

case "contact\_template":

start\_contact();

break;

case "admin\_template":

start\_admin();

break;

default:

page\_content = "main\_template";

language\_main\_template();

break;

}

},

error: function (err) {

write\_error\_to\_file(err);

window.location.reload();

}

});

}

function get\_top\_posts(which) {

$.ajax({

url: "models/top\_posts.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (p) {

text += `<div class="top\_post"><h4>${lang=="eng"?p.title:(lang=="srb"?p.SRBtitle:"Error")}</h4><div><span>${p.postDate}</span><span>${p.name} ${p.lastname}</span></div></div>`;

});

if (which == "footer") {

fill\_top\_posts\_footer(text);

}

},

error: function (err) {

write\_error\_to\_file(err);

fill\_top\_posts\_footer(err.responseText);

}

});

}

function get\_navigation(which) {

$.ajax({

url: "models/navigation.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (n) {

text += `<li><a href="index.php?page=${n.Link}">${lang=="eng"?n.Title:(lang=="srb"?n.SRBTitle:"Error")}</a></li>`;

});

if (which == "header") {

fill\_navigation\_header(text);

}

if (which == "footer") {

fill\_navigation\_footer(text);

}

},

error: function (err) {

write\_error\_to\_file(err.responseText);

if (which == "header") {

fill\_navigation\_header(err.responseText);

}

if (which == "footer") {

fill\_navigation\_footer(err.responseText);

}

}

});

}

function get\_tags(which) {

$.ajax({

url: "models/tags.php",

method: "GET",

dataType: "json",

success: function (data) {

var text = "";

data.forEach(function (n) {

text += `<span data-id="${n.ID}">${lang=="eng"?n.name:(lang=="srb"?n.SRBname:"Error")}</span>`;

});

if (which == "footer") {

fill\_tags\_footer(text);

}

},

error: function (err) {

write\_error\_to\_file(err);

fill\_tags\_footer(err.responseText);

}

});

}

function show\_notification(which, reload) {

$(".notification").css({"display": "flex", "opacity": 1});

switch (which) {

case "rss":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.rss\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.rss\_text;

break;

case "login":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.login\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.login\_text;

break;

case "register":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.register\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.register\_text;

break;

case "logout":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.logout\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.logout\_text;

break;

case "comment":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.comment\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.comment\_text;

break;

case "insert\_user":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.insert\_user\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.insert\_user\_text;

break;

case "insert\_tag":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.insert\_tag\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.insert\_tag\_text;

break;

case "insert\_post\_tag":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.insert\_post\_tag\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.insert\_post\_tag\_text;

break;

case "update\_user":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.update\_user\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.update\_user\_text;

break;

case "update\_post":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.update\_post\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.update\_post\_text;

break;

case "update\_tag":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.update\_tag\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.update\_tag\_text;

break;

case "delete\_user":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_user\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_user\_text;

break;

case "delete\_post":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_post\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_post\_text;

break;

case "delete\_tag":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_tag\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_tag\_text;

break;

case "delete\_picture":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_picture\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_picture\_text;

break;

case "delete\_comment":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_comment\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_comment\_text;

break;

case "delete\_newsletter":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_newsletter\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_newsletter\_text;

break;

case "delete\_post\_tag":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.delete\_post\_tag\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.delete\_post\_tag\_text;

break;

case "subscription":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.subscription\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.subscription\_text;

break;

case "contact\_mail\_sent":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.contact\_mail\_sent\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.contact\_mail\_sent\_text;

break;

case "word\_saved":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.word\_saved\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.word\_saved\_text;

break;

case "excel\_saved":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.excel\_saved\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.excel\_saved\_text;

break;

case "forgot\_password":

document.getElementsByClassName("notification")[0].firstElementChild.innerHTML = language\_content.notification.forgot\_password\_title;

document.getElementsByClassName("notification")[0].lastElementChild.innerHTML = language\_content.notification.forgot\_password\_text;

break;

}

$(".notification").animate({

opacity: 0

}, 5000, function() {

$(".notification").css("display", "none");

if (reload) { window.location.reload(); }

});

}

function write\_error\_to\_file(data) {

$.ajax({

url: "models/write\_error.php",

method: "POST",

dataType: "json",

data: {

text: data

},

success: function (data) {

console.log("Wrote error successfully");

},

error: function (err) {

console.log("Could not write error");

}

});

}



## assets/css/style.css

@font-face {

font-family: IndieFlower;

src: url(../fonts/IndieFlower.ttf);

}

@font-face {

font-family: Slabo;

src: url(../fonts/Slabo27px-Regular.ttf);

}

\* {

margin: 0px;

padding: 0px;

box-sizing: border-box;

}

h1 {

position: absolute;

top: 0;

z-index: -10;

visibility: hidden;

}

#main\_panel {

margin: 0px auto;

display: flex;

justify-content: space-around;

flex-wrap: wrap;

font-family: Slabo;

}

textarea.always\_hidden {

width: 0px;

height: 0px;

}

.notification {

position: sticky;

display: none;

flex-direction: column;

align-items: center;

bottom: 200px;

width: 40%;

margin-left: 30%;

background-color: rgba(255, 255, 255, 0.8);

font-family: Slabo;

}

.notification span {

padding: 10px;

font-weight: bold;

}

.notification span:first-child {

font-size: 24px;

}

.notification span:last-child {

font-size: 18px;

}

@media screen and (max-width: 600px) {

#main\_panel {

flex-direction: column;

align-items: center;

}

}



## assets/css/food\_post\_template.css

#main\_panel\_load\_food {

margin-top: 10px;

width: 60%;

}

#main\_panel\_load\_food .post\_header {

padding-top: 20px;

padding-bottom: 20px;

background-color: #fafafa;

}

#main\_panel\_load\_food .post\_header .post\_tags, #main\_panel\_load\_food .post\_details {

list-style-type: none;

display: flex;

justify-content: center;

}

#main\_panel\_load\_food .post\_header .post\_tags li, #main\_panel\_load\_food .post\_details li {

padding-left: 5px;

padding-right: 5px;

border-left: 1px solid #f6490d;

border-right: 1px solid #f6490d;

color: #f6490d;

font-size: 16px;

}

#main\_panel\_load\_food h2 {

font-style: italic;

text-align: center;

margin-top: 20px;

margin-bottom: 20px;

font-size: 24px;

}

#main\_panel\_load\_food .post\_image {

width: 100%;

position: relative;

display: flex;

flex-direction: column;

align-items: center;

}

#main\_panel\_load\_food .post\_image button {

position: absolute;

font-size: 20px;

padding: 13px 15px;

bottom: 20px;

background-color: #fafafa;

opacity: 0.5;

cursor: pointer;

border: none;

border-radius: 2px;

}

#main\_panel\_load\_food .post\_image button:first-child {

left: 0;

}

#main\_panel\_load\_food .post\_image button:last-child {

right: 0;

}

#main\_panel\_load\_food .post\_image img {

max-width: 100%;

object-fit: scale-down;

margin-top: 20px;

margin-bottom: 20px;

}

#main\_panel\_load\_food p {

font-size: 15px;

text-align: justify;

line-height: 25px;

}

#main\_panel\_load\_food .bigger\_letter::first-letter {

font-size: 45px;

float: left;

line-height: 20px;

}

#main\_panel\_load\_food .post\_button {

display: flex;

justify-content: center;

margin-top: 15px;

margin-bottom: 50px;

}

#main\_panel\_load\_food .post\_button button {

padding: 7px 14px;

cursor: pointer;

font-size: 15px;

border: none;

background-color: rgba(12, 221, 5, 0.8);

border-radius: 3px;

}

#main\_panel\_load\_edit {

margin-top: 10px;

width: 30%;

padding: 5px;

}

#main\_panel\_load\_edit div {

margin-top: 100px;

}

#main\_panel\_load\_edit .search\_field {

display: flex;

}

#main\_panel\_load\_edit .search\_field input, #main\_panel\_load\_edit .search\_field button, #main\_panel\_load\_edit .subscribe\_newsletter input, #add\_comment button {

font-family: Slabo;

font-size: 16px;

padding: 5px;

border: 1px solid #000;

background-color: #fafafa;

}

#main\_panel\_load\_edit .search\_field input {

width: 80%;

border-right: 1px solid rgba(0, 0, 0, 0.2);

}

#main\_panel\_load\_edit .how\_it\_started h2, #main\_panel\_load\_edit .subscribe\_newsletter h2, #main\_panel\_load\_edit .quote\_field h2 {

margin-bottom: 10px;

text-align: center;

}

#main\_panel\_load\_edit .how\_it\_started #subscribe\_text, #main\_panel\_load\_edit .subscribe\_newsletter p, #main\_panel\_load\_edit .quote\_field p {

text-align: justify;

}

#main\_panel\_load\_edit .subscribe\_newsletter #subscribe\_error {

min-height: 20px;

color: #ff0000;

text-align: center;

}

#main\_panel\_load\_edit .rss\_feed {

display: flex;

justify-content: center;

}

#main\_panel\_load\_edit .rss\_feed button {

border: none;

cursor: pointer;

font-size: 40px;

background-color: transparent;

}

#main\_panel\_load\_edit .subscribe\_newsletter input {

margin-left: 5%;

width: 90%;

}

#main\_panel\_load\_edit .subscribe\_newsletter p {

margin-top: 10px;

}

#main\_panel\_load\_edit .quote\_field p {

font-size: 25px;

font-style: italic;

text-decoration: underline;

}

#main\_panel\_load\_edit .quote\_field p span {

color: #ff0000;

}

.comment\_holder {

padding: 5px;

background-color: #fafafa;

border-radius: 5px;

margin-bottom: 50px;

}

.comment\_holder p {

font-size: 18px !important;

}

.comment\_holder p:first-child {

margin-left: 20px;

font-style: italic;

text-decoration: underline;

}

.comment\_holder p:nth-child(2) {

text-align: center !important;

}

.comment\_holder p:last-child {

font-weight: bold;

text-align: right !important;

}

#new\_post\_area {

width: 80%;

padding: 10px;

margin: 25px 0px;

background-color: #fafafa;

border-radius: 2px;

display: flex;

justify-content: space-between;

flex-wrap: wrap;

}

#new\_post\_area .add\_post\_area {

width: 100%;

display: flex;

flex-direction: column;

}

#new\_post\_area .add\_post\_area .welcome\_user {

display: flex;

justify-content: space-between;

font-size: 24px;

margin-bottom: 10px;

}

#new\_post\_area .add\_post\_area .welcome\_user span:last-child {

text-decoration: underline;

font-style: italic;

}

#new\_post\_area .add\_post\_area .welcome\_user button {

font-size: 18px;

padding: 5px 10px;

background-color: #fff;

border: 1px solid #000;

cursor: pointer;

}

#add\_comment {

display: flex;

flex-direction: column;

margin-bottom: 20px;

justify-content: center;

align-items:center;

}

#add\_comment button {

width: 200px;

cursor: pointer;

margin-top: 10px;

}

#add\_comment #add\_comment\_error {

min-height: 22px;

margin-top: 5px;

color: red;

font-size: 16px;

}

#add\_comment textarea {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

}

#main\_panel\_load\_food #page\_numbers {

display: flex;

justify-content: center;

margin-bottom: 30px;

}

#main\_panel\_load\_food #page\_numbers button {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

cursor: pointer;

margin-left: 10px;

}

@media screen and (max-width: 760px) {

#new\_post\_area {

width: 98%;

flex-direction: column;

}

}

@media screen and (max-width: 600px) {

#main\_panel\_load\_food {

width: 95%;

}

#main\_panel\_load\_edit {

width: 95%;

}

}



## assets/css/footer\_template.css

.footer {

width: 100%;

color: #ffffff;

font-family: Slabo;

}

.footer p {

background-color: #0cdd05;

font-size: 18px;

text-align: center;

padding: 5px;

}

.footer .footer\_holder .footer\_part {

width: 33%;

background-color: black;

padding: 10px;

}

.footer .footer\_holder {

display: flex;

justify-content: space-between;

background-color: #0cdd05;

}

.footer .footer\_holder h3 {

color: #00A78C;

text-align: center;

}

#navigation\_footer {

list-style-type: none;

margin-left: 3%;

}

#navigation\_footer li {

margin-top: 5px;

}

#navigation\_footer li a {

text-decoration: none;

color: #00A78C;

}

#navigation\_footer li a:hover {

color: #0cdd05;

}

#tags\_footer {

padding: 10px;

display: flex;

flex-wrap: wrap;

}

#tags\_footer span {

padding: 7px;

color: #00A78C;

border: 1px solid #00A78C;

margin-top: 7px;

margin-left: 7px;

border-radius: 5px;

cursor: pointer;

}

#tags\_footer span:hover {

color: #0cdd05;

border: 1px solid #0cdd05;

}

.selected\_footer\_tag {

color: #0073F3 !important;

border: 1px solid #0073F3 !important;

}

.footer\_part.footer\_recents {

color: #e8e8e8;

}

.footer\_part.footer\_recents h3 {

margin-bottom: 10px;

}

.footer\_part.footer\_recents .top\_post {

padding-left: 10px;

padding-right: 10px;

margin-bottom: 6px;

}

.footer\_part.footer\_recents .top\_post h4 {

text-align: center;

}

.footer\_part.footer\_recents .top\_post div {

display: flex;

justify-content: space-between;

margin-top: 3px;

font-style: italic;

}

@media screen and (max-width: 660px) {

.footer .footer\_holder {

flex-wrap: wrap;

justify-content: center;

}

.footer .footer\_holder .footer\_part {

margin-top: 10px;

width: 98%;

}

}



## assets/css/header\_template.css

.header {

width: 100%;

height: 50px;

padding: 5px;

background-color: #0cdd05;

display: flex;

justify-content: space-around;

align-items: center;

color: #ffffff;

font-family: IndieFlower;

}

.header .language {

width: 70px;

padding: 2px;

display: flex;

justify-content: space-between;

}

.header .language img {

cursor: pointer;

}

.header ul {

list-style-type: none;

display: flex;

width: 80%;

justify-content: space-around;

z-index: 2;

}

.header ul li a {

padding: 2px;

border-radius: 2px;

text-decoration: none;

color: #ffffff;

font-weight: bold;

font-size: 30px;

}

.header ul li a:hover {

background-color: #c2dd05;

}

.header .fas {

font-size: 30px;

margin-left: 15px;

display: none;

}

.header .logo\_and\_nav {

display: flex;

align-items: center;

}

.show\_menu {

height: auto !important;

display: block !important;

align-self: flex-start;

margin-left: 10px;

background-color: #0cdd05;

}

@media screen and (max-width: 920px) {

.header {

justify-content: space-between;

}

.header .fas {

display: block;

}

.header ul {

height: 0;

display: none;

}

}

@media screen and (max-width: 440px) {

.logo {

display: none;

}

}

## assets/css/login\_template.css

#new\_post\_area .login\_register {

width: 49%;

display: flex;

flex-direction: column;

}

#new\_post\_area .login\_register .login\_param {

display: flex;

align-items: center;

justify-content: space-between;

}

#new\_post\_area .login\_register:last-child {

padding-left: 5px;

border-left: 1px dashed red;

}

#new\_post\_area .login\_register:first-child {

padding-right: 5px;

border-right: 1px dashed red;

}

#new\_post\_area .login\_register .login\_param input, #new\_post\_area .login\_register button {

font-size: 16px;

margin-top: 10px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

}

#new\_post\_area .login\_register p {

min-height: 20px;

color: #ff0000;

font-size: 16px;

}

#new\_post\_area .login\_register button {

width: 120px;

cursor: pointer;

}

#new\_post\_area .login\_register button#forgot\_password {

width: 160px;

}

@media screen and (max-width: 760px) {

#new\_post\_area .login\_register {

width: 98%;

}

#new\_post\_area .login\_register:first-child {

border: none;

}

#new\_post\_area .login\_register:last-child {

margin-top: 20px;

border: none;

}

}

@media screen and (max-width: 400px) {

#new\_post\_area .login\_register .login\_param label {

width: 70%;

font-size: 14px;

}

#new\_post\_area .login\_register .login\_param input {

width: 29%;

font-size: 14px;

}

}

## assets/css/main\_template.css

.landing\_screen {

position: relative;

width: 100%;

display: flex;

justify-content: center;

}

.landing\_screen h2 {

font-family: IndieFlower;

font-size: 40px;

position: absolute;

top: 20%;

padding: 10px 30px;

background-color: rgba(12, 221, 5, 0.4);

color: #fff;

}

.landing\_screen img {

width: 100%;

}



## assets/css/add\_post\_template.css

.add\_post\_form h2 {

font-size: 28px;

}

.add\_post\_form .add\_post\_param {

display: flex;

align-items: center;

justify-content: center;

}

.add\_post\_form .add\_post\_param {

margin-top: 15px;

}

.add\_post\_form .add\_post\_param label {

margin-right: 10px;

font-style: italic;

}

.add\_post\_form .add\_post\_param input, .add\_post\_form .add\_post\_param textarea, .add\_post\_form input[type="submit"] {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

}

.add\_post\_form p {

min-height: 20px;

color: #ff0000;

font-size: 16px;

}

.add\_post\_form input[type="submit"] {

width: 120px;

cursor: pointer;

}

@media screen and (max-width: 400px) {

.add\_post\_form .add\_post\_param label, .add\_post\_form .add\_post\_param div label {

width: 29%;

font-size: 14px;

}

.add\_post\_form .add\_post\_param div input {

width: initial;

}

.add\_post\_form .add\_post\_param input, .add\_post\_form .add\_post\_param div {

width: 70%;

font-size: 14px;

}

}



## assets/css/admin\_template.css

#admin\_panel {

width: 96%;

margin-left: 2%;

display: flex;

flex-direction: column;

margin-top: 15px;

margin-bottom: 20px;

align-items: center;

}

#admin\_panel h2 {

text-align: center;

margin-bottom: 30px;

}

#admin\_panel #admin\_option {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

margin-top: 15px;

width: 300px;

}

#admin\_panel .admin\_controls {

width: 90%;

margin-top: 25px;

background-color: #fafafa;

border-radius: 5px;

padding-top: 10px;

display: flex;

flex-direction: column;

align-items: center;

}

#admin\_panel .admin\_controls #admin\_extra\_option {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

margin-top: 15px;

width: 300px;

margin-bottom: 50px;

display: none;

}

#admin\_panel .admin\_controls #admin\_extra\_space {

width: 100%;

display: flex;

flex-direction: column;

align-items: center;

}

#admin\_panel .admin\_controls #admin\_extra\_space form {

width: 100%;

display: flex;

flex-direction: column;

align-items: center;

}

#admin\_panel .admin\_controls #admin\_extra\_space span {

user-select: none;

font-size: 20px;

font-style: italic;

}

#admin\_panel .admin\_controls #admin\_extra\_space img {

width: 60%;

}

#admin\_panel .admin\_controls #admin\_extra\_space .admin\_option\_holder {

width: 98%;

margin-left: 1%;

display: flex;

justify-content: center;

align-items: center;

margin-top: 10px;

margin-bottom: 10px;

}

#admin\_panel .admin\_controls #admin\_extra\_space .admin\_option\_holder label {

font-size: 16px;

margin-right: 20px;

width: 20%;

text-align: right;

}

#admin\_panel .admin\_controls #admin\_extra\_space .admin\_option\_holder input, #admin\_panel .admin\_controls #admin\_extra\_space .admin\_option\_holder select, #admin\_panel .admin\_controls #admin\_extra\_space .admin\_option\_holder textarea {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

width: 60%;

}

#admin\_panel .admin\_controls #admin\_extra\_space button, #admin\_panel .admin\_controls.other\_controls button {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

width: 200px;

margin-top: 20px;

margin-bottom: 30px;

cursor: pointer;

}

#admin\_panel .admin\_controls.other\_controls button {

width: 220px;

}

#admin\_panel .admin\_controls.other\_controls a {

font-size: 24px;

text-decoration: none;

}

#admin\_panel .admin\_controls .admin\_file\_holder {

width: 100%;

display: flex;

align-items: center;

justify-content: center;

margin-top: 10px;

font-size: 20px;

}

#admin\_panel .admin\_controls #admin\_extra\_space p {

min-height: 20px;

color: #ff0000;

font-size: 16px;

}

@media screen and (max-width: 400px) {

#admin\_panel #admin\_option, #admin\_panel .admin\_controls #admin\_extra\_option {

width: 99%;

}

}



## assets/css/contact\_template.css

#contact\_template {

width: 96%;

margin-left: 2%;

display: flex;

justify-content: space-between;

flex-wrap: wrap;

padding: 10px;

}

#contact\_template .contact\_area {

display: flex;

flex-direction: column;

width: 65%;

}

#contact\_template .contact\_area h2 {

margin-top: 10px;

margin-left: 20px;

}

#contact\_template .contact\_area #message\_error {

min-height: 20px;

color: #ff0000;

}

#contact\_template .contact\_area input, #contact\_template .contact\_area textarea, #contact\_template .contact\_area button, #contact\_template .author button {

font-size: 16px;

border: 1px solid #000;

background-color: #fff;

padding: 5px;

border-radius: 4px;

margin-top: 15px;

}

#contact\_template .contact\_area button, #contact\_template .author button {

width: 200px;

cursor: pointer;

}

#contact\_template .author {

width: 33%;

display: flex;

flex-direction: column;

align-items: center;

}

#contact\_template .author h2 {

margin-top: 20px;

}

#contact\_template .author img {

margin-top: 20px;

width: 80%;

}

#contact\_template .author .author\_data {

margin-top: 10px;

display: flex;

}

#contact\_template .author .author\_data p {

text-align: center;

}

#contact\_template .author .author\_data p:first-child {

margin-right: 5px;

}

@media screen and (max-width: 820px) {

#contact\_template {

flex-direction: column;

}

#contact\_template .contact\_area, #contact\_template .author {

width: 100%;

}

}



## assets/css/page\_404.css

.page\_404 {

width: 100%;

}

.page\_404 img {

width: 100%;

}

## data/errors.txt (PRIMER)

ERROR\_TYPE 11-06-2019 19:01:05



## data/log.txt (PRIMER)

main\_template 10-06-2019 20:41:36 ::1

appetizer 10-06-2019 20:41:37 ::1

main\_course 10-06-2019 20:41:38 ::1

dessert 10-06-2019 20:41:39 ::1

contact\_template 10-06-2019 20:41:40 ::1

admin\_template 10-06-2019 20:41:42 ::1

page\_404 10-06-2019 20:41:46 ::1

admin\_template 10-06-2019 21:34:14 ::1

admin\_template 10-06-2019 21:36:23 ::1

admin\_template 10-06-2019 21:39:59 ::1

admin\_template 10-06-2019 21:40:48 ::1

admin\_template 10-06-2019 21:41:07 ::1

appetizer 10-06-2019 21:44:44 ::1

appetizer 10-06-2019 21:44:56 ::1

appetizer 10-06-2019 21:44:59 ::1

appetizer 10-06-2019 21:45:07 ::1

appetizer 10-06-2019 21:46:10 ::1

appetizer 10-06-2019 21:46:15 ::1

appetizer 10-06-2019 21:47:23 ::1

appetizer 10-06-2019 21:47:44 ::1

main\_course 10-06-2019 21:47:49 ::1

main\_template 10-06-2019 21:47:50 ::1

contact\_template 10-06-2019 21:47:52 ::1

main\_course 10-06-2019 21:50:13 ::1

main\_course 10-06-2019 21:51:23 ::1

main\_course 10-06-2019 21:51:35 ::1

admin\_template 10-06-2019 21:51:39 ::1

admin\_template 10-06-2019 21:51:53 ::1

admin\_template 10-06-2019 21:52:06 ::1

admin\_template 10-06-2019 21:52:20 ::1

admin\_template 10-06-2019 21:54:49 ::1

appetizer 10-06-2019 21:55:14 ::1

appetizer 10-06-2019 21:55:20 ::1

appetizer 10-06-2019 21:55:31 ::1

main\_course 10-06-2019 21:55:32 ::1

main\_course 10-06-2019 21:55:38 ::1

main\_course 10-06-2019 21:55:50 ::1

admin\_template 10-06-2019 21:55:57 ::1

admin\_template 10-06-2019 21:57:15 ::1

admin\_template 10-06-2019 21:57:36 ::1

admin\_template 10-06-2019 21:59:27 ::1

admin\_template 10-06-2019 22:02:11 ::1

admin\_template 10-06-2019 22:02:31 ::1

admin\_template 10-06-2019 22:04:36 ::1

admin\_template 11-06-2019 17:29:40 ::1

admin\_template 11-06-2019 17:29:49 ::1

main\_course 11-06-2019 17:36:21 ::1

main\_course 11-06-2019 17:36:28 ::1

main\_course 11-06-2019 17:37:03 ::1

main\_course 11-06-2019 17:42:14 ::1

dessert 11-06-2019 17:42:17 ::1

contact\_template 11-06-2019 17:42:18 ::1

main\_course 11-06-2019 17:42:19 ::1

appetizer 11-06-2019 17:42:19 ::1

main\_template 11-06-2019 17:42:20 ::1

main\_course 11-06-2019 17:42:23 ::1

dessert 11-06-2019 17:42:24 ::1

contact\_template 11-06-2019 17:42:25 ::1

contact\_template 11-06-2019 17:42:26 ::1

contact\_template 11-06-2019 17:42:28 ::1

appetizer 11-06-2019 17:42:29 ::1

appetizer 11-06-2019 17:43:31 ::1

contact\_template 11-06-2019 17:43:33 ::1

dessert 11-06-2019 17:43:34 ::1

dessert 11-06-2019 17:43:35 ::1

dessert 11-06-2019 17:58:03 ::1

admin\_template 11-06-2019 17:58:06 ::1

admin\_template 11-06-2019 17:59:16 ::1

admin\_template 11-06-2019 18:00:29 ::1

admin\_template 11-06-2019 18:01:05 ::1

admin\_template 11-06-2019 18:01:09 ::1

admin\_template 11-06-2019 18:01:29 ::1

admin\_template 11-06-2019 18:01:34 ::1

admin\_template 11-06-2019 18:03:55 ::1

admin\_template 11-06-2019 18:04:17 ::1

admin\_template 11-06-2019 18:04:34 ::1

admin\_template 11-06-2019 18:04:40 ::1

admin\_template 11-06-2019 18:04:55 ::1

admin\_template 11-06-2019 18:30:17 ::1

admin\_template 11-06-2019 18:31:02 ::1

admin\_template 11-06-2019 18:31:41 ::1

admin\_template 11-06-2019 18:37:06 ::1

admin\_template 11-06-2019 18:38:43 ::1

admin\_template 11-06-2019 18:43:04 ::1

main\_course 11-06-2019 18:45:15 ::1

main\_course 11-06-2019 18:45:28 ::1

main\_course 11-06-2019 18:45:36 ::1

main\_course 11-06-2019 18:48:02 ::1

main\_course 11-06-2019 18:48:05 ::1

main\_course 11-06-2019 18:48:07 ::1

main\_course 11-06-2019 18:49:44 ::1

main\_course 11-06-2019 19:00:40 ::1

main\_course 11-06-2019 19:00:58 ::1

main\_course 11-06-2019 19:01:47 ::1

main\_course 11-06-2019 19:03:36 ::1

main\_course 11-06-2019 19:04:33 ::1

main\_course 11-06-2019 19:52:09 ::1

main\_course 11-06-2019 19:52:11 ::1

dessert 11-06-2019 19:52:18 ::1

appetizer 11-06-2019 19:52:23 ::1

appetizer 11-06-2019 19:52:35 ::1

main\_course 11-06-2019 19:52:45 ::1

main\_course 11-06-2019 19:59:43 ::1

main\_course 11-06-2019 20:01:48 ::1

dessert 11-06-2019 20:01:50 ::1

main\_course 11-06-2019 20:01:51 ::1

appetizer 11-06-2019 20:01:52 ::1

appetizer 11-06-2019 20:03:49 ::1

appetizer 11-06-2019 20:05:32 ::1

appetizer 11-06-2019 20:06:26 ::1

appetizer 11-06-2019 20:07:08 ::1

appetizer 11-06-2019 20:07:19 ::1

appetizer 11-06-2019 20:07:53 ::1

appetizer 11-06-2019 20:08:03 ::1

appetizer 11-06-2019 20:08:12 ::1

appetizer 11-06-2019 20:08:20 ::1

appetizer 11-06-2019 20:08:29 ::1

appetizer 11-06-2019 20:10:01 ::1

appetizer 11-06-2019 20:10:13 ::1

appetizer 11-06-2019 20:12:23 ::1

appetizer 11-06-2019 20:12:30 ::1

appetizer 11-06-2019 20:16:23 ::1

appetizer 11-06-2019 20:16:47 ::1

appetizer 11-06-2019 20:17:47 ::1

appetizer 11-06-2019 20:17:55 ::1

appetizer 11-06-2019 20:19:00 ::1

appetizer 11-06-2019 20:19:47 ::1

appetizer 11-06-2019 20:20:20 ::1

appetizer 11-06-2019 20:20:47 ::1

appetizer 11-06-2019 20:21:13 ::1

appetizer 11-06-2019 20:23:05 ::1

main\_course 11-06-2019 20:23:10 ::1

page\_404 11-06-2019 20:27:03 ::1

page\_404 11-06-2019 20:27:07 ::1



## data/rss-eng.xml

<?xml version="1.0" encoding="UTF-8"?>

<rss version="2.0">

<channel>

<title>25Hour Kitchen Posts</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>All newest recipes on one place.</description>

<item>

<title>Buns</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Milan risotto</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Marko Markovic</description>

</item>

<item>

<title>Baked tomato</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Broccoli Orecchiette</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Pizza roll</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Marko Markovic</description>

</item>

<item>

<title>Stewed beans</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Stuffed peppers</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Fried chicken</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Beef stew</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Chocolate cake</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Blueberry pie</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Muhallebi</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Blinis with honey</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

</channel>

</rss>



## data/rss-srb.xml

<?xml version="1.0" encoding="UTF-8"?>

<rss version="2.0">

<channel>

<title>25Sati Kuhinja Objave</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Svi najnoviji recepti na jednom mestu.</description>

<item>

<title>Kiflice</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Milanski rižoto</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Marko Markovic</description>

</item>

<item>

<title>Pečen paradajz</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Brokoli Orecchiette</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Pizza rolat</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Marko Markovic</description>

</item>

<item>

<title>Pohovana boranija</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Punjene paprike</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Pohovana piletina</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

<item>

<title>Juneći gulaš</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Čokoladni kolač</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Nesto Tamo</description>

</item>

<item>

<title>Pita od borovnica</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Muhallebi</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Fantasticna Kuvarka</description>

</item>

<item>

<title>Blinis sa medom</title>

<link>http://25h-kitchen.epizy.com/index.php</link>

<description>Drugi User</description>

</item>

</channel>

</rss>



## config/.env

DBNAME=\*\*\*\*\*\*\*\*\*\*\*

SERVER=\*\*\*\*\*\*\*\*\*\*\*

USERNAME=\*\*\*\*\*\*\*\*\*\*\*

PASSWORD=\*\*\*\*\*\*\*\*\*\*\*



## config/config.php

<?php

define("ABSOLUTE\_PATH", $\_SERVER["DOCUMENT\_ROOT"]);

define("ENV\_FAJL", ABSOLUTE\_PATH."/config/.env");

define("LOG\_FAJL", ABSOLUTE\_PATH."/data/log.txt");

define("ERROR\_FAJL", ABSOLUTE\_PATH."/data/errors.txt");

define("SERVER", env("SERVER"));

define("DATABASE", env("DBNAME"));

define("USERNAME", env("USERNAME"));

define("PASSWORD", env("PASSWORD"));

function env($naziv){

$podaci = file(ENV\_FAJL);

$vrednost = "";

foreach($podaci as $key=>$value){

$konfig = explode("=", $value);

if($konfig[0]==$naziv){

$vrednost = trim($konfig[1]);

}

}

return $vrednost;

}



## models/admin/newsletter.php

<?php

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

include("../../config/connection.php");

$query = "SELECT email, ID FROM newsletter\_subscribers";

$subscribers = executeQuery($query);

echo json\_encode($subscribers);

?>



## config/connection.php

<?php

require\_once "config.php";

try {

$conn = new PDO("mysql:host=".SERVER.";dbname=".DATABASE.";charset=utf8", USERNAME, PASSWORD);

$conn->setAttribute(PDO::ATTR\_DEFAULT\_FETCH\_MODE, PDO::FETCH\_OBJ);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

}

catch(PDOException $ex){

echo $ex->getMessage();

}

function executeQuery($query){

global $conn;

return $conn->query($query)->fetchAll();

}

function zabeleziPristupStranici ($content) {

$open = fopen(LOG\_FAJL, "a");

if ($open) {

$date = date('d-m-Y H:i:s');

fwrite($open, "{$content}\t{$date}\t{$\_SERVER['REMOTE\_ADDR']}\t\n");

fclose($open);

}

if (isset($\_SESSION["user"])) {

global $conn;

$conn -> query("UPDATE users SET LastTimeSeen = NOW() WHERE ID = ".$\_SESSION["user"]->ID);

}

}



## robots.txt

User-agent: \*

Disallow: /assets/



## .htaccess

ErrorDocument 404 http://25h-kitchen.epizy.com/index.php?page=404

ErrorDocument 403 http://25h-kitchen.epizy.com/index.php?page=404

Options All -Indexes

# HIJERARHIJSKI PREGLED

