Visoka škola strukovnih studija za informacione i komunikacione tehnologije



# Dokumentacija

https://anotez.000webhostapp.com

Nikola Nedeljković 2/17

# Sadržaj

[Dokumentacija 1](#_Toc35439113)

[Sadržaj 2](#_Toc35439114)

[1. UVOD 5](#_Toc35439115)

[1.1 Korišćeni programski / markap jezici 5](#_Toc35439116)

[1.2 Opis funkcionalnosti 5](#_Toc35439117)

[1.3 Izgled stranica 10](#_Toc35439118)

[2. ORGANIZACIJA 11](#_Toc35439119)

[2.1 Organizacija sajta 11](#_Toc35439120)

[2.2 Organizacija baze 12](#_Toc35439124)

[2.3 Slike stranica sa funkcionalnostima 13](#_Toc35439129)

[3. KODOVI 20](#_Toc35439130)

[3.1 web.php 20](#_Toc35439131)

[3.2. admin.blade.php 21](#_Toc35439136)

[3.3. index.blade.php 22](#_Toc35439137)

[3.4 footer.blade.php 22](#_Toc35439144)

[3.5. header.blade.php 22](#_Toc35439146)

[3.6 contact.blade.php 23](#_Toc35439155)

[3.7 login.blade.php 23](#_Toc35439165)

[3.8. main.blade.php 24](#_Toc35439176)

[3.9 note.blade.php 24](#_Toc35439188)

[3.10 register.blade.php 25](#_Toc35439201)

[3.11. author.blade.php 26](#_Toc35439215)

[3.12 notes\_filter.blade.php 26](#_Toc35439230)

[3.13. addnote.blade.php 27](#_Toc35439246)

[3.14. admin\_edit\_category.blade.php 27](#_Toc35439263)

[3.15. admin\_edit\_user.blade.php 27](#_Toc35439281)

[3.16. admin\_navigation.blade.php 28](#_Toc35439300)

[3.17. admin\_show\_author.blade.php 28](#_Toc35439303)

[3.18. admin\_show\_categories.blade.php 29](#_Toc35439304)

[3.19. admin\_show\_logs.blade.php 29](#_Toc35439326)

[3.20. admin\_show\_users.blade.php 30](#_Toc35439330)

[3.21. admin\_add\_author.blade.php 30](#_Toc35439354)

[3.22. admin\_add\_category.blade.php 31](#_Toc35439379)

[3.23. admin\_add\_user.blade.php 31](#_Toc35439405)

[3.24. admin\_edit\_author.blade.php 32](#_Toc35439432)

[3.25. Kernel.php 32](#_Toc35439460)

[3.26. AddCategoryRequest.php 34](#_Toc35439489)

[3.27. AuthorEditImageRequest.php 35](#_Toc35439519)

[3.28. AuthorImageRequest.php 35](#_Toc35439550)

[3.29. ContectRequest.php 36](#_Toc35439582)

[3.30. DeleteUserRequest.php 36](#_Toc35439615)

[3.31. EditUserRequest.php 37](#_Toc35439649)

[3.32. GetNotesRequest.php 38](#_Toc35439684)

[3.33. LoginRequest.php 38](#_Toc35439720)

[3.34. NoteRequest.php 39](#_Toc35439757)

[3.35. AuthorImage.php 39](#_Toc35439795)

[3.36. Category.php 40](#_Toc35439834)

[3.37. LogAdmin.php 41](#_Toc35439874)

[3.38. Note.php 41](#_Toc35439915)

[3.39. User.php 43](#_Toc35439957)

[3.40. IsAdminMiddleware.php 45](#_Toc35440000)

[3.41. LogAdminMiddleware.php 45](#_Toc35440044)

[3.42. LoggedInMiddleware.php 46](#_Toc35440089)

[3.43. LogVisit.php 46](#_Toc35440135)

[3.44. ContactController.php 47](#_Toc35440182)

[3.45. LoginController.php 48](#_Toc35440230)

[3.46. MainController.php 49](#_Toc35440279)

[3.47. NoteController.php 50](#_Toc35440329)

[3.48. RegisterController.php 51](#_Toc35440380)

[3.49. AdminAuthorController.php 52](#_Toc35440432)

[3.50. AdminCategoryController.php 54](#_Toc35440485)

[3.51. AdminLogController.php 55](#_Toc35440539)

[3.52. AdminUserController.php 55](#_Toc35440594)

[3.53. AuthorController.php 57](#_Toc35440650)

[3.54. main.js 57](#_Toc35440707)

[3.55. search.js 59](#_Toc35440765)

[3.56. author\_game.js 61](#_Toc35440824)

[3.57. style.css 62](#_Toc35440884)

[4. HIJERARHIJSKI PREGLED 72](#_Toc35440885)

# UVOD

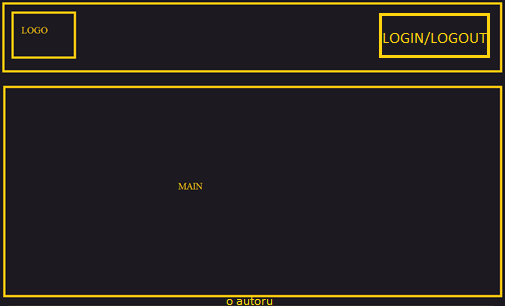
## Korišćeni programski / markap jezici

* Za izradu ovog sajta korišćene su sledeće tehnologije:
* Mark-up jezici: HTML 5
* CSS3
* Skripting-programski jezici: JavaScript, PHP
* JQuery – JavaScript framework
* Laravel – PHP framework
* Sajt je napisan u PHP Storm-u

## Opis funkcionalnosti

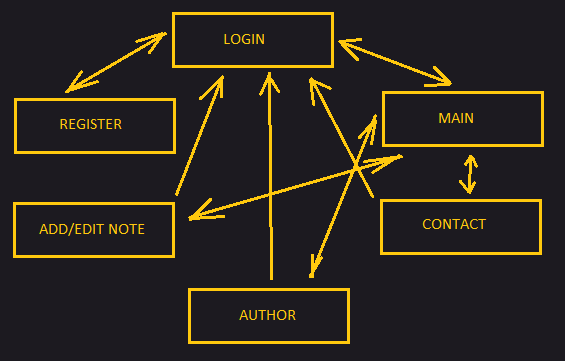
* **web.php**
* Rad sa rutama
* **views/layouts/admin.blade.php**
* Layout za admin panel
* **views/layouts/index.blade.php**
* Layout za sve ostale stranice sajta
* **views/fixed/footer.blade.php**
* Footer stranice
* **views/fixed/header.blade.php**
* Header stranice
* Dinamički prikaz naslova
* Dinamički prikaz login i logout linka
* Generisanje csrf tokena za ajax
* **views/pages/contact.blade.php**
* Kontakt stranica sajta
* Forma za slanje kontakt poruke
* **views/pages/login.blade.php**
* Login stranica sajta
* Forma za logovanje korisnika
* **views/pages/main.blade.php**
* Prikaz svih note-a korisnika
* Pretraživanje po naslovu putem ajax-a
* Pretraživanje po kategoriji putem ajax-a
* Straničenje note-a
* **views/pages/note.blade.php**
* Prikaz jedne note ili dodavanje nove
* Forma za insert i edit note
* **views/pages/register.blade.php**
* Stranica za registraciju korisnika
* Forma za registraciju
* **views/pages/author.blade.php**
* Autor stranica
* Dohvatanje slika putem ajax-a
* Igra sa slikama putem ajax-a i javascript-a
* **views/partials/notes\_filter.blade.php**
* Filtriranje note-a parcijalni view
* **views/partials/addnote.blade.php**
* Parcijalni view dodavanja nove note
* **views/admin/admin\_edit\_category.blade.php**
* Admin stranica za edit kategorije
* Forma za edit kategorije
* **views/admin/admin\_edit\_user.blade.php**
* Admin stranica za edit korisnika
* Forma za editovanje korisnika
* **views/admin/admin\_navigation.blade.php**
* Admin deo za navigaciju
* **views/admin/admin\_show\_author.blade.php**
* Tabelarni prikaz svih slika za igru na stranica autora
* **views/admin/admin\_show\_categories.blade.php**
* Tabelarni prikaz svih kategorija
* **views/admin/admin\_show\_logs.blade.php**
* Tabelarni prikaz svih logova u tabeli log
* **views/admin/admin\_show\_users.blade.php**
* Tabelarni prikaz svih korisnika
* **views/admin/admin\_add\_author.blade.php**
* Forma za dodavanje nove slike autora za igru na autor stranici
* **views/admin/admin\_add\_category.blade.php**
* Forma za dodavanje nove kategorije
* **views/admin/admin\_add\_user.blade.php**
* Forma za dodavanje novog korisnika
* **views/admin/admin\_edit\_author.blade.php**
* Forma za izmenu već postojećeg korisnika
* **Kernel.php**
* Rutiranje middleware-a
* **Requests/AddCategoryRequest.php**
* Request za dodavanje nove kategorije
* **Requests/AuthorEditImageRequest.php**
* Request za izmenu slike
* **Requests/AuthorImageRequest.php**
* Request za dodavanje nove slike
* **Requests/ContactRequest.php**
* Request za kontakt formu
* **Requests/DeleteUserRequest.php**
* Request za brisanje korisnika
* **Requests/EditUserRequest.php**
* Request za izmenu korisnika
* **Requests/GetNotesRequest.php**
* Request za dohvatanje svih note-a
* **Requests/LoginRequest.php**
* Request za logovanje korisnika
* **Requests/NoteRequest.php**
* Request za dodavanje nove note
* **Models/AuthorImage.php**
* Model za tabelu slika autora
* **Models/Category.php**
* Model za tabelu kategorije
* **Models/LogAdmin.php**
* Model za tabelu log
* **Models/Note.php**
* Model za tabelu note
* **Models/User.php**
* Modlel za tabelu korisnika
* **Middleware/IsAdminMiddleware.php**
* Middleware za proveru da li je korisnika admin
* **Middleware/LogAdminMiddleware.php**
* Middleware za logovanje desavanja na admin panelu
* **Middleware/LoggedInMiddleware.php**
* Middleware za proveru da li je korisnik logovan
* **Middleware/LogVisit.php**
* Middleware za logovanje posete stranice
* **Controllers/ContactController.php**
* Kontroler za slanje kontakt mail-a putem PHPMailer-a
* **Controllers/LoginController.php**
* Kontroler za login i logout
* **Controllers/MainController.php**
* Kontroler za stranicu prikaza svih note-a
* **Controllers/NoteController.php**
* Kontroler za dohvatanje, dodavanje, izmenu i brisanje note-a
* **Controllers/RegisterController.php**
* Kontroler za registraciju korisnika
* **Controllers/AdminAuthorController.php**
* Kontroler za prikaz, dodavanje, izmenu i brisanje autor slika sa admin strane
* **Controllers/AdminCategoryController.php**
* Kontroler za prikaz, dodavanje, izmenu i brisanje kategorija sa admin strane
* **Controllers/AdminLogController.php**
* Kontroler za prikaz logova na admin strani
* **Controllers/AdminUserController.php**
* Kontroler za prikaz, dodavanje, izmenu i brisanje korisnika sa admin strane
* **Controllers/AuthorController.php**
* Kontroler za prikaz slika o autoru na autor stranici
* **public/main.js**
* Javascript za front-end proveru svih formi
* **public/search.js**
* Javascript za pretragu i prikazivanje svih note-a putem javascripta sa straničenjem
* **public/author\_game.js**
* Javascript za prikaz slika i igru na stranici o autoru putem ajax-a
* **public/style.css**
* CSS za dizajn celog sajta

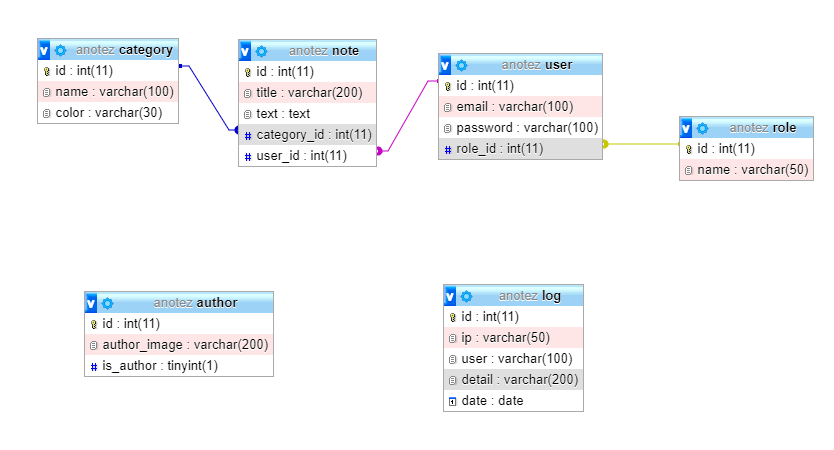
## Izgled stranica



# ORGANIZACIJA

## Organizacija sajta

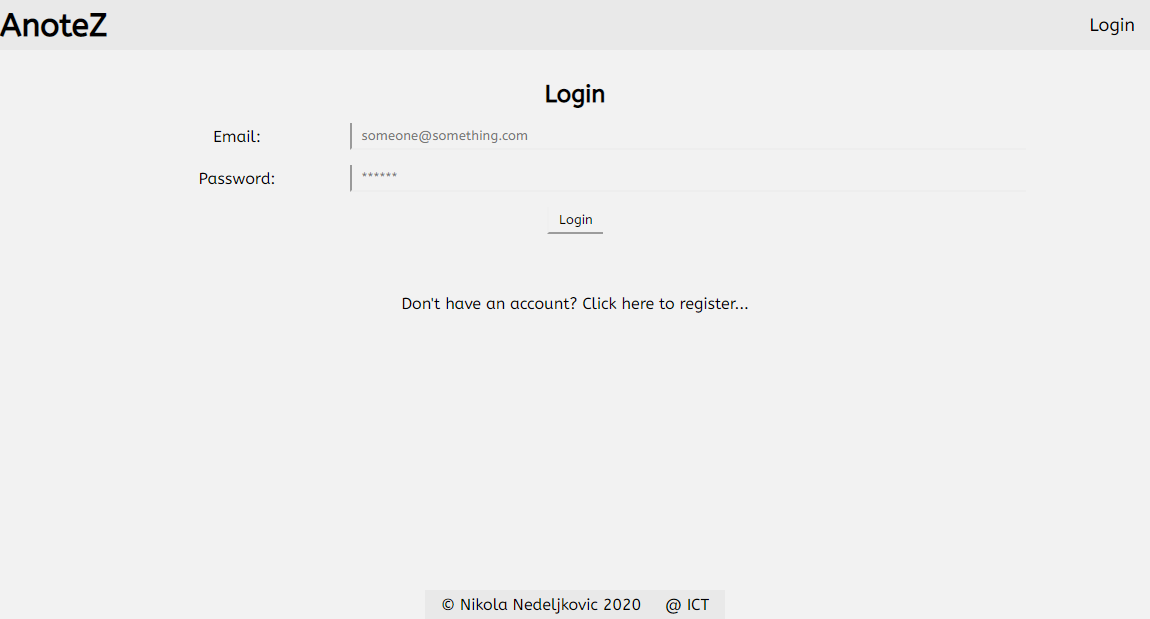


1. 

## Organizacija baze

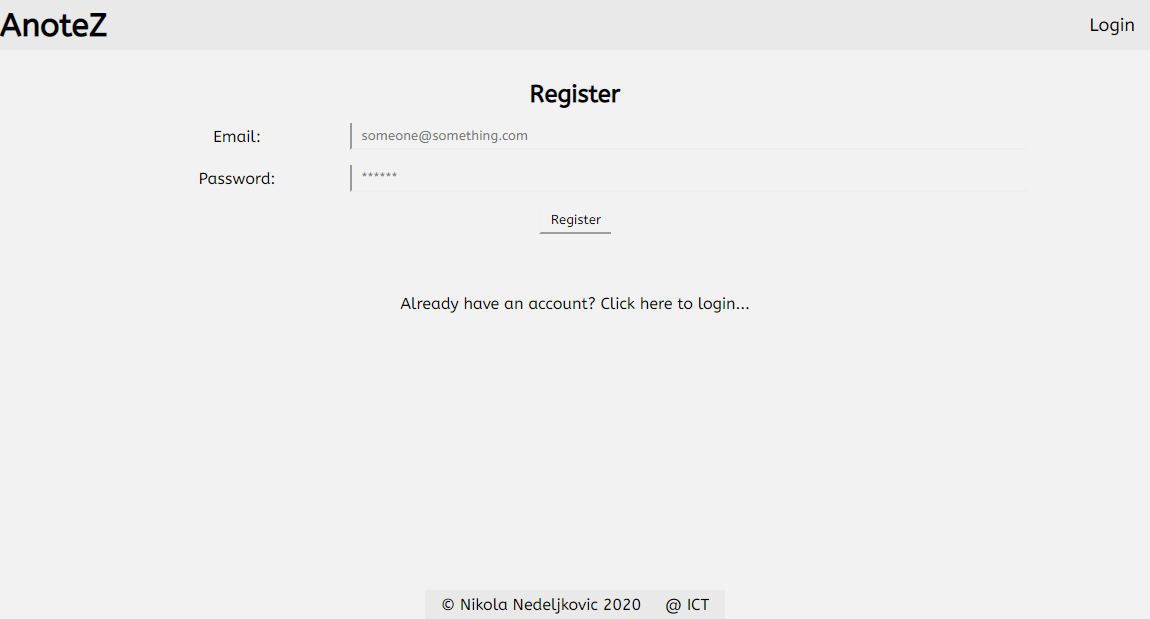


## Slike stranica sa funkcionalnostima



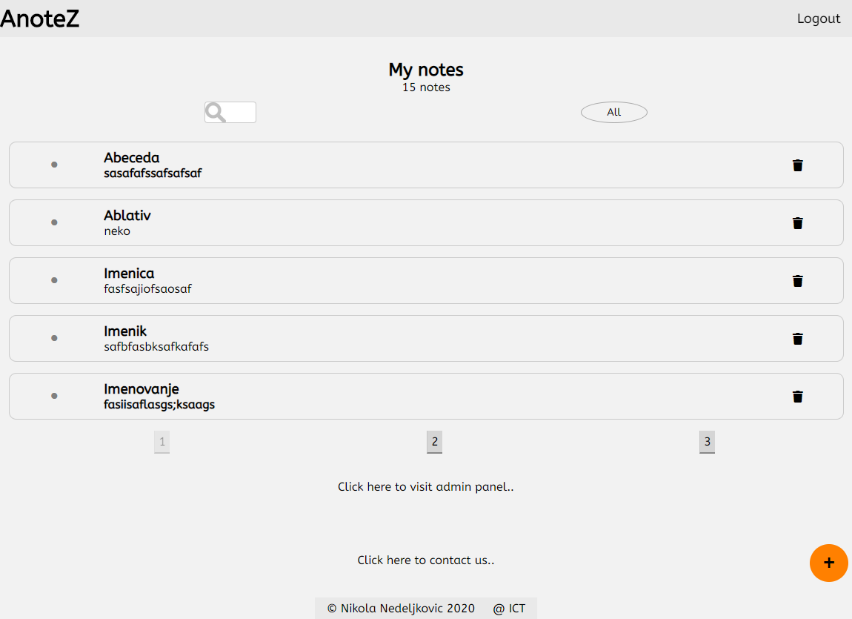
Login

* Login forma, obrada forme na korisničkoj strani, logovanje



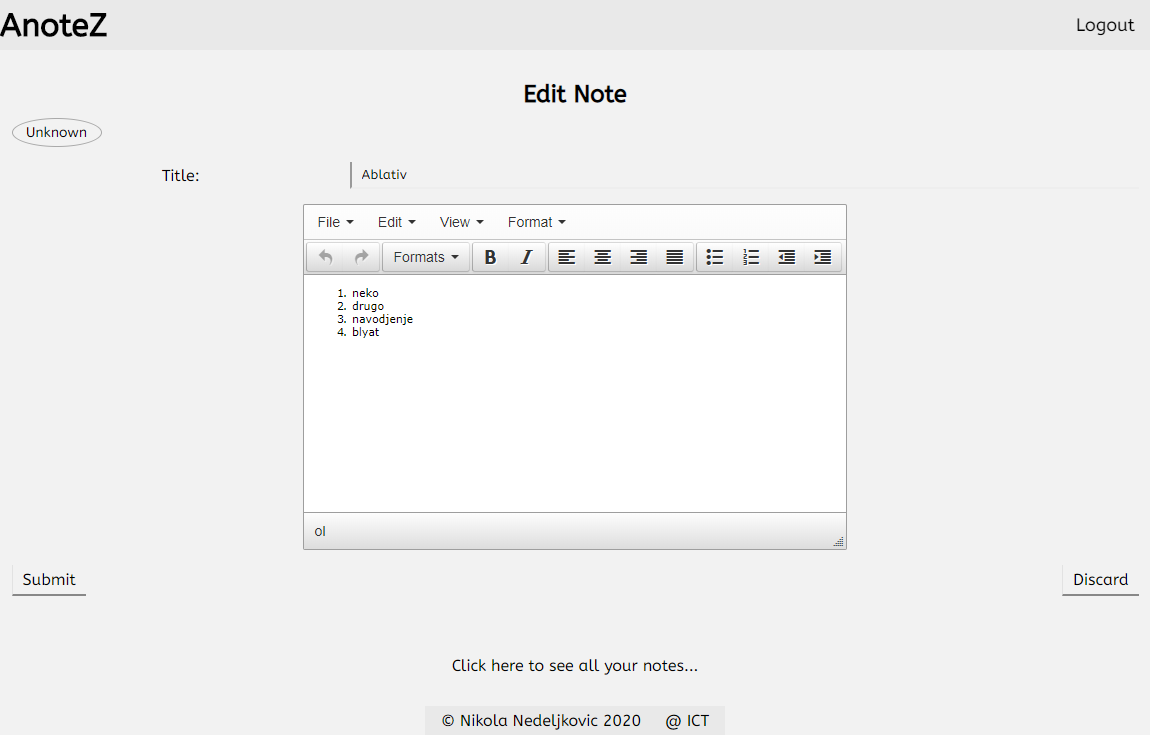
Login

* Registraciona forma, obrada forme na korisničkoj strani, registracija, provera duplog maila



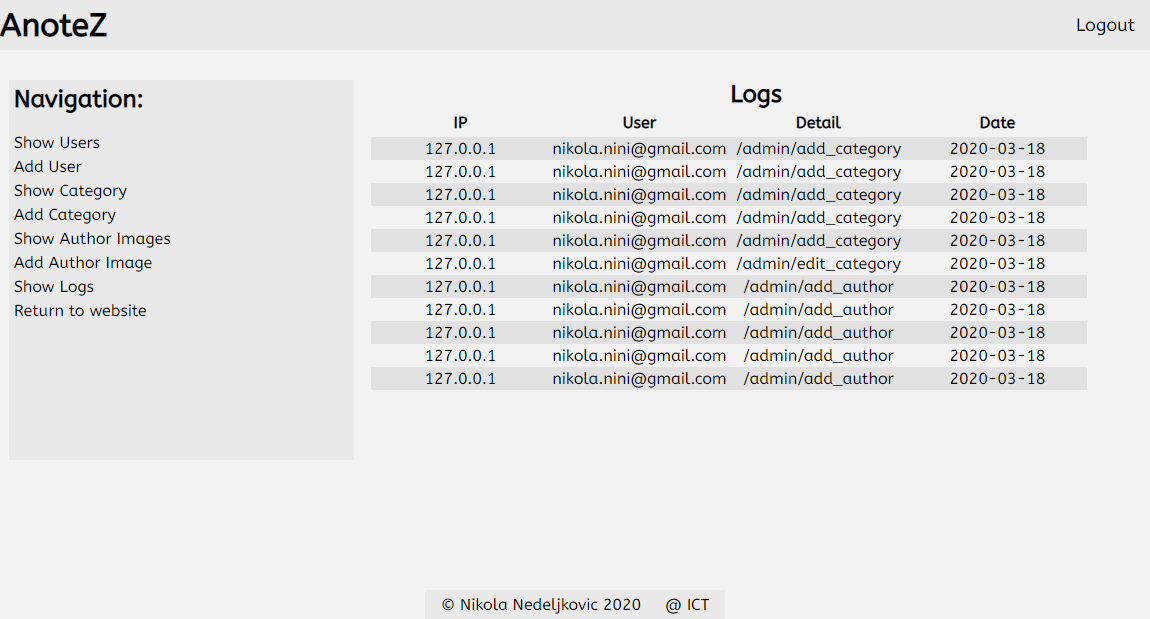
Main

* Dinamičko dohvatanje svih note-a preko ajax-a, straničenje, brisanje, formatiranje preko teksta i kategorije



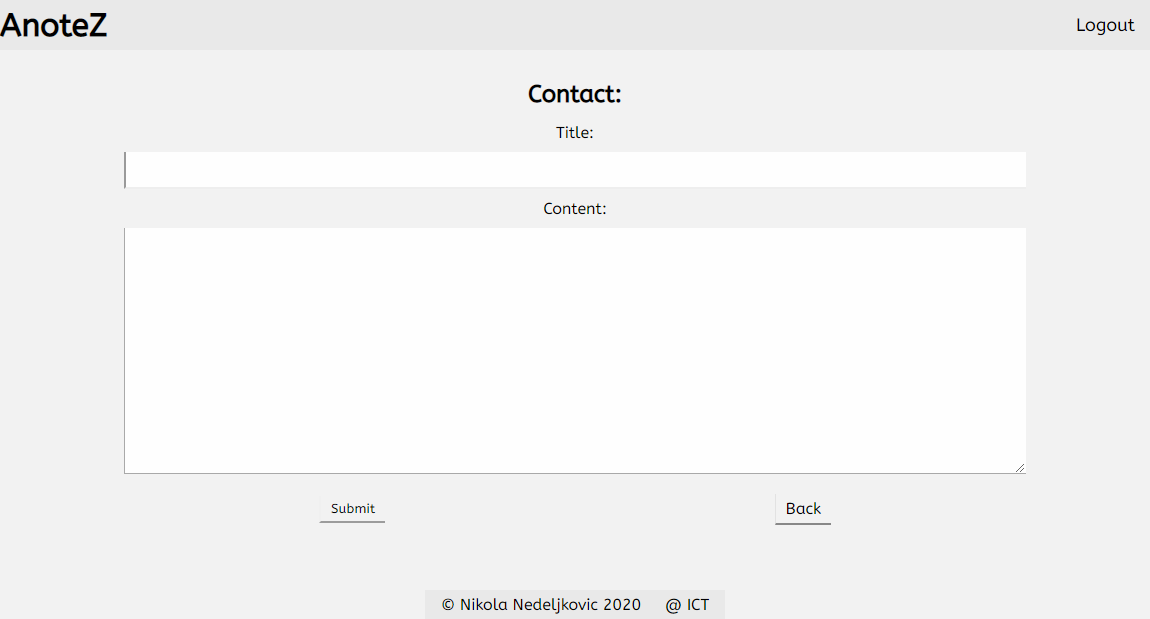
Add i Edit note

* Forma za izmenu i dodavanje note-e u zavisnosti od prosledjenih parametar, korisnička provera.



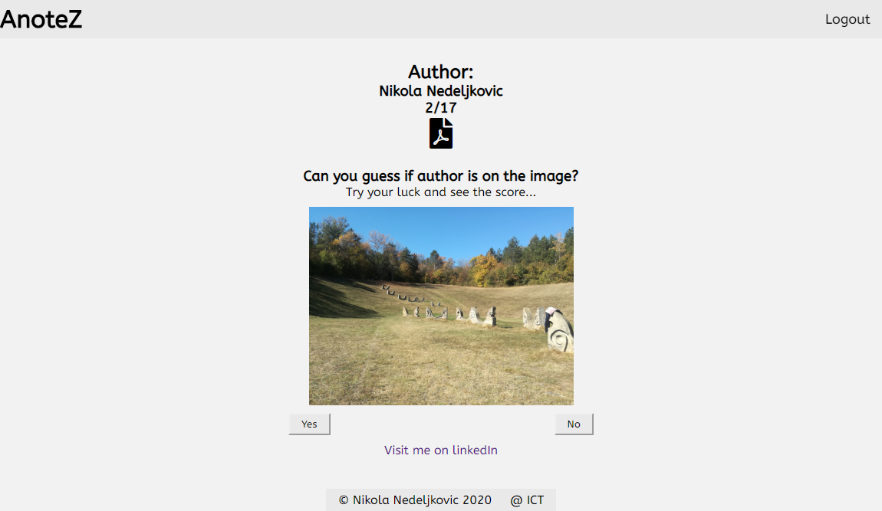
Admin

* Admin panel za sve funkcije administratora, tabelarni prikaz, izmenu i brisanje, kao i prikaz svih logova



Contact

* Foma za slanje poruke administratoru sajta.



Author

* Autor stranica sa igrom za rad sa slikama, putem ajax-a.

# KODOVI

## web.php

<?php

/\*

|--------------------------------------------------------------------------

| Web Routes

|--------------------------------------------------------------------------

|

| Here is where you can register web routes for your application. These

| routes are loaded by the RouteServiceProvider within a group which

| contains the "web" middleware group. Now create something great!

|

\*/

Route::get('/', 'MainController@main')->middleware('loggedIn');

Route::get('/get\_notes', 'MainController@getNotes')->middleware('loggedIn');

Route::post('/login', 'LoginController@loginDo');

Route::get('/login', 'LoginController@login')->middleware('loggedIn');

Route::get('/logout', 'LoginController@logout');

Route::get('/register', 'RegisterController@register')->middleware('loggedIn');

Route::post('/register', 'RegisterController@registerDo');

Route::get('/note/{id?}', 'NoteController@index')->middleware('loggedIn')->where(["id"=> "\d+"]);

Route::post('/editnote', 'NoteController@editNote')->middleware('loggedIn');

Route::post('/addnote', 'NoteController@addNote')->middleware('loggedIn');

Route::get('/deletenote/{id?}', 'NoteController@deleteNote')->middleware('loggedIn')->where(["id"=> "\d+"]);

Route::get('/author', 'AuthorController@index')->middleware('loggedIn');

Route::get('/author\_get\_images', 'AuthorController@getImages')->middleware('loggedIn');

Route::get('/contact', 'ContactController@index')->middleware('loggedIn');

Route::post('/contact\_us', 'ContactController@contactUs')->middleware('loggedIn');

Route::prefix("/admin")->middleware(["loggedIn", "isAdmin"])-> group(**function**(){

Route::get("/", 'AdminUserController@showUsers');

Route::get("/show\_users", 'AdminUserController@showUsers');

Route::get("/add\_user", 'AdminUserController@addUser');

Route::post("/add\_user", 'AdminUserController@adminAddUser')->middleware('logAdminVisit');

Route::get('/edit\_user/{id?}', 'AdminUserController@editUser')->where(["id"=> "\d+"]);

Route::post('/edit\_user', 'AdminUserController@adminEditUser')->middleware('logAdminVisit');

Route::get('/delete\_user/{id?}', 'AdminUserController@deleteUser')->where(["id"=> "\d+"])->middleware('logAdminVisit');

Route::get("/show\_category", 'AdminCategoryController@showCategory');

Route::get("/add\_category", 'AdminCategoryController@addCategory');

Route::post("/add\_category", 'AdminCategoryController@adminAddCategory')->middleware('logAdminVisit');

Route::get('/edit\_category/{id?}', 'AdminCategoryController@editCategory')->where(["id"=> "\d+"]);

Route::post('/edit\_category', 'AdminCategoryController@adminEditCategory')->middleware('logAdminVisit');

Route::get('/delete\_category/{id?}', 'AdminCategoryController@deleteCategory')->where(["id"=> "\d+"])->middleware('logAdminVisit');

Route::get("/show\_author", 'AdminAuthorController@showAuthor');

Route::get("/add\_author", 'AdminAuthorController@addAuthor');

Route::post("/add\_author", 'AdminAuthorController@adminAddAuthor')->middleware('logAdminVisit');

Route::get('/edit\_author/{id?}', 'AdminAuthorController@editAuthor')->where(["id"=> "\d+"]);

Route::post('/edit\_author', 'AdminAuthorController@adminEditAuthor')->middleware('logAdminVisit');

Route::get('/delete\_author/{id?}', 'AdminAuthorController@deleteAuthor')->where(["id"=> "\d+"])->middleware('logAdminVisit');

Route::get("/show\_logs", 'AdminLogController@showLogs');

});



## admin.blade.php

**@include('fixed.header')**

<div class=**"admin\_part"**>

**@include('admin.admin\_navigation')**

<div class=**"admin\_layout\_content"**>

**@yield('admin\_part')**

<p class=**"error\_text"** id=**"error\_text"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

**@isset($errors)**

**@foreach(($errors->all()) as $er)**

**{{$er}}**

**@endforeach**

**@endisset**

</p>

</div>

</div>

**@include('fixed.footer')**

## index.blade.php

**@include('fixed.header')**

<div class=**"middle\_part"**>

**@yield('middle\_part')**

</div>

**@include('fixed.footer')**



## footer.blade.php

<div class=**"footer\_data"**><a href=**"{{url('/author')}}"**><p class=**"copyright"**>*&copy;* **Nikola Nedeljkovic 2020** *&nbsp;&nbsp;&nbsp;* **@ ICT**</p></a></div>

<script type=**"text/javascript"** src=**"{{ asset('main.js') }}"**></script>

</body>

</html>



## header.blade.php

<!DOCTYPE html>

<html>

<head>

<title>**AnoteZ - @yield('title\_part')**</title>

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

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

<meta name=**"copyright"** content=**"RarOks 1008 Entertainment @ 2020"**/>

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

<meta name=**"keywords"** content=**"notes, anotez"**/>

<meta name=**"description"** content=**"From any device access your notes."**/>

<meta name=**"abstract"** content=**"Best notes app around."**/>

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

<link href=**"**https://fonts.googleapis.com/css?family=ABeeZee&display=swap**"** rel=**"stylesheet"**/>

<link rel=**"stylesheet"** href=**"{{ asset('style.css') }}"**/>

<script src=**"**http://tinymce.cachefly.net/4.0/tinymce.min.js**"**></script>

<script type=**"text/javascript"** src=**"{{ asset('jquery.min.js') }}"**></script>

<script type=**"text/javascript"**>

***var*** csrf **=** '{{csrf\_token()}}'**;**

</script>

</head>

<body>

<div class=**"header\_holder"**>

<a href=**"{{url('/')}}"** >

<h1>**AnoteZ**</h1>

</a>

**@if(session()->has('user'))**

<a href=**"{{url('/logout')}}"** class=**"login\_logout\_button"**>**Logout**</a>

**@else**

<a href=**"{{url('/login')}}"** class=**"login\_logout\_button"**>**Login**</a>

**@endif**

</div>



## contact.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**Contact Us**

**@endsection**

**@section('middle\_part')**

<h2>**Contact:**</h2>

<form class=**"contact\_form"** action=**"{{url('contact\_us')}}"** method=**"POST"** onsubmit=**"return contact\_check()"**>

**@csrf**

<label>**Title:** </label>

<input type=**"text"** id=**"title"** name=**"title"**/>

<label>**Content:** </label>

<textarea name=**"content"** id=**"content"** rows=**"15"** columns=**"50"**></textarea>

<div>

<input type=**"submit"** value=**"Submit"** name=**"note\_submit"**/>

<a href=**"{{url('/')}}"**>**Back**</a>

</div>

<p class=**"error\_text"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

**@isset($errors)**

**@foreach(($errors->all()) as $er)**

**{{$er}}**

**@endforeach**

**@endisset**

</p>

</form>

**@endsection**



## login.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**Login**

**@endsection**

**@section('middle\_part')**

<h2>**Login**</h2>

<form action=**"{{ url("**/login") }}**" method="**POST" onsubmit=**"return login\_check()"**>

**@csrf**

<div class=**"divform"**>

<label>**Email:**</label>

<input type=**"email"** name=**"email"** id=**"email"** placeholder=**"someone@something.com"**/>

</div>

<div class=**"divform"**>

<label>**Password:**</label>

<input type=**"password"** name=**"password"** id=**"password"** placeholder=**"\*\*\*\*\*\*"**/>

</div>

<div class=**"divform"**>

<input type=**"submit"** name=**"btnLogin"** value=**"Login"**/>

</div>

<p class=**"error\_text"** id=**"error\_text\_place"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

**@isset($errors)**

**@foreach(($errors->all()) as $er)**

**{{$er}}**

**@endforeach**

**@endisset**

</p>

</form>

<a class=**"login\_register\_link"** href=**"{{url('/register')}}"**>**Don't have an account? Click here to register...**</a>

**@endsection**



## main.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**My notes**

**@endsection**

**@section('middle\_part')**

<h2>**My notes**</h2>

<p class=**"note\_count"**>**{{$note\_count}} notes**</p>

**@include("partials.notes\_filter")**

<div id=**"my\_notes"**></div>

<p class=**"error\_text"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

</p>

**@if(strtolower(session('user')->name) == 'admin')**

<a class=**"admin\_panel\_link"** href=**"{{url('/admin')}}"**><p>**Click here to visit admin panel..**</p></a>

**@endif**

<a class=**"admin\_panel\_link"** href=**"{{url('/contact')}}"**><p>**Click here to contact us..**</p></a>

<script type=**"text/javascript"** src=**"{{ asset('search.js') }}"**></script>

**@endsection**

**@include('partials.addnote')**



## note.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**@isset($note)**

**{{$note->title}}**

**@else**

**Add note**

**@endisset**

**@endsection**

**@section('middle\_part')**

**@isset($note)**

<h2>**Edit Note**</h2>

**@else**

<h2>**Add note**</h2>

**@endisset**

<form class=**"note\_form"** action=**"@isset($note) {{url("**/editnote")}} @else {{url(**"/addnote"**)}} @endisset" method=**"POST"** onsubmit=**"return note\_check()"**>

**@csrf**

<select name=**"note\_category"** id=**"note\_category"**>

**@foreach( $categories as $cat)**

<option value=**"{{$cat->id}}"** @isset($note) @if($note->**name==$cat->name) selected @endif @endisset>{{$cat->name}}**</option>

**@endforeach**

</select>

<div class=**"divform addedittitle"**>

<label>**Title:** </label>

<input type=**"text"** @isset($note) value=**"{{$note->title}}"** @endisset id=**"note\_title"** name=**"note\_title"**/>

</div>

<textarea name=**"note\_text"** id=**"note\_text"** rows=**"15"** columns=**"50"**>**@isset($note){{$note->text}}@endisset**</textarea>

**@isset($note)**

<input type=**"hidden"** value=**"{{$note->id}}"** name=**"note\_id"**/>

**@endisset**

<div class=**"divform"**>

<input type=**"submit"** value=**"Submit"** name=**"note\_submit"**/>

<a href=**"{{url('/')}}"**>**Discard**</a>

</div>

</form>

<p class=**"error\_text"** id=**"error\_text"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

**@isset($errors)**

**@foreach(($errors->all()) as $er)**

**{{$er}}**

**@endforeach**

**@endisset**

**</p>**

**<a class="login\_register\_link" href="{{url('/')}}">Click here to see all your notes...</a>**

**<script type="application/x-javascript">**

**tinymce.init({selector:'#note\_text'});**

**</script>**

**@endsection**



## register.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**Register**

**@endsection**

**@section('middle\_part')**

<h2>**Register**</h2>

<form action=**"{{ url("**/register") }}**" method="**POST" onsubmit=**"return register\_check()"**>

**@csrf**

<div class=**"divform"**>

<label>**Email:**</label>

<input type=**"email"** name=**"email"** id=**"email"** placeholder=**"someone@something.com"**/>

</div>

<div class=**"divform"**>

<label>**Password:**</label>

<input type=**"password"** name=**"password"** id=**"password"** placeholder=**"\*\*\*\*\*\*"**/>

</div>

<div class=**"divform"**>

<input type=**"submit"** name=**"btnRegister"** value=**"Register"**/>

</div>

<p class=**"error\_text"** id=**"error\_text\_place"**>

**@if(session()->has('message'))**

**{{ session('message') }}**

**@endif**

**@isset($errors)**

**@foreach(($errors->all()) as $er)**

**{{$er}}**

**@endforeach**

**@endisset**

</p>

</form>

<a class=**"login\_register\_link"** href=**"{{url('/login')}}"**>**Already have an account? Click here to login...**</a>

**@endsection**



## author.blade.php

**@extends('layouts.index')**

**@section('title\_part')**

**Author**

**@endsection**

**@section('middle\_part')**

<h2>**Author:**</h2>

<h3 class=**"author\_center"**>**Nikola Nedeljkovic**</h3>

<h3 class=**"author\_center"**>**2/17**</h3>

<a href=**"{{ asset('documentation.pdf') }}"** class=**"documentation\_holder"**><span class=**"fas fa-file-pdf"**></span></a>

<div class=**"author\_game"** id=**"author\_game"**>

</div>

<p class=**"error\_text author\_error\_text"** id=**"error\_text"**>

<a href=**"**https://www.linkedin.com/in/nikola-nedeljkovic-8110b5150/**"**>**Visit me on linkedIn**</a>

</p>

<script type=**"text/javascript"** src=**"{{ asset('author\_game.js') }}"**></script>

**@endsection**



## notes\_filter.blade.php

<div class=**"notes\_filter"**>

<input type=**"text"** class=**"search\_notes"** id=**"search\_notes"** name=**"searchtext"**/>

<select name=**"category\_select"** id=**"category\_select"**>

<option value=**"0"**>**All**</option>

**@foreach( $categories as $cat)**

<option value=**"{{$cat->id}}"**>**{{$cat->name}}**</option>

**@endforeach**

</select>

</div>



## addnote.blade.php

<div class=**"button\_add\_note"**><a href=**"{{url('/note')}}"**>*&nbsp;&#43;&nbsp;*</a></div>



## admin\_edit\_category.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Edit Category**

**@endsection**

**@section('admin\_part')**

<h2>**Edit Category**</h2>

<form action=**"{{ url("**/admin/edit\_category") }}**" method="**POST" onsubmit=**"return addcategory\_check()"**>

**@csrf**

<div>

<label>**Name:**</label>

<input type=**"text"** name=**"categoryname"** id=**"categoryname"** value=**"{{$category->name}}"**/>

</div>

<div>

<label>**Color:**</label>

<input type=**"color"** name=**"categorycolor"** id=**"categorycolor"** value=**"{{$category->color}}"**/>

</div>

<input type=**"hidden"** name=**"id"** value=**"{{$category->id}}"**/>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

<a href=**"{{url("**/admin/show\_category")}}**" class="**admin\_back\_link">**Back**</a>

</div>

</form>

**@endsection**



## admin\_edit\_user.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Edit User**

**@endsection**

**@section('admin\_part')**

<h2>**Edit User**</h2>

<form action=**"{{ url("**/admin/edit\_user") }}**" method="**POST" onsubmit=**"return admin\_edit\_user()"**>

**@csrf**

<div>

<label>**Email:**</label>

<input type=**"email"** name=**"email"** id=**"email"** value=**"{{$user->email}}"** placeholder=**"someone@something.com"**/>

</div>

<div>

<label>**Password:**</label>

<input type=**"password"** name=**"password"** id=**"password"** placeholder=**"\*\*\*\*\*\*"**/>

</div>

<input type=**"hidden"** name=**"id"** value=**"{{$user->id}}"**/>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

<a href=**"{{url("**/admin/show\_users")}}**" class="**admin\_back\_link">**Back**</a>

</div>

</form>

**@endsection**



## admin\_navigation.blade.php

<div class=**"admin\_navigation"**>

<h2>**Navigation:**</h2>

<ul>

<li><a href=**"{{url('admin/show\_users')}}"**>**Show Users**</a></li>

<li><a href=**"{{url('admin/add\_user')}}"**>**Add User**</a></li>

<li><a href=**"{{url('admin/show\_category')}}"**>**Show Category**</a></li>

<li><a href=**"{{url('admin/add\_category')}}"**>**Add Category**</a></li>

<li><a href=**"{{url('admin/show\_author')}}"**>**Show Author Images**</a></li>

<li><a href=**"{{url('admin/add\_author')}}"**>**Add Author Image**</a></li>

<li><a href=**"{{url('admin/show\_logs')}}"**>**Show Logs**</a></li>

<li><a href=**"{{url('/')}}"**>**Return to website**</a></li>

</ul>

</div>



## admin\_show\_author.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Show Author Images**

**@endsection**

**@section('admin\_part')**

<h2>**Author Images**</h2>

<table>

<tr class=**"fourcolumn"**>

<th>**Image**</th>

<th>**Is Author?**</th>

<th>**Edit**</th>

<th>**Delete**</th>

</tr>

**@foreach($author\_images as $img)**

<tr class=**"fourcolumn"**>

<td><img src=**"{{asset($img->author\_image)}}"** alt=**"Author Image {{$img->id}}"**/></td>

<td>**{{$img->is\_author?'true':'false'}}**</td>

<td><a href=**"{{url('admin/edit\_author/'.$img->id)}}"**><span class=**"fas fa-edit"**></span></a></td>

<td><a href=**"{{url('admin/delete\_author/'.$img->id)}}"**><span class=**"fa fa-trash"** aria-hidden=**"true"**></span></a></td>

</tr>

**@endforeach**

</table>

**@endsection**

## admin\_show\_categories.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Show Categories**

**@endsection**

**@section('admin\_part')**

<h2>**Categories**</h2>

<table>

<tr class=**"fourcolumn"**>

<th>**Category**</th>

<th>**Color**</th>

<th>**Edit**</th>

<th>**Delete**</th>

</tr>

**@foreach($categories as $category)**

<tr class=**"fourcolumn"**>

<td>**{{$category->name}}**</td>

<td style=**"color: {{$category->color}}"**>**{{$category->color}}**</td>

<td><a href=**"{{url('admin/edit\_category/'.$category->id)}}"**><span class=**"fas fa-edit"**></span></a></td>

<td><a href=**"{{url('admin/delete\_category/'.$category->id)}}"**><span class=**"fa fa-trash"** aria-hidden=**"true"**></span></a></td>

</tr>

**@endforeach**

</table>

**@endsection**



## admin\_show\_logs.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Logs**

**@endsection**

**@section('admin\_part')**

<h2>**Logs**</h2>

<table>

<tr class=**"fourcolumn"**>

<th>**IP**</th>

<th>**User**</th>

<th>**Detail**</th>

<th>**Date**</th>

</tr>

**@foreach($logs as $log)**

<tr class=**"fourcolumn"**>

<td>**{{$log->ip}}**</td>

<td>**{{$log->user}}**</td>

<td>**{{$log->detail}}**</td>

<td>**{{$log->date}}**</td>

</tr>

**@endforeach**

</table>

**@endsection**



## admin\_show\_users.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Show Users**

**@endsection**

**@section('admin\_part')**

<h2>**Users**</h2>

<table>

<tr class=**"fourcolumn"**>

<th>**Email**</th>

<th>**Role**</th>

<th>**Edit**</th>

<th>**Delete**</th>

</tr>

**@foreach($users as $user)**

<tr class=**"fourcolumn"**>

<td>**{{$user->email}}**</td>

<td>**{{$user->name}}**</td>

<td><a href=**"{{url('admin/edit\_user/'.$user->id)}}"**><span class=**"fas fa-edit"**></span></a></td>

<td><a href=**"{{url('admin/delete\_user/'.$user->id)}}"**><span class=**"fa fa-trash"** aria-hidden=**"true"**></span></a></td>

</tr>

**@endforeach**

</table>

**@endsection**



## admin\_add\_author.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Add Author Image**

**@endsection**

**@section('admin\_part')**

<h2>**Add Author Image**</h2>

<form action=**"{{ url("**/admin/add\_author") }}**" method="**POST" onsubmit=**"return admin\_add\_author()"** enctype=**"multipart/form-data"**>

**@csrf**

<div>

<label>**Image:**</label>

<input type=**"file"** name=**"authorimage"** id=**"authorimage"** accept=**".jpeg, .png, .jpg"**/>

</div>

<div>

<label>**Is Author?**</label>

<span>**True** <input type=**"radio"** id=**"isauthor"** name=**"isauthor"** value=**"true"** checked=**"checked"**/></span>

<span>**False** <input type=**"radio"** id=**"isnotauthor"** name=**"isauthor"** value=**"false"**/></span>

</div>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

</div>

</form>

**@endsection**



## admin\_add\_category.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Add Category**

**@endsection**

**@section('admin\_part')**

<h2>**Add Category**</h2>

<form action=**"{{ url("**/admin/add\_category") }}**" method="**POST" onsubmit=**"return addcategory\_check()"**>

**@csrf**

<div>

<label>**Name:**</label>

<input type=**"text"** name=**"categoryname"** id=**"categoryname"** placeholder=**"Name"**/>

</div>

<div>

<label>**Color:**</label>

<input type=**"color"** name=**"categorycolor"** id=**"categorycolor"**/>

</div>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

</div>

</form>

**@endsection**



## admin\_add\_user.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Add User**

**@endsection**

**@section('admin\_part')**

<h2>**Add User**</h2>

<form action=**"{{ url("**/admin/add\_user") }}**" method="**POST" onsubmit=**"return admin\_add\_user()"**>

**@csrf**

<div>

<label>**Email:**</label>

<input type=**"email"** name=**"email"** id=**"email"** placeholder=**"someone@something.com"**/>

</div>

<div>

<label>**Password:**</label>

<input type=**"password"** name=**"password"** id=**"password"** placeholder=**"\*\*\*\*\*\*"**/>

</div>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

</div>

</form>

**@endsection**



## admin\_edit\_author.blade.php

**@extends('layouts.admin')**

**@section('title\_part')**

**Admin Edit Author**

**@endsection**

**@section('admin\_part')**

<h2>**Edit Author Image**</h2>

<form action=**"{{ url("**/admin/edit\_author") }}**" method="**POST" enctype=**"multipart/form-data"**>

**@csrf**

<div>

<img src=**"{{asset($image->author\_image)}}"** alt=**"Author Image {{$image->id}}"**/>

</div>

<div>

<label>**Image:**</label>

<input type=**"file"** name=**"authorimage"** id=**"authorimage"** accept=**".jpeg, .png, .jpg"**/>

</div>

<div>

<label>**Is Author?**</label>

<span>**True** <input type=**"radio"** id=**"isauthor"** name=**"isauthor"** value=**"true"** {{$image->**is\_author ? 'checked' : ''}}/>**</span>

<span>**False** <input type=**"radio"** id=**"isnotauthor"** name=**"isauthor"** value=**"false"** {{$image->**is\_author ? '' : 'checked'}}/>**</span>

</div>

<input type=**"hidden"** value=**"{{$image->id}}"** name=**"id"**/>

<input type=**"hidden"** value=**"{{$image->author\_image}}"** name=**"lastImageUrl"**/>

<div>

<input type=**"submit"** name=**"btnSubmit"** value=**"Submit"**/>

<a href=**"{{url("**/admin/show\_author")}}**" class="**admin\_back\_link">**Back**</a>

</div>

</form>

**@endsection**



## Kernel.php

<?php

**namespace** App\Http;

**use** Illuminate\Foundation\Http\Kernel **as** HttpKernel;

**class** Kernel **extends** HttpKernel

{

/\*\*

\* The application's global HTTP middleware stack.

\*

\* These middleware are run during every request to your application.

\*

\* @var array

\*/

**protected** $middleware = [

\App\Http\Middleware\TrustProxies::**class**,

\App\Http\Middleware\CheckForMaintenanceMode::**class**,

\Illuminate\Foundation\Http\Middleware\ValidatePostSize::**class**,

\App\Http\Middleware\TrimStrings::**class**,

\Illuminate\Foundation\Http\Middleware\ConvertEmptyStringsToNull::**class**,

];

/\*\*

\* The application's route middleware groups.

\*

\* @var array

\*/

**protected** $middlewareGroups = [

'web' => [

\App\Http\Middleware\EncryptCookies::**class**,

\Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::**class**,

\Illuminate\Session\Middleware\StartSession::**class**,

// \Illuminate\Session\Middleware\AuthenticateSession::class,

\Illuminate\View\Middleware\ShareErrorsFromSession::**class**,

\App\Http\Middleware\VerifyCsrfToken::**class**,

\Illuminate\Routing\Middleware\SubstituteBindings::**class**,

\App\Http\Middleware\LogVisit::**class**,

],

'api' => [

'throttle:60,1',

\Illuminate\Routing\Middleware\SubstituteBindings::**class**,

],

];

/\*\*

\* The application's route middleware.

\*

\* These middleware may be assigned to groups or used individually.

\*

\* @var array

\*/

**protected** $routeMiddleware = [

'auth' => \App\Http\Middleware\Authenticate::**class**,

'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::**class**,

'bindings' => \Illuminate\Routing\Middleware\SubstituteBindings::**class**,

'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::**class**,

'can' => \Illuminate\Auth\Middleware\Authorize::**class**,

'guest' => \App\Http\Middleware\RedirectIfAuthenticated::**class**,

'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::**class**,

'signed' => \Illuminate\Routing\Middleware\ValidateSignature::**class**,

'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::**class**,

'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::**class**,

'loggedIn' => \App\Http\Middleware\LoggedInMiddleware::**class**,

'isAdmin' => \App\Http\Middleware\IsAdminMiddleware::**class**,

'logAdminVisit' => \App\Http\Middleware\LogAdminMiddeleware::**class**,

];

/\*\*

\* The priority-sorted list of middleware.

\*

\* This forces non-global middleware to always be in the given order.

\*

\* @var array

\*/

**protected** $middlewarePriority = [

\Illuminate\Session\Middleware\StartSession::**class**,

\Illuminate\View\Middleware\ShareErrorsFromSession::**class**,

\App\Http\Middleware\Authenticate::**class**,

\Illuminate\Routing\Middleware\ThrottleRequests::**class**,

\Illuminate\Session\Middleware\AuthenticateSession::**class**,

\Illuminate\Routing\Middleware\SubstituteBindings::**class**,

\Illuminate\Auth\Middleware\Authorize::**class**,

];

}



## AddCategoryRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** AddCategoryRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"categoryname" => "required",

"categorycolor" => "required"

];

}

}



## AuthorEditImageRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** AuthorEditImageRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"isauthor" => "required",

"authorimage" => 'mimes:jpeg,png,jpg',

"id" => "required",

"lastImageUrl" => "required"

];

}

}



## AuthorImageRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** AuthorImageRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"isauthor" => "required",

"authorimage" => 'required|mimes:jpeg,png,jpg'

];

}

}



## ContectRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** ContactRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"title" => "required|string|min:3|max:255",

"content" => "required|string|min:3|max:255"

];

}

}



## DeleteUserRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** DeleteUserRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"userid" => "required"

];

}

}



## EditUserRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** EditUserRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"email" => "required|email",

"id" => "required"

];

}

}



## GetNotesRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** GetNotesRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"category" => "required",

"page" => "required"

];

}

}



## LoginRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** LoginRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"email" => "required|email",

"password" => "required"

];

}

}



## NoteRequest.php

<?php

**namespace** App\Http\Requests;

**use** Illuminate\Foundation\Http\FormRequest;

**class** NoteRequest **extends** FormRequest

{

/\*\*

\* Determine if the user is authorized to make this request.

\*

\* @return bool

\*/

**public** **function** authorize()

{

**return** **true**;

}

/\*\*

\* Get the validation rules that apply to the request.

\*

\* @return array

\*/

**public** **function** rules()

{

**return** [

"note\_title" => "required",

"note\_text" => "required",

"note\_category" => "required"

];

}

}



## AuthorImage.php

<?php

**namespace** App\Models;

**class** AuthorImage {

**private** $table = 'author';

**public** **function** getAll() {

**return** \DB::table($this->table)

->select("\*")

->get();

}

**public** **function** getImageWithId($id) {

**return** \DB::table($this->table)

->select("\*")

->where([

["id", "=", $id]

])

->first();

}

**public** **function** addAuthorImage($author\_image, $is\_author) {

**return** \DB::table($this->table)

->insertGetId(['author\_image' => $author\_image, 'is\_author' => $is\_author]);

}

**public** **function** deleteAuthorImage($id) {

\DB::table($this->table)

->where('id', $id)

->**delete**();

}

**public** **function** editWithAuthorImage($author\_image, $is\_author, $id) {

\DB::table($this->table)

->where('id', $id)

->update(['author\_image' => $author\_image, 'is\_author' => $is\_author]);

}

**public function editIsAuthor($is\_author, $id) {**

**\DB::table($this->table)**

**->where('id', $id)**

**->update(['is\_author' => $is\_author]);**

**}**

**}**



## Category.php

<?php

**namespace** App\Models;

**class** Category {

**private** $table = 'category';

**public** **function** getCategories () {

**return** \DB::table($this->table)

->select("id", "name", "color")

->get();

}

**public** **function** getCategoryWithId($id) {

**return** \DB::table($this->table)

->select("id", "name", "color")

->where([

["id", "=", $id]

])

->first();

}

**public** **function** insertCategory($name, $color) {

**return** \DB::table($this->table)

->insertGetId(['name' => $name, 'color' => $color]);

}

**public** **function** editCategory($name, $color, $id) {

\DB::table($this->table)

->where('id', $id)

->update(['name' => $name, 'color' => $color]);

}

**public** **function** deleteCategory($id) {

\DB::table($this->table)

->where('id', $id)

->**delete**();

}

**}**



## LogAdmin.php

<?php

**namespace** App\Models;

**class** LogAdmin {

**private** $table = 'log';

**public** **function** getLogs () {

**return** \DB::table($this->table)

->select("ip", "user", "detail", "date")

->get();

}

**public** **function** insertLog($ip, $user, $detail, $date) {

**return** \DB::table($this->table)

->insertGetId(['ip' => $ip, 'user' => $user, 'detail' => $detail, 'date' => $date]);

}

}



## Note.php

<?php

**namespace** App\Models;

**class** Note {

**private** $table\_long = 'note AS n';

**private** $table\_short = 'note';

**public** **function** getNoteCount ($email) {

**return** \DB::table($this->table\_long)

->**join**("category AS c", "n.category\_id", "=", "c.id")

->**join**("user AS u", "n.user\_id", "=", "u.id")

->where([

["u.email", "=", $email]

])

->**count**();

}

**public** **function** getNoteWithId ($email, $id) {

**return** \DB::table($this->table\_long)

->**join**("category AS c", "n.category\_id", "=", "c.id")

->**join**("user AS u", "n.user\_id", "=", "u.id")

->select("n.title", "n.text", "n.id", "c.color", "c.name")

->where([

["u.email", "=", $email],

["n.id", "=", $id]

])

->first();

}

**public** **function** insertNote ($title, $text, $category, $user) {

**return** \DB::table($this->table\_short)

->insertGetId(['title' => $title, 'text' => $text, 'category\_id' => $category, 'user\_id' => $user]);

}

**public** **function** editNote ($id, $title, $text, $category) {

\DB::table($this->table\_short)

->where('id', $id)

->update(['title' => $title, 'text' => $text, 'category\_id' => $category]);

}

**public** **function** deleteNote ($id) {

**\DB::table($this->table\_short)**

**->where('id', $id)**

**->delete();**

**}**

**public function getNotes ($email) {**

**return \DB::table($this->table\_long)**

**->join("category AS c", "n.category\_id", "=", "c.id")**

**->join("user AS u", "n.user\_id", "=", "u.id")**

**->select("n.title", "n.text", "n.id", "c.color", "c.name")**

**->where([**

**["u.email", "=", $email]**

**])**

**->paginate(5);**

**}**

**public function getNotesWithCategory ($category, $email) {**

**return \DB::table($this->table\_long)**

**->join("category AS c", "n.category\_id", "=", "c.id")**

**->join("user AS u", "n.user\_id", "=", "u.id")**

**->select("n.title", "n.text", "n.id", "c.color", "c.name")**

**->where([**

**["u.email", "=", $email],**

**["c.id", "=", $category]**

**])**

**->paginate(5);**

**}**

**public function getNotesWithTitle ($title, $email) {**

**return \DB::table($this->table\_long)**

**->join("category AS c", "n.category\_id", "=", "c.id")**

**->join("user AS u", "n.user\_id", "=", "u.id")**

**->select("n.title", "n.text", "n.id", "c.color", "c.name")**

**->where([**

**["u.email", "=", $email],**

**["n.title", "like", '%'.$title.'%']**

**])**

**->paginate(5);**

**}**

**public function getNotesWithCategoryAndTitle ($category, $title, $email) {**

**return \DB::table($this->table\_long)**

**->join("category AS c", "n.category\_id", "=", "c.id")**

**->join("user AS u", "n.user\_id", "=", "u.id")**

**->select("n.title", "n.text", "n.id", "c.color", "c.name")**

**->where([**

**["u.email", "=", $email],**

**["c.id", "=", $category],**

**["n.title", "like", '%'.$title.'%']**

**])**

**->paginate(5);**

**}**

**}**



## User.php

<?php

**namespace** App\Models;

**class** User {

**private** $table\_long = 'user AS u';

**private** $table\_short = 'user';

**public** **function** getLogin($email, $password){

**return** \DB::table($this->table\_long)

->**join**("role AS r", "u.role\_id", "=", "r.id")

->select("r.name", "u.email", "u.id")

->where([

["email", "=", $email],

["password", "=", **md5**($password)]

])

->first();

}

**public** **function** emailTaken($email) {

**return** \DB::table($this->table\_short)

->select("email")

->where([

["email", "=", $email]

])

->first();

}

**public** **function** emailTakenWithId($email, $id) {

**return** \DB::table($this->table\_short)

->select("email")

->where([

["email", "=", $email],

["id", "!=", $id]

])

->first();

}

**public** **function** addUser($email, $password) {

\DB::table($this->table\_short)

->insert(['email' => $email, 'password' => **md5**($password)]);

**}**

**public function getUsers() {**

**return \DB::table($this->table\_long)**

**->join("role AS r", "u.role\_id", "=", "r.id")**

**->select("r.name", "u.email", "u.id")**

**->get();**

**}**

**public function deleteUser($id) {**

**\DB::table($this->table\_short)**

**->where('id', $id)**

**->delete();**

**}**

**public function getUserWithId($id) {**

**return \DB::table($this->table\_long)**

**->join("role AS r", "u.role\_id", "=", "r.id")**

**->select("r.name", "u.email", "u.id")**

**->where([**

**["u.id", "=", $id]**

**])**

**->first();**

**}**

**public function editUser($email, $id) {**

**\DB::table($this->table\_short)**

**->where('id', $id)**

**->update(['email' => $email]);**

**}**

**public function editUserWithPassword($email, $password, $id) {**

**\DB::table($this->table\_short)**

**->where('id', $id)**

**->update(['email' => $email, 'password' => md5($password)]);**

**}**

**}**



## IsAdminMiddleware.php

<?php

**namespace** App\Http\Middleware;

**use** Closure;

**class** IsAdminMiddleware

{

/\*\*

\* Handle an incoming request.

\*

\* @param \Illuminate\Http\Request $request

\* @param \Closure $next

\* @return mixed

\*/

**public** **function** handle($request, Closure $next)

{

**if**(**strtolower**($request->session()->get('user')->name) == "admin")

**return** $next($request);

**else**

**return** redirect("/")->with("message", "You don't have the right to visit this page!");

}

}



## LogAdminMiddleware.php

<?php

**namespace** App\Http\Middleware;

**use** App\Models\LogAdmin;

**use** Closure;

**class** LogAdminMiddeleware

{

/\*\*

\* Handle an incoming request.

\*

\* @param \Illuminate\Http\Request $request

\* @param \Closure $next

\* @return mixed

\*/

**public** **function** handle($request, Closure $next)

{

$model = **new** LogAdmin();

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

$ip = $request->ip();

$user = $request->session()->get('user')->email;

$detail = $request->getPathInfo();

$model->insertLog($ip, $user, $detail, $date);

**return** $next($request);

}

}



## LoggedInMiddleware.php

<?php

**namespace** App\Http\Middleware;

**use** Closure;

**class** LoggedInMiddleware

{

/\*\*

\* Handle an incoming request.

\*

\* @param \Illuminate\Http\Request $request

\* @param \Closure $next

\* @return mixed

\*/

**public** **function** handle($request, Closure $next)

{

**if**($request->session()->has("user"))

**if** ($request->getPathInfo() == "/login" || $request->getPathInfo() == "/register")

**return** redirect("/");

**else**

**return** $next($request);

**else**

**if** ($request->getPathInfo() == "/login" || $request->getPathInfo() == "/register")

**return** $next($request);

**else**

**return** redirect("/login")->with("message", "You must be logged in to continue!");

}

}



## LogVisit.php

<?php

**namespace** App\Http\Middleware;

**use** Closure;

**class** LogVisit

{

/\*\*

\* Handle an incoming request.

\*

\* @param \Illuminate\Http\Request $request

\* @param \Closure $next

\* @return mixed

\*/

**public** **function** handle($request, Closure $next)

{

// Storage -> Logs -> laravel.log

\**Log**::info("Visit time: ". **date**("Y-m-d H:i:s"). ", visit ip: ". $request->ip(). ", location visited: ". $request->fullUrl());

**return** $next($request);

}

}



## ContactController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\ContactRequest;

**use** Illuminate\Http\Request;

**use** PHPMailer\PHPMailer\PHPMailer;

**use** PHPMailer\PHPMailer\SMTP;

**use** PHPMailer\PHPMailer\Exception;

**class** ContactController **extends** Controller

{

**public** **function** index() {

**return** view('pages.contact');

}

**public** **function** contactUs(ContactRequest $request) {

$subject = $request->input("title");

$message = $request->input("content");

$to = $request->session()->get('user')->email;

$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 = PaziStaCitas;

$mail->SMTPSecure = 'tls';

$mail->Port = 587;

$mail->SMTPOptions = **array**(

'ssl' => **array**(

'verify\_peer' => **false**,

'verify\_peer\_name' => **false**,

'allow\_self\_signed' => **true**

)

**);**

**$mail->setFrom('twentyfivekitchen@gmail.com', 'Anotez');**

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

**$mail->addReplyTo($to, 'Visitor');**

**$mail->isHTML(true);**

**$mail->Subject = $subject;**

**$mail->Body = $message;**

**$mail->send();**

**return redirect("/contact")->with("message", "Message sent successfully.");**

**} catch (Exception $e) {**

**dd($e);**

**return redirect("/contact")->with("message", "Message sending failed.");**

**}**

**}**

**}**



## LoginController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\LoginRequest;

**use** App\Models\User;

**use** Illuminate\Http\Request;

**class** LoginController **extends** Controller

{

**protected** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** User();

}

**public** **function** login() {

**return** view('pages.login');

}

**public** **function** loginDo(LoginRequest $request) {

$email = $request->input("email");

$password = $request->input("password");

$user = $this->model->getLogin($email, $password);

**if** ($user) {

$request->session()->put("user", $user);

**return** redirect("/");

} **else** {

**return** redirect("/login")->with("message", "Wrong params.");

}

}

**public** **function** logout(Request $request) {

$request->session()->forget("user");

$request->session()->**flush**();

**return** redirect("/login");

}

}



## MainController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\GetNotesRequest;

**use** App\Models\Category;

**use** App\Models\Note;

**use** Illuminate\Http\Request;

**class** MainController **extends** Controller

{

**protected** $model;

**protected** $category\_model;

**public** **function** **\_\_construct**() {

$this->model = **new** Note();

$this->category\_model = **new** Category();

}

**public** **function** **main**(Request $request) {

$db\_data = $this->model->getNoteCount($request->session()->get('user')->email);

$cat\_data = $this->category\_model->getCategories();

$data = **array**();

$data['note\_count'] = $db\_data;

$data['categories'] = $cat\_data;

**return** view('pages.main', $data);

}

**public** **function** getNotes(GetNotesRequest $request) {

$category = $request->input("category");

$title = $request->input("title");

$email = $request->session()->get('user')->email;

**if** ($category == 0 && $title == **null**) {

$notes = $this->model->getNotes($email);

} **else** **if** ($title == **null**) {

$notes = $this->model->getNotesWithCategory($category, $email);

} **else** **if** ($category == 0) {

$notes = $this->model->getNotesWithTitle($title, $email);

} **else** {

**$notes = $this->model->getNotesWithCategoryAndTitle($category, $title, $email);**

**}**

**return $notes;**

**}**

**}**



## NoteController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\NoteRequest;

**use** App\Models\Category;

**use** App\Models\Note;

**use** Illuminate\Http\Request;

**class** NoteController **extends** Controller

{

**protected** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** Note();

}

**public** **function** index(Request $request, $id = 0) {

$category\_model = **new** Category();

$data = **array**();

**if** ($id != 0) {

$note = $this->model->getNoteWithId($request->session()->get('user')->email, $id);

**if** ($note) {

$data['note'] = $note;

} **else** {

**return** redirect("/")->with("message", "You don't have the right to see this note or the note does not exist.");

}

}

$category = $category\_model->getCategories();

$data['categories'] = $category;

**return** view('pages.note', $data);

}

**public** **function** editNote(NoteRequest $request) {

$category = $request->input("note\_category");

$title = $request->input("note\_title");

$text = $request->input("note\_text");

$id = $request->input("note\_id");

$note = $this->model->getNoteWithId($request->session()->get('user')->email, $id);

**if** (!$note) {

**return** redirect("/")->with("message", "You don't have the right to edit this note or the note does not exist.");

}

$this->model->editNote($id, $title, $text, $category);

**return** $this->index($request, $id);

}

**public** **function** addNote(NoteRequest $request) {

$title = $request->input("note\_title");

$text = $request->input("note\_text");

$category = $request->input("note\_category");

$lastInsertId = $this->model->insertNote($title, $text, $category, $request->session()->get('user')->id);

**return** $this->index($request, $lastInsertId);

}

**public** **function** deleteNote(Request $request, $id = 0) {

**if** ($id != 0) {

$note = $this->model->getNoteWithId($request->session()->get('user')->email, $id);

**if** ($note) {

$this->model->deleteNote($id);

**return** redirect("/");

} **else** {

**return** redirect("/")->with("message", "You don't have the right to delete this note or the note does not exist.");

}

} **else** {

**return** redirect("/")->with("message", "Note must exist.");

}

}

}



## RegisterController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\LoginRequest;

**use** App\Models\User;

**use** Illuminate\Http\Request;

**class** RegisterController **extends** Controller

{

**protected** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** User();

}

**public** **function** register() {

**return** view('pages.register');

}

**public** **function** registerDo(LoginRequest $request) {

$email = $request->input("email");

$password = $request->input("password");

$user = $this->model->emailTaken($email);

**if** ($user) {

**return** redirect("/register")->with("message", "Email already taken.");

} **else** {

$this->model->addUser($email, $password);

**return** redirect("/login")->with("message", "User created successfully.");

}

}

}



## AdminAuthorController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\AuthorEditImageRequest;

**use** App\Http\Requests\AuthorImageRequest;

**use** App\Models\AuthorImage;

**use** Illuminate\Http\Request;

**class** AdminAuthorController **extends** Controller

{

**private** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** AuthorImage();

}

**public** **function** showAuthor(Request $request) {

$data = [];

$data['author\_images'] = $this->model->getAll();

**return** view('admin.admin\_show\_author', $data);

}

**public** **function** addAuthor(Request $request) {

**return** view('admin.admin\_add\_author');

}

**public** **function** adminAddAuthor(AuthorImageRequest $request) {

\DB::beginTransaction();

$image = $request->**file**("authorimage");

$name = **time**()."\_".$image->getClientOriginalName();

$isUploaded = $image->move(public\_path()."/img/author", $name);

**if** (!$isUploaded) {

\DB::rollback();

**return** redirect("admin/show\_author");

}

$name\_full = "/img/author/".$name;

$is\_author = $request->input("isauthor");

$author = $is\_author == "true" ? 1 : 0;

**try** {

$lastId = $this->model->addAuthorImage($name\_full, $author);

**if** (!$lastId) {

\DB::rollback();

**return** redirect("admin/show\_author");

}

} **catch** (\Exception $ex) {

\DB::rollback();

**return** redirect("admin/show\_author");

}

\DB::commit();

**return** redirect("admin/show\_author");

}

**public** **function** editAuthor(Request $request, $id = 0) {

$data = [];

$data['image'] = $this->model->getImageWithId($id);

**if** ($data['image']) {

**return** view('admin.admin\_edit\_author', $data);

}

**return** redirect("admin/show\_author")->with("message", "You don't have the right to see this image or the image does not exist.");

}

**public** **function** adminEditAuthor(AuthorEditImageRequest $request) {

$is\_author = $request->input("isauthor");

$author = $is\_author == "true" ? 1 : 0;

$lastImageUrl = $request->input("lastImageUrl");

$id = $request->input("id");

$image = $request->**file**("authorimage");

**if** ($image) {

\DB::beginTransaction();

$name = **time**()."\_".$image->getClientOriginalName();

$isUploaded = $image->move(public\_path()."/img/author", $name);

**if** (!$isUploaded) {

\DB::rollback();

**return** redirect("admin/show\_author");

}

$name\_full = "/img/author/".$name;

**if**(\**File**::exists(public\_path().$lastImageUrl)) {

\**File**::**delete**(public\_path().$lastImageUrl);

}

**try** {

$this->model->editWithAuthorImage($name\_full, $author, $id);

\DB::commit();

} **catch** (\Exception $ex) {

\DB::rollback();

**return** redirect("admin/show\_author");

}

} **else** {

$this->model->editIsAuthor($author, $id);

}

**return** redirect("admin/show\_author");

}

**public** **function** deleteAuthor(Request $request, $id = 0) {

$image = $this->model->getImageWithId($id);

**if** ($image) {

**if**(\**File**::exists(public\_path().$image->author\_image)) {

\**File**::**delete**(public\_path().$image->author\_image);

$this->model->deleteAuthorImage($id);

}

**return** redirect("admin/show\_author");

}

**return** redirect("admin/show\_author")->with("message", "You don't have the right to see this image or the image does not exist.");

}

}



## AdminCategoryController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\AddCategoryRequest;

**use** App\Models\Category;

**use** Illuminate\Http\Request;

**class** AdminCategoryController **extends** Controller

{

**private** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** Category();

}

**public** **function** showCategory(Request $request) {

$data = [];

$data['categories'] = $this->model->getCategories();

**return** view('admin.admin\_show\_categories', $data);

}

**public** **function** addCategory(Request $request) {

**return** view('admin.admin\_add\_category');

}

**public** **function** adminAddCategory(AddCategoryRequest $request) {

$name = $request->input("categoryname");

$color = $request->input("categorycolor");

$this->model->insertCategory($name, $color);

**return** redirect("/admin/show\_category");

}

**public** **function** editCategory(Request $request, $id = 0) {

$data = [];

$data['category'] = $this->model->getCategoryWithId($id);

**if** ($data['category']) {

**return** view('admin.admin\_edit\_category', $data);

}

**return redirect("admin/show\_category")->with("message", "You don't have the right to see this category or the category does not exist.");**

**}**

**public function adminEditCategory(AddCategoryRequest $request) {**

**$name = $request->input("categoryname");**

**$color = $request->input("categorycolor");**

**$id = $request->input("id");**

**$this->model->editCategory($name, $color, $id);**

**return redirect("/admin/show\_category");**

**}**

**public function deleteCategory(Request $request, $id = 0) {**

**$category = $this->model->getCategoryWithId($id);**

**if ($category) {**

**$this->model->deleteCategory($id);**

**return redirect("admin/show\_category");**

**}**

**return redirect("admin/show\_category")->with("message", "You don't have the right to see this category or the category does not exist.");**

**}**

**}**



## AdminLogController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Models\LogAdmin;

**use** Illuminate\Http\Request;

**class** AdminLogController **extends** Controller

{

**private** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** LogAdmin();

}

**public** **function** showLogs(Request $request) {

$data = [];

$data['logs'] = $this->model->getLogs();

**return** view('admin.admin\_show\_logs', $data);

}

}



## AdminUserController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Http\Requests\DeleteUserRequest;

**use** App\Http\Requests\EditUserRequest;

**use** App\Http\Requests\LoginRequest;

**use** App\Models\User;

**use** Illuminate\Http\Request;

**class** AdminUserController **extends** Controller

{

**private** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** User();

}

**public** **function** showUsers(Request $request) {

$data = [];

$data['users'] = $this->model->getUsers();

**return** view('admin.admin\_show\_users', $data);

}

**public** **function** editUser(Request $request, $id = 0) {

$data = [];

$data['user'] = $this->model->getUserWithId($id);

**if** ($data['user']) {

**return** view('admin.admin\_edit\_user', $data);

}

**return** redirect("admin/show\_users")->with("message", "You don't have the right to see this user or the user does not exist.");

}

**public** **function** adminEditUser(EditUserRequest $request) {

$email = $request->input("email");

$password = $request->input("password");

$id = $request->input("id");

$user = $this->model->emailTakenWithId($email, $id);

**if** ($user) {

**return** redirect("/admin/edit\_user/".$id)->with("message", "Email already taken.");

} **else** {

**if** (!$password) {

$this->model->editUser($email, $id);

} **else** {

$this->model->editUserWithPassword($email, $password, $id);

}

**return** redirect("/admin/show\_users");

}

}

**public** **function** deleteUser(Request $request, $id = 0) {

$user = $this->model->getUserWithId($id);

**if** ($user) {

$this->model->deleteUser($id);

**return** redirect("admin/show\_users");

}

**return** redirect("admin/show\_users")->with("message", "You don't have the right to see this user or the user does not exist.");

}

**public** **function** addUser(Request $request) {

**return** view('admin.admin\_add\_user');

}

**public** **function** adminAddUser(LoginRequest $request) {

$email = $request->input("email");

$password = $request->input("password");

$user = $this->model->emailTaken($email);

**if** ($user) {

**return** redirect("/admin/add\_user")->with("message", "Email already taken.");

} **else** {

$this->model->addUser($email, $password);

**return** redirect("/admin/show\_users");

}

}

}



## AuthorController.php

<?php

**namespace** App\Http\Controllers;

**use** App\Models\AuthorImage;

**use** Illuminate\Http\Request;

**class** AuthorController **extends** Controller

{

**private** $model;

**public** **function** **\_\_construct**() {

$this->model = **new** AuthorImage();

}

**public** **function** index() {

**return** view('pages.author');

}

**public** **function** getImages() {

**return** $this->model->getAll();

}

}



## main.js

**function** check\_params**(**email**,** **password)** **{**

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

**if** **(**email **==** "" **||** **password** **==** ""**)** **{**

**document.**getElementById**(**"error\_text\_place"**).**innerHTML **=** "Email and password must not be empty"**;**

**return** **false;**

**}**

**if** **(!**email\_pattern**.**test**(**email**))** **{**

**document.**getElementById**(**"error\_text\_place"**).**innerHTML **=** "Email not valid"**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text\_place"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** login\_check**()** **{**

**var** email **=** **document.**getElementById**(**"email"**).**value**,**

**password** **=** **document.**getElementById**(**"password"**).**value**;**

**return** check\_params**(**email**,** **password);**

**}**

**function** register\_check**()** **{**

**var** email **=** **document.**getElementById**(**"email"**).**value**,**

**password** **=** **document.**getElementById**(**"password"**).**value**;**

**return** check\_params**(**email**,** **password);**

**}**

**function** note\_check**()** **{**

**var** title **=** **document.**getElementById**(**"note\_title"**).**value**,**

**text** **=** tinyMCE**.**get**(**'note\_text'**).**getContent**();**

**if** **(**title **==** '' **||** **text** **==** ''**)** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "None field should be left empty."**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** addcategory\_check**()** **{**

**var** name **=** **document.**getElementById**(**"categoryname"**).**value**,**

color **=** **document.**getElementById**(**"categorycolor"**).**value**;**

**if** **(**name **==** '' **||** color **==** ''**)** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "None field should be left empty."**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** admin\_add\_user**()** **{**

**var** email **=** **document.**getElementById**(**"email"**).**value**,**

**password** **=** **document.**getElementById**(**"password"**).**value**,**

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

**if** **(**email **==** "" **||** **password** **==** ""**)** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "Email and password must not be empty"**;**

**return** **false;**

**}**

**if** **(!**email\_pattern**.**test**(**email**))** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "Email not valid"**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** admin\_edit\_user**()** **{**

**var** email **=** **document.**getElementById**(**"email"**).**value**,**

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

**if** **(**email **==** ""**)** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "Email must not be empty"**;**

**return** **false;**

**}**

**if** **(!**email\_pattern**.**test**(**email**))** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "Email not valid"**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** admin\_add\_author**()** **{**

**var** authorimage **=** **document.**getElementById**(**"authorimage"**).**value**;**

**if** **(**authorimage **==** ""**)** **{**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** "Must select an image"**;**

**return** **false;**

**}**

**document.**getElementById**(**"error\_text"**).**innerHTML **=** ""**;**

**return** **true;**

**}**

**function** contact\_check**()** **{**

**var** title **=** **document.**getElementById**(**"title"**).**value**,**

content **=** **document.**getElementById**(**"content"**).**value**;**

**if** **(**title **==** "" **||** content **==** ""**)** **{**

$**(**".error\_text"**).**html**(**"None field must be left empty"**);**

**return** **false;**

**}**

$**(**".error\_text"**).**html**(**""**);**

**return** **true;**

**}**



## search.js

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

get\_notes**(**""**,** 0**,** 1**);**

$**(**"#search\_notes"**).**keyup**(function()** **{**

**var** search\_text **=** $**(**"#search\_notes"**).**val**(),**

category\_select **=** $**(**"#category\_select"**).**val**(),**

page **=** $**(**"#current\_page"**).**html**();**

**if** **(!**page**)** **{** page **=** 1**;** **}**

get\_notes**(**search\_text**,** category\_select**,** page**);**

**});**

$**(**"#category\_select"**).**change**(function()** **{**

**var** search\_text **=** $**(**"#search\_notes"**).**val**(),**

category\_select **=** $**(**"#category\_select"**).**val**(),**

page **=** $**(**"#current\_page"**).**html**();**

**if** **(!**page**)** **{** page **=** 1**;** **}**

get\_notes**(**search\_text**,** category\_select**,** page**);**

**});**

**});**

**function** get\_notes**(**title**,** category**,** page**)** **{**

$**.**ajax**({**

url**:** "/get\_notes"**,**

method**:** "GET"**,**

dataType**:** "json"**,**

data**:** **{**

title**:** title**,**

category**:** category**,**

page**:** page**,**

\_token**:** csrf

**},**

success**:** **function** **(**data**)** **{**

**if(!**data**.**data**.**length **&&** data**.**total **>** 0**)** **{**

get\_notes**(**title**,** category**,** page **-** 1**);**

**}** **else** **{**

fill\_note\_data**(**data**);**

**}**

**},**

error**:** **function** **(**err**)** **{**

$**(**".error\_text"**).**html**(**err**.**responseText**);**

**}**

**});**

**}**

**function** fill\_note\_data**(**data**)** **{**

**var** note\_holder **=** **document.**getElementById**(**"my\_notes"**),**

**text** **=** ""**;**

**if** **(**data**.**total **==** 0**)** **{**

**text** **=** `<h2 class="no\_notes\_text">You don't have any notes so far. You can add your first one by clicking + symbol.</h2>`**;**

**}** **else** **{**

**text** **=** `<ul class="my\_notes">`**;**

data**.**data**.**forEach**(function** **(**note**)** **{**

**var** note\_text **=** ""**;**

**if** **(**note**.text.**indexOf**(**"<li>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<li>"**)** **+** 4**,** note**.text.**indexOf**(**"</li>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<p>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<p>"**)** **+** 3**,** note**.text.**indexOf**(**"</p>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h1>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h1>"**)** **+** 4**,** note**.text.**indexOf**(**"</h1>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h2>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h2>"**)** **+** 4**,** note**.text.**indexOf**(**"</h2>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h3>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h3>"**)** **+** 4**,** note**.text.**indexOf**(**"</h3>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h4>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h4>"**)** **+** 4**,** note**.text.**indexOf**(**"</h4>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h5>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h5>"**)** **+** 4**,** note**.text.**indexOf**(**"</h5>"**));**

**}** **else** **if** **(**note**.text.**indexOf**(**"<h6>"**)** **>=** 0**)** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<h6>"**)** **+** 4**,** note**.text.**indexOf**(**"</h6>"**));**

**}** **else** **{**

note\_text **=** note**.text.**substring**(**note**.text.**indexOf**(**"<div>"**)** **+** 5**,** note**.text.**indexOf**(**"</div>"**));**

**}**

**text** **+=** `<li><div class="note\_category\_color" style="color: ${note.color};"><abbr title="${note.name}">&#8226;</abbr></div><div class="note\_info\_section"><a href="/note/${note.id}"><h3>${note.title}</h3><p>${note\_text}</p></a></div><div class="note\_delete\_area"><a href="/deletenote/'${note.id}"><span class="fa fa-trash" aria-hidden="true"></span></a></div></li>`**;**

**});**

**text** **+=** `</ul>`**;**

**text** **+=** `<ul id="note\_paging">`**;**

**for** **(var** i **=** 1**;** i **<=** data**.**last\_page**;** i**++)** **{**

**if** **(**i **==** data**.**current\_page**)** **{**

**text** **+=** `<li id="current\_page">${i}</li>`**;**

**}** **else** **{**

**text** **+=** `<li>${i}</li>`**;**

**}**

**}**

**text** **+=** `</ul>`**;**

**}**

note\_holder**.**innerHTML **=** **text;**

$**(**"#note\_paging li"**).**on**(**"click"**,** **function()** **{**

**if** **(**$**(this).**attr**(**"id"**)** **==** "current\_page"**)** **{** **return;** **}**

**var** search\_text **=** $**(**"#search\_notes"**).**val**(),**

category\_select **=** $**(**"#category\_select"**).**val**(),**

page **=** $**(this).**html**();**

get\_notes**(**search\_text**,** category\_select**,** page**);**

**});**

**}**



## author\_game.js

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

get\_images**();**

**});**

**function** get\_images**()** **{**

$**.**ajax**({**

url**:** "/author\_get\_images"**,**

method**:** "GET"**,**

dataType**:** "json"**,**

data**:** **{**

\_token**:** csrf

**},**

success**:** **function** **(**data**)** **{**

**var** score **=** **{**

correct**:** 0**,**

done**:** 0

**};**

fill\_author\_image**(**score**,** data**,** 0**);**

**},**

error**:** **function** **(**err**)** **{**

$**(**"#error\_text"**).**html**(**err**.**responseText**);**

**}**

**});**

**}**

**function** fill\_author\_image**(**score**,** author\_images**,** which\_image**)** **{**

**var** game\_section **=** **document.**getElementById**(**"author\_game"**);**

**if** **(**score**.**done **==** author\_images**.**length**)** **{**

**var** percentage **=** **(**score**.**correct **/** score**.**done**)** **\*** 100**;**

game\_section**.**innerHTML **=** `<h3>Congratulations!</h3>

<p>You had ${percentage}% correct answers.</p>

<button id="try\_again">Try Again?</button>`**;**

$**(**"#try\_again"**).**on**(**"click"**,** **function()** **{**

score**.**correct **=** 0**;**

score**.**done **=** 0**;**

fill\_author\_image**(**score**,** author\_images**,** 0**);**

**});**

**}** **else** **{**

game\_section**.**innerHTML **=** `<h3>Can you guess if author is on the image?</h3>

<p>Try your luck and see the score...</p>

<img src="${author\_images[which\_image].author\_image}" alt="Author Image ${author\_images[which\_image].id}"/>

<div class="author\_game\_buttons"><button id="yes\_button">Yes</button><button id="no\_button">No</button></div>`**;**

$**(**"#yes\_button"**).**on**(**"click"**,** **function()** **{**

author\_button\_click**(**score**,** author\_images**,** which\_image**,** **true);**

**});**

$**(**"#no\_button"**).**on**(**"click"**,** **function()** **{**

author\_button\_click**(**score**,** author\_images**,** which\_image**,** **false);**

**});**

**}**

**}**

**function** author\_button\_click**(**score**,** author\_images**,** which\_image**,** yes\_no**)** **{**

**var** round\_result **=** **(**author\_images**[**which\_image**].**is\_author **==** yes\_no**);**

score**.**done**++;**

**if(**round\_result**)** **{** score**.**correct**++;** **}**

fill\_author\_image**(**score**,** author\_images**,** which\_image **+** 1**);**

**}**



## style.css

\* **{**

**margin: 0;**

**padding: 0;**

**box-sizing: border-box;**

**font-family:** 'ABeeZee'**;**

**}**

html **{**

**background-color: rgb(242, 242, 242);**

**}**

input**,** select **{**

**background-color: rgb(242, 242, 242);**

**}**

**.**error\_text **{**

**margin-top: 10px;**

**color: #FF0000;**

**}**

p**.**add\_space\_below **{**

**margin-bottom: 40px;**

**}**

input**:focus,** textarea**:focus,** select**:focus** **{**

**border-color: #FF8000;**

**outline: none;**

**}**

**.**documentation\_holder **{**

**font-size: 40px;**

**color: #000;**

**}**

**.**documentation\_holder**:hover** **{**

**color: #FF8000;**

**}**

/\*Header\*/

**.**header\_holder **{**

**width: 100%;**

**height: 50px;**

**display: flex;**

**justify-content: space-between;**

**align-items: center;**

**position: fixed;**

**}**

**.**header\_holder**,** **.**footer\_data **{**

**background-color: rgb(233, 233, 233);**

**}**

**.**header\_holder a **{**

**color: #000;**

**text-decoration: none;**

**}**

.header\_holder .login\_logout\_button {

font-size: 18px;

padding: 5px 15px;

}

.header\_holder .search\_form input[type="submit"] {

display: none;

}

/\*Footer\*/

.footer\_data {

position: fixed;

bottom: 0;

width: 300px;

margin-left: calc(50% - 150px);

}

.footer\_data a {

text-decoration: none;

color: #000;

}

.footer\_data a p {

text-align: center;

padding: 5px 0;

}

/\*Notes Filter\*/

.notes\_filter {

width: 90%;

margin: 10px 0;

display: flex;

justify-content: space-around;

}

.notes\_filter .search\_notes {

width: 70px;

border: 1px solid #ccc;

border-radius: 4px;

font-size: 14px;

background-color: #FFFFFF;

background-image: url('img/searchicon.png');

background-repeat: no-repeat;

padding: 4px 6px 4px 25px;

-webkit-transition: width 0.4s ease-in-out;

transition: width 0.4s ease-in-out;

}

.notes\_filter .search\_notes:focus {

width: 130px;

}

.notes\_filter select {

align-self: flex-start;

padding: 5px 10px;

border-radius: 50%;

-webkit-appearance: none;

-moz-appearance: none;

font-size: 14px;

text-align-last: center;

}

/\*Middle Part\*/

.middle\_part {

padding-top: 80px;

display: flex;

align-items: center;

flex-direction: column;

}

.middle\_part .my\_notes {

list-style-type: none;

width: 98%;

}

.middle\_part .admintitle {

margin-top: 30px;

}

.middle\_part .my\_notes li {

margin-top: 15px;

border: 1px solid #ccc;

padding: 10px 5px;

border-radius: 10px;

color: #000;

text-decoration: none;

display: flex;

justify-content: space-between;

}

.middle\_part .my\_notes li:hover {

border: 1px solid #FF8000;

}

.middle\_part .my\_notes li .note\_category\_color {

width: 10%;

display: flex;

justify-content: center;

align-items: center;

font-size: 30px;

user-select: none;

}

.middle\_part .my\_notes li .note\_category\_color abbr {

text-decoration: none;

}

.middle\_part .my\_notes li .note\_info\_section {

width: 78%;

}

.middle\_part .my\_notes li .note\_info\_section a {

color: #000;

text-decoration: none;

}

.middle\_part .my\_notes li .note\_info\_section h3, .middle\_part .my\_notes li .note\_info\_section p {

overflow: hidden;

text-overflow: ellipsis;

white-space: nowrap;

}

.middle\_part .my\_notes li .note\_delete\_area {

width: 10%;

display: flex;

justify-content: center;

align-items: center;

}

.middle\_part .my\_notes li .note\_delete\_area a {

color: #000;

}

.middle\_part form {

width: 98%;

display: flex;

align-items: center;

flex-direction: column;

}

.middle\_part form div.divform.addedittitle {

margin-bottom: 15px;

}

.middle\_part form div.divform {

margin-top: 15px;

width: 100%;

display: flex;

justify-content: center;

align-items: center;

}

.middle\_part form div input {

padding: 5px 10px;

border-top: none;

border-right: none;

}

.middle\_part form div input[type="submit"] {

cursor: pointer;

}

.middle\_part form div input[type="email"], .middle\_part form div input[type="password"] {

width: 60%;

}

.middle\_part form div label {

width: 20%;

text-align: center;

}

.middle\_part .login\_register\_link {

color: #000;

text-align: center;

text-decoration: none;

margin-top: 50px;

margin-bottom: 60px;

}

.middle\_part .login\_register\_link:hover {

text-decoration: underline;

}

.middle\_part #my\_notes {

width: 100%;

display: flex;

flex-direction: column;

align-items: center;

}

.middle\_part #note\_paging {

margin-top: 15px;

margin-bottom: 15px;

width: 96%;

margin-left: 2%;

display: flex;

flex-wrap: wrap;

justify-content: space-around;

list-style-type: none;

}

.middle\_part #note\_paging li {

background-color: rgb(213, 213, 213);

padding: 5px;

border-bottom: 2px solid #888888;

border-left: 1px solid rgba(88, 88, 88, 0.1);

cursor: pointer;

}

.middle\_part #note\_paging li:hover {

color: #FF0000;

}

.middle\_part #note\_paging li#current\_page {

opacity: 0.4;

cursor: initial;

color: #000;

user-select: none;

}

.middle\_part .no\_notes\_text {

margin-top: 20px;

text-align: center;

font-style: italic;

}

.middle\_part .note\_form {

margin-top: 10px;

width: 98%;

display: flex;

flex-direction: column;

}

.middle\_part .note\_form div.divform {

display: flex;

justify-content: space-between;

}

.middle\_part .note\_form div input[type="color"] {

align-self: flex-start;

}

.middle\_part .note\_form div label {

width: 30%;

}

.middle\_part .note\_form div input[type="text"] {

width: 70%;

}

.middle\_part .note\_form textarea {

width: 100%;

margin-top: 10px;

padding: 5px;

}

.middle\_part .note\_form div input[type="submit"], .middle\_part .note\_form a {

padding: 5px 10px;

border-bottom: 2px solid #888888;

border-left: 1px solid rgba(88, 88, 88, 0.1);

font-size: 16px;

color: #000;

}

.middle\_part .note\_form a {

text-decoration: none;

margin-left: 10px;

}

.middle\_part .note\_form select {

align-self: flex-start;

padding: 5px 10px;

border-radius: 50%;

-webkit-appearance: none;

-moz-appearance: none;

font-size: 14px;

text-align-last: center;

}

.middle\_part .admin\_panel\_link {

margin-top: 10px;

text-align: center;

color: #000;

text-decoration: none;

margin-bottom: 70px;

}

.middle\_part .author\_center {

text-align: center;

}

.middle\_part .author\_game {

margin-top: 20px;

display: flex;

flex-direction: column;

align-items: center;

}

.middle\_part .author\_error\_text {

margin-bottom: 70px;

}

.middle\_part .author\_error\_text a {

text-decoration: none;

}

.middle\_part .author\_error\_text a:hover {

color: #FF0000;

}

.middle\_part .author\_game img {

margin-top: 10px;

width: 30%;

}

.middle\_part .author\_game button {

margin-top: 10px;

padding: 5px 15px;

background-color: rgb(233, 233, 233);

cursor: pointer;

}

.middle\_part .author\_game .author\_game\_buttons {

width: 60%;

display: flex;

justify-content: space-around;

}

.middle\_part .author\_game h3, .middle\_part .author\_game p {

text-align: center;

}

.middle\_part .contact\_form {

margin-top: 15px;

}

.middle\_part .contact\_form div {

margin-top: 10px;

display: flex;

align-items: center;

justify-content: space-around;

width: 80%;

}

.middle\_part .contact\_form input[type="text"], .middle\_part .contact\_form textarea {

margin: 10px 0;

border-top: none;

border-right: none;

background-color: rgb(254, 254, 254);

padding: 10px;

width: 80%;

}

.middle\_part .contact\_form a {

padding: 5px 10px;

border-top: none;

border-right: none;

text-decoration: none;

border-bottom: 2px solid #888888;

border-left: 1px solid rgba(88, 88, 88, 0.1);

color: #000;

}

.button\_add\_note {

position: fixed;

z-index: 10;

bottom: 50px;

right: 5px;

padding: 5px;

background-color: #FF8000;

border-radius: 50%;

}

.button\_add\_note a {

color: #000;

background-color: #FF8000;

text-decoration: none;

font-size: 35px;

border-radius: 50%;

}

/\*Admin Panel\*/

.admin\_part {

padding-top: 80px;

display: flex;

justify-content: space-around;

flex-direction: row;

}

.admin\_part .error\_text {

margin-bottom: 70px;

}

.admin\_part .admin\_navigation {

width: 30%;

background-color: rgb(233, 233, 233);

padding: 5px;

}

.admin\_part .admin\_back\_link {

text-decoration: none;

color: #000;

}

.admin\_part .admin\_navigation h2, .admin\_part .admin\_navigation ul, .admin\_part .admin\_navigation ul li, .admin\_part .admin\_navigation ul li a {

background-color: rgb(233, 233, 233);

}

.admin\_part .admin\_navigation h2 {

margin-bottom: 20px;

}

.admin\_part .admin\_navigation ul {

list-style-type: none;

}

.admin\_part .admin\_navigation ul li {

margin: 5px 0;

}

.admin\_part .admin\_navigation ul li a {

text-decoration: none;

color: #000;

}

.admin\_part .admin\_navigation ul li a:hover {

color: #FF8000;

}

.admin\_part .admin\_layout\_content {

width: 67%;

}

.admin\_part .admin\_layout\_content h2 {

text-align: center;

}

.admin\_part .admin\_layout\_content table {

width: 100%;

border-spacing: 0;

display: block;

overflow-x: auto;

margin-bottom: 70px;

}

.admin\_part .admin\_layout\_content table tr {

width: 100%;

}

.admin\_part .admin\_layout\_content table tr:nth-child(2n) {

background-color: rgb(225, 225, 225);

}

.admin\_part .admin\_layout\_content table tr:hover {

background-color: rgb(150, 150, 150);

}

.admin\_part .admin\_layout\_content table th {

padding: 5px;

}

.admin\_part .admin\_layout\_content table td {

padding: 2px;

}

.admin\_part .admin\_layout\_content table th, .admin\_part .admin\_layout\_content table td {

text-align: center;

}

.admin\_part .admin\_layout\_content table td img {

width: 50%;

}

.admin\_part .admin\_layout\_content table .fourcolumn th, .admin\_part .admin\_layout\_content table .fourcolumn td {

width: 25%;

}

.admin\_part .admin\_layout\_content table .threecolumn th, .admin\_part .admin\_layout\_content table .threecolumn td {

width: 25%;

}

.admin\_part .admin\_layout\_content table a {

color: #000;

}

.admin\_part .admin\_layout\_content form {

margin-top: 20px;

display: flex;

flex-direction: column;

margin-bottom: 70px;

}

.admin\_part .admin\_layout\_content form div {

display: flex;

justify-content: space-between;

}

.admin\_part .admin\_layout\_content form input {

margin-bottom: 10px;

border-top: none;

border-right: none;

}

.admin\_part .admin\_layout\_content form input[type="email"], .admin\_part .admin\_layout\_content form input[type="password"], .admin\_part .admin\_layout\_content form input[type="text"], .admin\_part .admin\_layout\_content form input[type="color"] {

width: 80%;

}

.admin\_part .admin\_layout\_content form input[type="submit"] {

cursor: pointer;

}

.admin\_part .admin\_layout\_content form input, .admin\_part .admin\_layout\_content form label {

padding: 5px;

}

.admin\_part form div img {

margin-left: 35%;

width: 30%;

}

/\*Media Queries\*/

@media screen and (max-width: 500px) {

.admin\_part {

flex-direction: column;

align-items: center;

}

.admin\_part .admin\_navigation, .admin\_part .admin\_layout\_content {

width: 99%;

}

}

@media screen and (max-width: 400px) {

.middle\_part form div {

flex-direction: column;

align-items: center;

}

.middle\_part .note\_form a {

margin-left: 0;

margin-top: 10px;

}

.middle\_part form div input[type="email"], .middle\_part form div input[type="password"], .middle\_part form div label {

width: 90%;

}

}

@media screen and (max-width: 320px) {

.header\_holder {

flex-direction: column;

height: 100px;

}

.header\_holder .login\_logout\_button {

margin-top: 5px;

margin-bottom: 5px;

}

.footer\_data {

width: 200px;

margin-left: calc(50% - 100px);

}

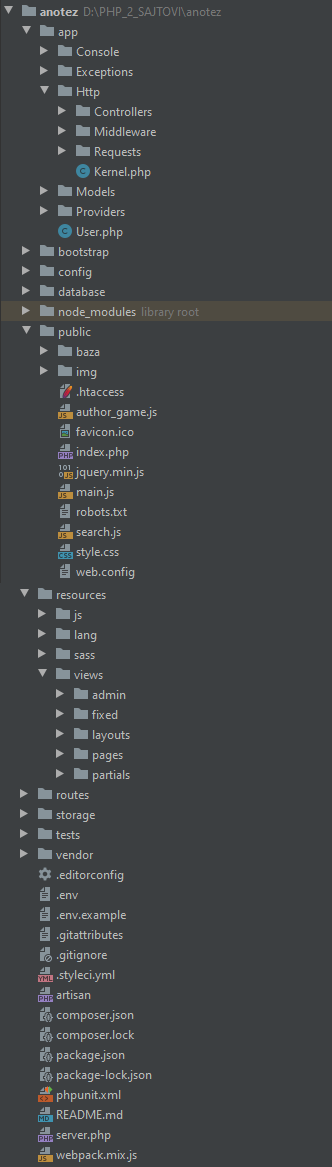
.middle\_part, .admin\_part {

padding-top: 130px;

}

}

# HIJERARHIJSKI PREGLED

****