VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE I KOMUNIKACIONE TEHNOLOGIJE



**WEB PROGRAMIRANJE PHP2**

**DOKUMENTACIJA PROJEKTA**

Student:

Marko Lazić 59/14

Beograd 2019.

SADRŽAJ

**Contents**

1.Opis funkcionalnosti ............................................................................................................... 1

2. Skica struktura stranica ..........................................................................................................

1

[3. Dijagram baze podataka 2](#_Toc3823900)

[3.MVC Organizacija 5](#_Toc3823901)

[Kontroleri 5](#_Toc3823902)

[Modeli 6](#_Toc3823903)

[View 8](#_Toc3823904)

[4. javaScript kod 9](#_Toc3823905)

[5. PHP kod 10](#_Toc3823906)

[5.1 GalleryController.php 10](#_Toc3823907)

[5.2 Navigation Controller.php 14](#_Toc3823908)

[5.3 PostController.php 17](#_Toc3823909)

[5.4 CommentsController.php 20](#_Toc3823910)

[5.5 RoleController.php 21](#_Toc3823911)

[5.6 UsersController.php 24](#_Toc3823912)

[5.7 FrontendController.php 28](#_Toc3823913)

[5.8 LoginController.php 31](#_Toc3823914)

[5.9 AdminMiddleware.php 34](#_Toc3823915)

[5.10 UserMiddleware.php 35](#_Toc3823916)

[5.11 Gallery.php 35](#_Toc3823917)

[5.12 Korisnik.php 37](#_Toc3823918)

[5.13 Meni.php 40](#_Toc3823919)

[5.14 Post.php 41](#_Toc3823920)

[5.15 CommentsModel.php 44](#_Toc3823921)

[5.16 Slika.php 45](#_Toc3823922)

[5.17 Uloga.php 46](#_Toc3823923)

[5.18 admin.components.common.footer.blade.php 48](#_Toc3823924)

[5.19 admin.components.common.head.blade.php 48](#_Toc3823925)

[5.20 admin.components.common.scripts.blade.php 49](#_Toc3823926)

[5.21 admin.components.common.sidenav.blade.php 49](#_Toc3823927)

[5.22 admin.components.common.topnav.blade.php 51](#_Toc3823928)

[5.23 admin.components.gallery.table.blade.php 51](#_Toc3823929)

[5.24 admin.components.gallery.insert\_form.blade.php 52](#_Toc3823930)

[5.25 admin.components.gallery.edit\_form.blade.php 54](#_Toc3823931)

[5.26 admin.components.navigation.table.blade.php 56](#_Toc3823932)

[5.27 admin.components.navigation.insert\_form.blade.php 57](#_Toc3823933)

[5.28 admin.components.navigation.edit\_form.blade.php 58](#_Toc3823934)

[5.29 admin.components.posts.post\_tab.blade.php 60](#_Toc3823935)

[5.30 admin.components.posts.insert\_form.blade.php 61](#_Toc3823936)

[5.31 admin.components.posts.edit\_form.blade.php 63](#_Toc3823937)

[5.32 admin.components.roles.table.blade.php 66](#_Toc3823938)

[5.33 admin.components.roles.insert\_form.blade.php 66](#_Toc3823939)

[5.34 admin.components.roles.edit\_form.blade.php 67](#_Toc3823940)

[5.35 admin.components.users.table.blade.php 69](#_Toc3823941)

[5.36 admin.components.users.insert\_form.blade.php 69](#_Toc3823942)

[5.37 admin.components.users.edit\_form.blade.php 71](#_Toc3823943)

[5.38 admin.components.pages.gallery.blade.php 72](#_Toc3823944)

[5.39 admin.components.pages.navigation.blade.php 73](#_Toc3823945)

[5.40 admin.components.pages.posts.blade.php 74](#_Toc3823946)

[5.41 admin.components.pages.roles.blade.php 75](#_Toc3823947)

[5.42 admin.components.pages.users.blade.php 76](#_Toc3823948)

[5.43 components.footer.blade.php 76](#_Toc3823949)

[5.44 components.header.blade.php 77](#_Toc3823950)

[5.45 layouts.admin.blade.php 78](#_Toc3823951)

[5.46 layouts.front.blade.php 80](#_Toc3823952)

[5.47 pages.autor.blade.php 84](#_Toc3823953)

[5.48 pages.createPost.blade.php 84](#_Toc3823954)

[5.49 pages.gallery.blade.php 86](#_Toc3823955)

[5.50 pages.home.blade.php 87](#_Toc3823956)

[5.51 pages.register.blade.php 91](#_Toc3823957)

[5.52 pages.single.blade.php 93](#_Toc3823958)

[5.53 web.php 94](#_Toc3823959)

**1.Opis funkcionalnosti**

U okviru ove sekcije navedite detaljan opis funkcionalnosti u formi kratke priče ili kreirajte *use case* dijagram.

**-Logovanje i zaštita sesijama**: Logovanje se nalazi u sidebaru sa desne strane i pristupa mu se klikom na button login. Proverava se regularnim izrazima, ukoliko korisnik postoji pristupa se delu sajta za autorizovane korisnike koji mogu da dodaju svoj tutorijal. **-Registracija**:

Korisnik popunjava svoje podatke a zatim se vrši provera regularnim izrazima. Ukoliko je sve dobro popunjeno korisnik će biti obavešten o uspešnoj registraciji i potom će moći da se uloguje.

**Pregled tutorijala:** Autorizovani korisnici imaju pravo da pristupe svakom tutorijalu ponaosob, neautorizovanim korisnicima je to uskraćeno.

**Straničenje**: Nalazi se na početnoj strani gde se prikazuju 2 tutorijala po stranici.

**Kontrola komentara**: Korisnik moze da ostavi nov komentar na postu, edituje I obriše.

**Potpuna kontrola sadržaja sajta:** Korisnik sa ulogom admin moţe da pregleda, doda, aţurira i obriše bilo koji sadrţaj na sajtu u admin panelu. Admin deo je zaštićen preko middleware.

|  |  |
| --- | --- |
| **Koriščeni jezici** | * HTML 4 i HTML 5 * CSS * javaScript PHP. |
| **Korišćeni HTML šabloni (framework-ci)** | [https://startbootstrap.com/templateoverviews/sbadmin/](https://startbootstrap.com/template-overviews/sb-admin/)    template za admin panel    [https://w3layouts.com/floral-a-blogging-categoryflatbootstrap-responsive-web-template/](https://w3layouts.com/floral-a-blogging-category-flat-bootstrap-responsive-web-template/)    template za ostale stranice |
| **Koriščene biblioteke i njihovi urlovi odakle ste ih preuzeli** | * jQuery sa jquery.com      * js/lightbox-plus-jquery.min.js sa * <http://lokeshdhakar.com/projects/lightbox2/> * https://github.com/rap2hpoutre/laravel-log-viewer |
| **Korišćene biblioteke iz framework-a** | / |
| **Korišćene php biblioteke van fremework-a** | / |
| **Koje funkcionalnosti su realizove putem AJAX-a** | / |

1

|  |  |
| --- | --- |
| **Navesti lokacije gde je napisan AJAX kod (fajlovi ili delovi html stranice)** | / |
| **Navesti lokacije gde je napisan sopstveni javaScript kod (fajlovi**  **ili delovi html stranice)** | Resources/views/pages/register.blade.php 47-75 linija koda |
| **URL sajta/aplikacije** | http://187sajt.byethost31.com/ |
| **Pristupni parametri** | Parametri su poslati mejlom |

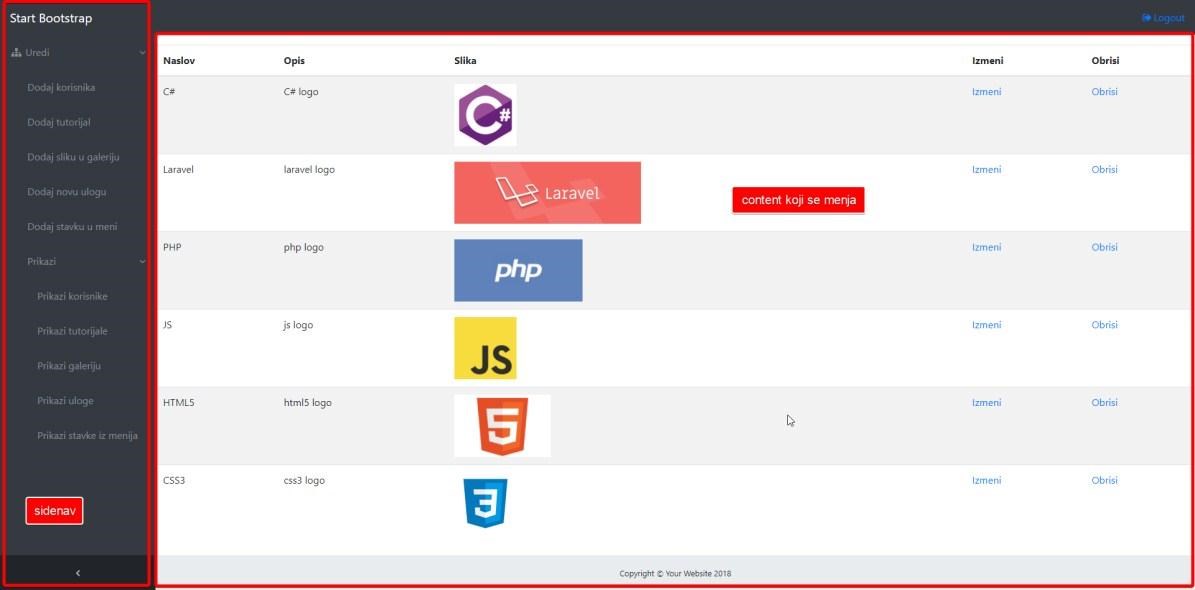
2

**2. Skica struktura stranica**

U okviru ove sekcije potrebno je navesti skice osnovnih tipova strana koje se koriste u okviru web sajta/aplikacije. Da bi kreirali skice moţete koristiti alatku *Pencil* ili bilo koji drugi alat za kreiranje *wireframe*-a. Takođe moguće je odraditi i *print screen* strukture stranice. Template za sve strane sem admin panela



Template za admin panel



1

# 3. Dijagram baze podataka

Tabela korisnik



Tabela uloga



Tabela za meni



Tabela galerija



Tabela post

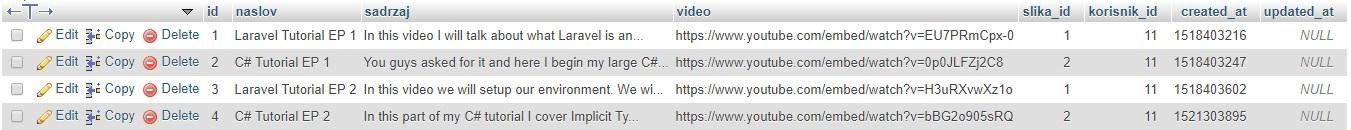
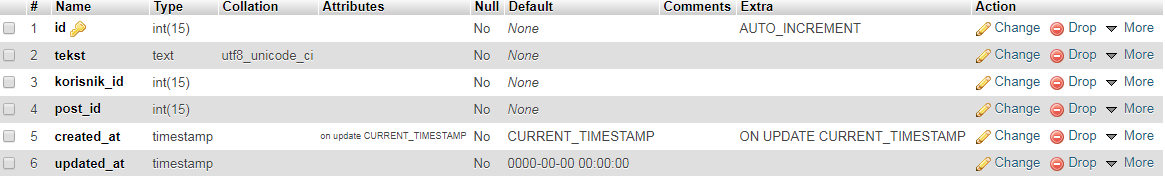


Tabela za slike



Tabela za komentare



# 3.MVC Organizacija

# Kontroleri

|  |
| --- |
| GalleryController |
| Index() |
| Create() |
| Store(Request $request) |
| Destroy($id) |
| show($id) |
| Update(Request $request,$id) |

|  |
| --- |
| NavigationController |
| Index() |
| Create() |
| Store(Request $request) |
| Destroy($id) |
| show($id) |
| Update(Request $request,$id) |

|  |
| --- |
| PostController |
| Index() |
| Create() |
| Store(Request $request) |
| Destroy($id) |
| show($id) |
| Update(Request $request,$id) |

|  |
| --- |
| RoleController |
| Index() |
| Create() |
| Store(Request $request) |
| Destroy($id) |
| show($id) |
| Update(Request $request,$id) |

UsersController

|  |
| --- |
| Index() |
| Create() |
| Store(Request $request) |
| Destroy($id) |
| show($id) |
| Update(Request $request,$id) |

|  |
| --- |
| LoginController |
| Login() |
| Logout() |
| registerStore(Request $request) |

|  |
| --- |
| CommentsController |
| postComment((Request $request,$postId) |
| showEditForm((Request $request,$commentId) |
| update(Request $request,$commentId)  delete($id) |

|  |
| --- |
| FrontendController |
| Index() |
| showRegistrationForm() |
| Single($id) |
| Gallery() |
| Autor() |
| createPost() |
| storePost() |

# Modeli

|  |
| --- |
| Gallery |
| All() |
| Save() |
| Find($id) |
| Delete($id) |
| Update($id) |

|  |
| --- |
| Korisnik |
| getAll() |
| Get() |
| selectOne() |
| getByUsernameAndPassword() |
| Update($id) |
| Save() |
| Delete($id) |

|  |
| --- |
| Meni |
| getAll() |
| Save() |
| Delete($id) |
| Update($id) |

|  |
| --- |
| Post |
| getAll() |
| getAll2() |
| get() |
| Get2() |
| Update($id) |
| Insert($id) |
| Save() |

|  |
| --- |
| CommentsModel |
| save() |
| getSingleComment($commentId) |
| update($id) |
| getPostComments($postId) |
| delete($id) |
| find($id) |

|  |
| --- |
| Slika |
| save() |
| Save2() |
| Delete($id) |
| find($id) |

|  |
| --- |
| Meni |
| SelectAll() |
| insert() |
| Delete($id) |
| Update($id) |
| selectOne($id) |

# View

|  |
| --- |
| Spisak view fajlova |
| layouts.front.blade.php |
| Layouts.admin.blade.php |
| pages.home.blade.php |
| Pages.single.blade.php |
| Pages.autor.blade.php |
| Pages.gallery.blade.php |
| Pages.register.blade.php |
| Pages.createPost.blade.php |
| Components.header.blade.php |
| Components.forma\_za\_komentar.blade.php |
| Components.komentari.blade.php |
| Components.footer.blade.php |
| Admin.pages.gallery.blade.php |
| Admin.pages.navigation.blade.php |
| Admin.pages.posts.blade.php |
| Admin.pages.roles.blade.php |
| Admin.pages.users.blade.php |
| Admin.components.common.footer.blade.php |
| Admin.components.common.head.blade.php |
| Admin.components.common.scripts.blade.php |
| Admin.components.common.sidenav.blade.php |
| Admin.components.common.topnav.blade.php |
| Admin.components.gallery.table.blade.php |
| Admin.components.gallery.insert\_form.blade.php |
| Admin.components.gallery.edit\_form.blade.php |
| Admin.components.navigation.table.blade.php |
| Admin.components.navigation.insert\_form.blade.php |
| Admin.components.navigation.edit\_form.blade.php |
| Admin.components.posts.post\_tab.blade.php |
| Admin.components.posts.insert\_form.blade.php |
| Admin.components.posts.edit\_form.blade.php |
| Admin.components.roles.table.blade.php |
| Admin.components.roles.insert\_form.blade.php |
| Admin.components.roles.edit\_form.blade.php |
| Admin.components.users.table.blade.php |
| Admin.components.users.insert\_form.blade.php |
| Admin.components.users.edit\_form.blade.php |

# 4. javaScript kod

<script type="text/javascript"> function provera(){ var regKorIme = /^[A-Za-z]{3,15}$/; var regEmail= /^[\w\d]{2,20}(\.+[\w\d]{2,20})\*\@([a-z]{2,6}\.)+[a-z]{2,6}$/; var regPassword=/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])[0-9a-zA-Z]{8,}$/

var korIme=document.getElementById("korisnickoIme").value; var email=document.getElementById("email").value; var lozinka=document.getElementById("lozinka").value;

if (regKorIme.test(korIme)) { document.getElementById("korisnickoIme").style.border = "2px solid green"; }

else {

document.getElementById("korisnickoIme").style.border = "2px solid red"; }

if (regEmail.test(email)) {

document.getElementById("email").style.border = "2px solid green"; }

else {

document.getElementById("email").style.border = "2px solid red"; }

if (regPassword.test(lozinka)) { document.getElementById("lozinka").style.border = "2px solid green"; }

else {

document.getElementById("lozinka").style.border = "2px solid red"; }

}

</script>

# 5. PHP kod

**U okviru ove sekcije navesti PHP kod koji je student isključivo sam napisao. Prored uzmedju redova ne sme biti veći od 6pt.**

### 5.1 GalleryController.php

<?php

namespace App\Http\Controllers\Admin;

use App\Models\Gallery; use App\Models\Slika; use

Illuminate\Database\QueryException; use Illuminate\Http\Request; use

App\Http\Controllers\Controller; use

Symfony\Component\HttpFoundation\File\Exception\FileException;

class GalleryController extends BackendController

{ public function \_\_construct()

{

parent::construct('admin.pages.gallery',"Gallery management", "Manage your webiste's

gallery pictures", "gallery.create", "gallery.index");

$this->model = new Gallery();

}

public function index()

{

$this->data['galleries'] = $this->model->all(); return view($this->getView(), $this->data);

}

public function create()

{

$this->data['form'] = 'insert'; return view('admin.pages.gallery', $this->data);

}

public function store(Request $request)

{

$rules = [

'title' => 'required|alpha\_num|min:3|max:100|unique:galerija',

'description' => 'required',

'picture' => 'required|image|mimes:jpeg,png,jpg|max:2000'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

try

{

$file = $request->file('picture');

$directory = public\_path("images/");

$fileName = time() . "\_" . $file->getClientOriginalName(); $file->move($directory, $fileName);

$slika = new slika();

$slika->putanja = "images/" . $fileName;

$slika->alt = "tutorijal gallery";

$this->model->picture\_id = $slika->save2();

$this->model->title = $request->get('title');

$this->model->description = $request->get('description');

$this->model->save();

return redirect(route('gallery.index'))->with("success", "Slika je uspesno dodata u

galeriju!");

} catch (FileException $e) {

\Log::error("An error occured while uploading gallery picture " . $e->getMessage());

} catch (QueryException $e) {

\Log::error("An error occured while inserting gallery picture into database " . $e>getMessage());

}

return redirect()->back()->with("error", "An error occured, please try again later");

}

public function destroy($id)

{ try {

$pictureId = $this->model->find($id)->slika\_id;

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

try {

$pictureModel = new Slika();

$picture = $pictureModel->find($pictureId); unlink(public\_path($picture>putanja));

$pictureModel->delete($pictureId);

} catch(\Exception $e) {

\Log::error("Greska pri brisanju slike galerije " . $e->getMessage());

}

return redirect()->back()->with("success", "Picture successfully deleted!");

} catch (QueryException $e) { return redirect()->back()->with("error", "An error occured, please try again later");

}

}

public function show($id)

{

$this->data['gallery'] = $this->model->find($id);

$this->data['form'] = 'edit';

return view($this->getView(), $this->data);

}

public function update(Request $request, $id)

{

$rules = [

'title' => 'required|alpha\_num|min:3|max:100',

'description' => 'required',

'picture' => 'nullable|image|mimes:jpeg,png,jpg|max:2000'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

$oldPictureId = null; try

{

if ($request->hasFile("picture")) {

$oldPictureId = $this->model->find($id)->slika\_id;

$file = $request->file('picture');

$directory = public\_path("images/");

$fileName = time() . "\_" . $file->getClientOriginalName();

$file->move($directory, $fileName);

$pictureModel = new Slika();

$pictureModel->putanja = "images/" . $fileName;

$pictureModel->alt = "blog gallery";

$this->model->picture\_id = $pictureModel->save();

}

$this->model->title = $request->get('title');

$this->model->description = $request->get('description');

$this->model->update($id); try {

if($oldPictureId) {

$pictureModel = new

Slika();

$picture = $pictureModel->find($oldPictureId); unlink(public\_path($picture->putanja));

$pictureModel->delete($oldPictureId);

}

} catch (\Exception $e) {

\Log::error("Greska pri brisanju slike navigacije: " . $e->getMessage()); }

return redirect()->back()->with("success", "Picture successfully updated!");

} catch (FileException $e) {

\Log::error("An error occured while uploading gallery picture " . $e->getMessage());

} catch (QueryException $e) {

\Log::error("An error occured while inserting gallery picture into database " . $e>getMessage());

}

return redirect()->back()->with("error", "An error occured, please try again later");

} }

### 5.2 Navigation Controller.php

<?php

namespace App\Http\Controllers\Admin;

use App\Models\Meni; use

Illuminate\Database\QueryException; use

Illuminate\Http\Request; use

App\Http\Controllers\Controller;

class NavigationController extends BackendController

{ public function

\_\_construct()

{

parent::construct('admin.pages.navigation',"Navigation links management", "Manage

your webiste's main navigation", "navigation.create", "navigation.index"); $this>model = new Meni();

}

public function index()

{

$this->data['navigations'] = $this->model->getAll(); return view($this->getView(), $this->data);

}

public function create()

{

$this->data['form'] = 'insert'; return view($this->getView(), $this->data);

}

public function store(Request $request)

{

$rules = [

'route' => 'required',

'name' => 'required'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

try {

$this->model->route = $request->get('route');

$this->model->name = $request->get("name");

$this->model->save(); return redirect(route('navigation.index'))->with("success", "Navigation link

successfully added!");

} catch(QueryException $e) {

\Log::error("Greska pri unosu navigacionog linka " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

public function destroy($id)

{ try { $this->model->delete($id); return redirect(route('navigation.index'))-

>with("success", "Navigation link successfully deleted!");

} catch (QueryException $e) {

\Log::error("Greska pri brisanju navigacionog linka: " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

public function show($id)

{

$this->data['form'] = 'edit';

$this->data['navigation'] = $this->model->find($id); return view($this->getView(), $this->data);

}

public function update(Request $request, $id)

{

$rules = [

'route' => 'required',

'name' => 'required'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate(); try {

$this->model->route = $request->get('route');

$this->model->name = $request->get("name");

$this->model->update($id); return redirect(route('navigation.index'))->with("success", "Navigation link

successfully updated!");

} catch(QueryException $e) {

\Log::error("Greska pri unosu izmeni linka " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later"); }

}

}

### 5.3 PostController.php

<?php

namespace App\Http\Controllers\Admin;

use App\Models\Meni; use

Illuminate\Database\QueryException; use

Illuminate\Http\Request; use

App\Http\Controllers\Controller;

class NavigationController extends BackendController

{ public function \_\_construct()

{

parent::construct('admin.pages.navigation',"Navigation links management", "Manage

your webiste's main navigation", "navigation.create", "navigation.index"); $this>model = new Meni();

}

public function index()

{

$this->data['navigations'] = $this->model->getAll(); return view($this->getView(), $this->data);

}

public function create()

{

$this->data['form'] = 'insert'; return view($this->getView(), $this->data);

}

public function store(Request $request)

{

$rules = [

'route' => 'required',

'name' => 'required'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

try {

$this->model->route = $request->get('route');

$this->model->name = $request->get("name");

$this->model->save(); return redirect(route('navigation.index'))-

>with("success", "Navigation link successfully added!");

} catch(QueryException $e) {

\Log::error("Greska pri unosu navigacionog linka " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

public function destroy($id)

{ try {

$this->model->delete($id); return redirect(route('navigation.index'))->with("success", "Navigation link

successfully deleted!");

} catch (QueryException $e) {

\Log::error("Greska pri brisanju navigacionog linka: " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

public function show($id)

{

$this->data['form'] = 'edit';

$this->data['navigation'] = $this->model->find($id); return view($this->getView(), $this->data);

}

public function update(Request $request, $id)

{

$rules = [

'route' => 'required',

'name' => 'required'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

try {

$this->model->route = $request->get('route');

$this->model->name = $request->get("name");

$this->model->update($id); return redirect(route('navigation.index'))->with("success", "Navigation link

successfully updated!");

} catch(QueryException $e) {

\Log::error("Greska pri unosu izmeni linka " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

}

### 5.4 CommentsController.php

**<?php  
  
namespace** App\Http\Controllers;  
  
**use** App\Models\CommentsModel;  
**use** Illuminate\Database\QueryException;  
**use** Illuminate\Http\Request;  
  
**class** CommentsController **extends** Controller  
{  
 **public function** postComment(Request $request, $postId)  
 {  
 **if** ($request->get("content")) {  
 $commentsModel = **new** CommentsModel();  
 $commentsModel->content = $request->get("content");  
 $commentsModel->korisnik\_id=session()->get('user')[0]->id;  
 $commentsModel->post\_id = $postId;  
  
 **try** {  
 $commentsModel->save();  
 \Log::*info*('Korisnik je dodao nov komentar: '.$request->session()->get('user')[0]->korisnicko\_ime ." ".date("Y-m-d H:i:s"));  
 **return** redirect()->back()->with('success', "Komentar uspesno dodat.");  
 } **catch** (QueryException $e) {  
 \Log::*error*("Greska pri dodavanju komentara " . $e->getMessage());  
 **return** redirect()->back()->with('warning', "Doslo je do greske na serveru.");  
 }  
  
 }  
 **return** redirect()->back()->with('warning', "Poruka ne sme biti prazna.");  
 }  
 **public function** showEditForm(Request $request, $commentId){  
  
 $promenljiva=[];  
 $commentModel=**new** CommentsModel();  
 $promenljiva['comments']=$commentModel->getSingleComment($commentId);  
 $request->session()->put('commentId',$commentId);  
 **return** redirect()->back()->with('singlecomment',$promenljiva['comments']->tekst);  
 }  
 **public function** update(Request $request, $commentId){  
 $commentsModel= **new** CommentsModel();  
 $commentsModel->content=$request->get('content');  
 $commentsModel->update($commentId);  
 **return** redirect()->back()->with('success','Komentar editovan!');  
 }  
 **public function** delete($commentId)  
 {  
 $commentModel=**new** CommentsModel();  
 $commentModel->delete($commentId);  
 **return** redirect()->back()->with('success','Komentar obrisan!');  
  
 }  
  
}

### 5.5 RoleController.php

<?php

namespace App\Http\Controllers\Admin;

use App\Models\Uloga; use

Illuminate\Database\QueryException; use

Illuminate\Http\Request; use

App\Http\Controllers\Controller;

class RoleController extends BackendController

{ public function

\_\_construct()

{

parent::construct("admin.pages.roles", "Roles management", "Manage your user's roles", "roles.create", "roles.index");

$this->roleModel = new Uloga();

}

public function index()

{

$this->data['roles'] = $this->roleModel->selectAll(); return view($this->getView(), $this->data);

}

public function create()

{

$this->data['form'] = 'insert'; return view($this->getView(), $this->data);

}

public function store(Request $request)

{

$request->validate([

'name' => 'required|alpha|min:2|max:20'

], $request->all());

$this->roleModel->name = $request->get('name'); try {

$this->roleModel->insert(); return redirect(route('roles.index'))->with("success", "Uloga je uspesno dodata!");

} catch(QueryException $e) {

\Log::error("Greška pri dodavanju uloge: " . $e->getMessage()); return redirect()->back()->with("error", "An error has occurred! Please check the log file.");

}

}

public function destroy($id)

{ try {

$this->roleModel->delete($id); return redirect(route('roles.index'))-

>with("success", "Ruta uspesno obrisana!");

} catch (QueryException $e) {

\Log::error("Greska pri brisanju rute " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

}

public function show($id)

{

$this->data['form'] = 'edit';

$this->data['role'] = $this->roleModel->selectOne($id); return view($this->getView(), $this->data);

}

public function update(Request $request, $id)

{

$rules = [

'name' => 'required'

];

$validator = \Validator::make($request->all(), $rules);

$validator->validate();

try {

$this->roleModel->name = $request->get("name");

$this->roleModel->update($id); return redirect(route('roles.index'))-

>with("success", "Navigation link successfully updated!");

} catch(QueryException $e) {

\Log::error("Greska pri unosu izmeni linka " . $e->getMessage()); return redirect()->back()->with("error", "An error occurred, please try again later");

}

} }

### 5.6 UsersController.php

<?php

namespace App\Http\Controllers\Admin;

use App\Models\Korisnik; use

App\Models\Uloga; use

Illuminate\Database\QueryException; use

Illuminate\Http\Request; use

App\Http\Controllers\Controller;

class UsersController extends BackendController

{

public function \_\_construct()

{

parent::construct('admin.pages.users',"Users management", "Manage your webiste's users", "users.create", "users.index");

$this->model = new Korisnik();

}

public function index()

{

$this->data['korisnici'] = $this->model->getAll(); return view($this->getView(), $this->data);

}

public function create()

{

$this->data['form'] = 'insert';

$uloga = new Uloga();

$this->data['roles'] = $uloga->selectAll(); return view($this->getView(), $this->data);

}

public function store(Request $request)

{

//Definisanje pravila za validaciju

$rules = [

'korisnickoIme' => 'required|alpha|min:4|max:20',

'email' => 'required|email|unique:korisnik',

'lozinka' => [

'required',

'min:6',

'regex:/^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)[a-zA-Z\d]{6,}$/'

],

'role\_id' => 'required'

];

//Kastomizacija poruka

$messages = [

"password.regex" => 'Password must contain one uppercase letter and one number.', 'role\_id.required' => 'User role must be selected.'

];

//Primena validacije, ukoliko je ima grešaka vrši se redirekcija requesta- na prethodnu stranu sa sve greškama

$validator = \Validator::make($request->all(), $rules, $messages); $validator>validate();

//Ukoliko je validacija uspesna, korisnik se upisuje u bazu

$korisnik = new Korisnik();

$korisnik->korisnicko\_ime = $request->get('korisnickoIme');

$korisnik->email = $request->get("email");

$korisnik->lozinka = $request->get("lozinka"); $korisnik->uloga\_id = $request->get("role\_id"); try {

$korisnik->save(); return redirect(route("users.index"))->with("success", "Korisnik uspesno dodat!!");

} catch(\Exception $e) {

\Log::error($e->getMessage()); return redirect()->back()->with("error", "Doslo je do greske, pokusajte kasnije!");

}

}

public function show($id)

{

$this->data['korisnik'] = $this->model->selectOne($id);

$this->data['form'] = 'edit';

$uloga = new Uloga();

$this->data['roles'] = $uloga->selectAll(); return view($this->getView(), $this->data);

}

public function update(Request $request, $id)

{

//Definisanje pravila za validaciju

$rules = [

'korisnickoIme' => 'required|alpha|min:4|max:20',

'email' => 'required|email|unique:korisnik',

'lozinka' => [

'required',

'min:6',

'regex:/^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)[a-zA-Z\d]{6,}$/'

],

'role\_id' => 'required'

];

//Kastomizacija poruka

$messages = [

"password.regex" => 'Password must contain one uppercase letter and one number.', 'role\_id.required' => 'User role must be selected.'

];

//Primena validacije, ukoliko je ima grešaka vrši se redirekcija requesta- na prethodnu stranu sa sve greškama

$validator = \Validator::make($request->all(), $rules, $messages); $validator>validate();

//Ukoliko je validacija uspesna, korisnik se upisuje u bazu

$korisnik = new Korisnik();

$korisnik->korisnicko\_ime = $request->get('korisnickoIme');

$korisnik->email = $request->get("email");

$korisnik->lozinka = $request->get("lozinka"); $korisnik->uloga\_id = $request->get("role\_id"); try {

$korisnik->update($id);

return redirect(route("users.index"))->with("success", "Korisnik uspesno izmenjen!"); } catch(\Exception $e) {

\Log::error($e->getMessage()); return redirect()->back()->with("error", "Doslo je do greske, pokusajte kasnije!");

}

}

public function destroy($id)

{ try {

$this->model->delete($id); return redirect(route("users.index"))-

>with("success", "Korisnik uspesno obrisan");

} catch (\Exception $e) { \Log::error($e->getMessage()); return redirect(route("users.index"))->with("error", "Doslo je do greske, pokusajte kasnije!");

}

}

}

### 5.7 FrontendController.php

**<?php**/\*  
 \* To change this license header, choose License Headers in Project Properties.  
 \* To change this template file, choose Tools | Templates  
 \* and open the template in the editor.  
 \*/  
  
**namespace** App\Http\Controllers;  
**use** App\Models\Meni;  
**use** App\Models\Post;  
**use** App\Models\Slika;  
**use** App\Models\Gallery;  
**use** App\Models\CommentsModel;  
**use** Illuminate\Http\Response;  
**use** Illuminate\Http\Request;  
**use** Illuminate\Support\Facades\File;  
*/\*\*  
 \* Description of FrontendController  
 \*  
 \** ***@author*** *korisnik  
 \*/***class** FrontendController **extends** Controller {  
 //put your code here  
 **private** $data = [];  
  
 **public function** \_\_construct(){  
 $meni = **new** Meni();  
 $this->data['menus'] = $meni->getAll();  
 }  
 **public function** index(){  
 $post = **new** Post();  
 $this->data['posts'] = $post->getAll();  
 **return** view('pages.home', $this->data);  
 }  
 **public function** showRegistrationForm()  
 {  
 **return** view('pages.register', $this->data);  
 }  
 **public function** single($id)  
 {  
 $post = **new** Post();  
 $post->id = $id;  
 $this->data['post']=$post->get();  
  
 $commentsModel = **new** CommentsModel();  
 $commentsModel->post\_id=$id;  
 $this->data['comments'] = $commentsModel->getPostComments($id);  
// $this->data['post'] = $post;  
 **return** view('pages.single',$this->data);  
 }  
  
 **public function** gallery()  
 {  
 $gallery = **new** Gallery();  
 $this->data['galleries'] = $gallery->all();  
 **return** view("pages.gallery", $this->data);  
 }  
 **public function** contact(){  
 **return** view('pages.contact',$this->data);  
 }  
 **public function** kontaktPosalji(Request $request){  
 $request->validate(  
 [  
 'name' => 'required|min:3',  
 'email' => 'required|email',  
 'headline' => 'required|min:3',  
 'message' => 'required|min:3'  
 ]  
 );  
// $moj\_email;  
//// $moje  
//// $data = array('name'=>$request->input('name'),  
//// "email" => $request->input('email'),  
//// "message" =>$request->input('message')  
//// );  
//// $email=$request->input('email');  
////  
//// mail::send($email,$data,function ($message) use )  
///  
 }  
 **public function** autor()  
 {  
 **return** view("pages.autor",$this->data);  
 }   
  
 **public function** createPost()  
 {  
 **return** view('pages.createPost',$this->data);   
 }  
 **public function** storePost(Request $request)  
 {  
 $rules = [  
 'title' => 'required',  
 'body' => 'required',  
 'embedLink' => 'required',  
 'photo' => 'required|mimes:jpg,jpeg,png,gif|max:3000',  
 'alt' => 'required'  
 ];  
 $custom\_messages = [  
 'required' => 'Polje :attribute je obavezno!',  
 'title.regex' => 'Polje title nije u ispravnom formatu!',  
 'max' => 'Fajl ne sme biti veci od :max KB.',  
 'mimes' => 'Dozvoljeni formati su: :values.'  
 ];  
 $request->validate($rules, $custom\_messages);   
  
 $photo = $request->file('photo');  
 $extension = $photo->getClientOriginalExtension();  
 $tmp\_path = $photo->getPathName();  
   
 $folder = 'images/';  
 $file\_name = time().".".$extension;  
 $new\_path = public\_path($folder).$file\_name;  
  
 **try** {  
 // 4 - Upload slike na server  
  
 File::*move*($tmp\_path, $new\_path);  
  
 // 5 - Unos slike u bazu  
  
 $slika = **new** Slika();  
 $slika->alt = trim($request->get('alt'));  
 $slika->putanja = 'images/'.$file\_name;  
 $slika\_id = $slika->save();  
  
 // 6 - Unos posta u bazu  
 $post = **new** Post();  
 $post->naslov = $request->get('title');  
 $post->sadrzaj = $request->get('body');  
 $post->video = $request->get('embedLink');  
 $post->korisnik\_id = $request->session()->get('user')[0]->id;  
 $post->slika\_id = $slika\_id;  
 $post->save();  
  
 // Ako se ne desi nijedan Exception - otici ce na pocetnu stranu!  
 \Log::*info*('Korisnik je dodao nov tutorijal: '.$request->session()->get('user')[0]->korisnicko\_ime ." ".date("Y-m-d H:i:s"));  
 **return** redirect('/')->with('success','Uspesno ste dodali post i sliku!');  
 }  
 **catch**(\Illuminate\Database\QueryException $ex){ // greske u upitu  
 \Log::*error*($ex->getMessage());  
 **return** redirect()->back()->with('error','Greska pri dodavanju posta u bazu!');  
 }  
 **catch**(\Symfony\Component\HttpFoundation\File\Exception\FileException $ex) { // greske sa fajlom  
 \Log::*error*('Problem sa fajlom!! '.$ex->getMessage());  
 **return** redirect()->back()->with('error','Greska pri dodavanju slike!');  
 }  
 **catch**(\ErrorException $ex) {   
 \Log::*error*('Problem sa fajlom!! '.$ex->getMessage());  
 **return** redirect()->back()->with('error','Desila se greska..');  
 }   
 }  
}

### 5.8 LoginController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request; use App\Models\Korisnik; use

Illuminate\Support\Facades\File;

class LoginController extends Controller {

public function login(Request $request){

$request->validate([

'tbKorisnickoIme' => 'required|alpha',

'tbLozinka' => 'required'

]);

$korisnickoIme = $request->get('tbKorisnickoIme');

$lozinka = $request->get('tbLozinka');

$korisnik = new Korisnik();

$korisnik->korisnicko\_ime = $korisnickoIme;

$korisnik->lozinka = $lozinka;

$loginKorisnik = $korisnik->getByUsernameAndPassword(); if(!empty($loginKorisnik)){

$request->session()->push('user', $loginKorisnik); return redirect()->back()->with('success','Uspesno ste se ulogovali!');

}

return redirect()->back()->with('error','Niste registrovani!');

}

public function logout(Request $request){

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

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

return redirect('/');

}

public function registerStore(Request $request)

{

$request->validate([

'korisnickoIme' => 'required|unique:korisnik,korisnicko\_ime',

'email' => 'required|email',

'lozinka' => 'required'

]);

$korisnicko\_ime=$request->get('korisnickoIme');

$lozinka = $request->get('lozinka');

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

/\*

$slika = $request->file('slika');

$tmp\_putanja = $slika->getPathName();

$ekstenzija = $slika->getClientOriginalExtension();

$ime\_fajla = time().'.'.$ekstenzija;

$putanja = 'images/'.$ime\_fajla;

$putanja\_server = public\_path($putanja);\*/

try {

/\*File::move($tmp\_putanja, $putanja\_server);\*/

// unos u bazu

$korisnik = new Korisnik();

$korisnik->korisnicko\_ime = $korisnicko\_ime;

$korisnik->email = $email;

$korisnik->lozinka = $lozinka;

$rez = $korisnik->save();

if($rez == 1)

{

return redirect('/register')->with('message','Uspesna registracija!');

} else

{

return redirect('/register')->with('message','Greska pri registraciji!');

}

}

catch (\Exception $ex){

\Log::error('MOJA GRESKA: '.$ex->getMessage());

}

}

}

### 5.9 AdminMiddleware.php

<?php

namespace App\Http\Middleware;

use Closure;

class AdminMiddleware

{ public function handle($request, Closure

$next)

{

if($request->session()->has('user')){

$korisnik = $request->session()->get('user')[0]; if($korisnik->naziv == 'admin'){ return

$next($request);

} else { return redirect()->back()->with('error','Nemate pravo pristupa ovoj stranici!');

}

}

return redirect()->back()->with('error','Nemate pravo pristupa ovoj stranici!'); } }

### 5.10 UserMiddleware.php

<?php

namespace App\Http\Middleware;

use Closure;

class UserMiddleware

{ public function handle($request, Closure

$next)

{

if($request->session()->has('user')){

$korisnik = $request->session()->get('user')[0];

if($korisnik->naziv == 'korisnik'){

return $next($request);

} else { return redirect()->back()->with('error','Nemate pravo pristupa ovoj stranici!');

}

}

return redirect()->back()->with('error','Nemate pravo pristupa ovoj stranici!'); } }

### 5.11 Gallery.php

<?php

namespace App\Models; use

Illuminate\Support\Facades\DB;

class Gallery

{ public $title; public

$description; public

$picture\_id; private

$table = 'galerija';

public function all()

{ return

DB::table('galerija')

->join('slika', 'galerija.slika\_id', '=', 'slika.id')

->select("galerija.\*", "slika.putanja AS putanja", "slika.alt AS alt")

->get();

}

public function save()

{ return \DB::table($this-

>table) ->insertGetId([

'title' => $this->title,

'description' => $this->description,

'slika\_id' => $this->picture\_id

]);

}

public function find($id) { return \DB::table($this-

>table)

->join('slika', 'galerija.slika\_id', '=', 'slika.id')

->where('galerija.id', $id)

->select("galerija.\*", "slika.putanja", "slika.alt")

->get()->first();

}

public function delete($id)

{

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

->delete($id);

}

public function update($id)

{ if($this-

>picture\_id) {

$updateData = [

'title' => $this->title,

'description' => $this->description,

'slika\_id' => $this->picture\_id

];

} else {

$updateData = [

'title' => $this->title,

'description' => $this->description

];

} return \DB::table($this->table) >where('id', $id)

->update($updateData);

} }

### 5.12 Korisnik.php

<?php

namespace App\Models; use

Illuminate\Support\Facades\DB;

class Korisnik

{

public $id; public $korisnicko\_ime; public $lozinka; public $email; public $uloga\_id=2;

private $table = 'korisnik';

|  |  |
| --- | --- |
| public function getAll(){  $rezultat = DB::table('korisnik') |  |
|  | ->select('\*', 'korisnik.id AS korisnikId') |
|  | ->join('uloga','uloga.id','=','korisnik.uloga\_id') |
|  | ->get(); |

return $rezultat;

}

public function get() {

$rezultat = DB::table('korisnik')

->select('\*')

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

->first();

return $rezultat;

}

public function selectOne($id)

{

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

->where("id", $id)->first();

}

public function getByUsernameAndPassword(){

$rezultat = DB::table('korisnik')

->select('korisnik.\*','uloga.naziv')

->join('uloga','korisnik.uloga\_id', '=', 'uloga.id')

->where([

'korisnicko\_ime' => $this->korisnicko\_ime,

'lozinka' => md5($this->lozinka)

])

->first(); return $rezultat;

}

public function save() {

$rez = DB::table('korisnik')->insert([

'korisnicko\_ime' => $this->korisnicko\_ime,

'lozinka' => md5($this->lozinka),

'email' => $this->email,

'uloga\_id' => $this->uloga\_id

]);

return $rez;

}

public function update($id) { return \DB::table($this-

>table)

->where("id", $id)

->update([

'korisnicko\_ime' => $this->korisnicko\_ime,

'lozinka' => md5($this->lozinka),

'email' => $this->email,

'uloga\_id' => $this->uloga\_id

]);

}

public function delete($id)

{

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

->delete($id);

} }

### 5.13 Meni.php

<?php

namespace App\Models;

use Illuminate\Support\Facades\DB; class Meni {

public $route; public $name; private

$table = 'meni';

public function getAll(){

$rezultat = DB::table('meni')->get(); return $rezultat;

}

public function save()

{

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

>insertGetId([

'link' => $this->route,

'naziv' => $this->name

]);

}

public function delete($id)

{

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

->delete($id);

}

public function find($id)

{

return \DB::table($this->table)->where('id', $id)->get()->first();

}

public function update($id)

{

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

->where('id', $id)

->update([

'link' => $this->route,

'naziv' => $this->name

]);

}

}

### 5.14 Post.php

<?php

namespace App\Models; use

Illuminate\Support\Facades\DB;

class Post {

public $id; public $naslov; public $video; public $sadrzaj; public $korisnik\_id; public $slika\_id;

|  |  |
| --- | --- |
| public function getAll(){  $rezultat = DB::table('post') |  |
|  | ->select('\*','post.id as postId') |
|  | ->join('slika as s','post.slika\_id','=','s.id') |
|  | ->join('korisnik as k','post.korisnik\_id','=','k.id') |
|  | ->orderBy('post.created\_at','desc') |
| return $rezultat;    } | ->paginate(2); |
| public function getAll2(){  $rezultat = DB::table('post') |  |
|  | ->select('\*','post.id as postId') |
|  | ->join('slika as s','post.slika\_id','=','s.id') |

->join('korisnik as k','post.korisnik\_id','=','k.id')

->orderBy('post.created\_at','desc')

->get();

return $rezultat;

}

public function get(){ return DB::table('post')

->select('\*','post.id as postId')

->join('slika as s','post.slika\_id','=','s.id')

->join('korisnik as k','post.korisnik\_id','=','k.id')

->where('post.id', $this->id)

->orderBy('post.created\_at','desc')

->first();

}

public function get2($id){ return DB::table('post')

->select('\*','post.id as postId')

->join('slika as s','post.slika\_id','=','s.id')

->join('korisnik as k','post.korisnik\_id','=','k.id')

->where('post.id', $id)

->first();

}

public function save() {

$rezultat = DB::table('post')->insert([

'naslov' => $this->naslov,

'sadrzaj' => $this->sadrzaj,

'video' => $this->video,

'korisnik\_id' => $this->korisnik\_id,

'created\_at' => time(),

'slika\_id' => $this->slika\_id

]);

return $rezultat;

}

public function delete($id)

{ return

\DB::table('post')

->delete($id); } public function update($id)

{

$updateData = [

'naslov' => $this->naslov,

'sadrzaj' => $this->sadrzaj,

'video' => $this->video

];

if ($this->slika\_id != null) { $updateData['slika\_id'] = $this->slika\_id;

} return \DB::table('post') >where('id', $id)

->update($updateData);

} }

### 5.15 CommentsModel.php

**<?php  
  
  
namespace** App\Models;  
  
  
**class** CommentsModel  
{  
 **public** $content;  
 **public** $post\_id;  
 **public** $korisnik\_id;  
 **private** $table = 'komentari';  
  
 **public function** save()  
 {  
 **return** \DB::*table*($this->table)  
 ->insert([  
 'korisnik\_id' => $this->korisnik\_id,  
 'post\_id' => $this->post\_id,  
 'tekst' => $this->content,  
 'created\_at' => date("Y-m-d H:i:s"),  
 'updated\_at' => date("Y-m-d H:i:s")  
 ]);  
 }  
 **public function** getSingleComment($commentId){  
 **return** \DB::*table*($this->table)  
 ->where('id',$commentId)  
 ->first();  
 }  
  
 **public function** update($id)  
 {  
 **return** \DB::*table*($this->table)  
 ->where('id', $id)  
 ->update([  
 'tekst' => $this->content,  
 'updated\_at' => date("Y-m-d H:i:s")  
 ]);  
 }  
  
 **public function** getPostComments($postId)  
 {  
 **return** \DB::*table*($this->table)  
 ->join("korisnik", "komentari.korisnik\_id", "=", "korisnik.id")  
 ->where('post\_id', $postId)  
 ->select('komentari.\*', 'korisnik.korisnicko\_ime')  
 ->get();  
 }  
  
  
 **public function** delete($id)  
 {  
 **return** \DB::*table*($this->table)  
 ->delete($id);  
 }  
  
  
  
 **public function** find($id)  
 {  
 **return** \DB::*table*($this->table)  
 ->where('id', $id)->get()->first();  
 }  
  
}

### 5.16 Slika.php

<?php

namespace App\Models; use Illuminate\Support\Facades\DB;

class Slika {

public $id;

public $alt; public

$putanja;

public function save(){ $id = DB::table('slika')

->insertGetId([

'alt' => $this->alt,

'putanja' => $this->putanja

]);

return $id;

}

public function save2()

{ return

\DB::table('slika')

->insertGetId([

'putanja' => $this->putanja,

'alt' => $this->alt

]);

}

public function find($id)

{

return \DB::table('slika')

->where('id', $id)

->get()

->first();

}

public function delete($id)

{ return

\DB::table('slika')

->delete($id);

}

}

### 5.17 Uloga.php

<?php

namespace App\Models;

class Uloga

{ public

$name;

public function selectAll()

{

return \DB::table("uloga")->get();

}

public function insert()

{ return \DB::table('uloga') >insertGetId([

'naziv' => $this->name

]);

}

public function delete($id)

{

return \DB::table("uloga")

->delete($id);

}

public function update($id)

{ return

\DB::table('uloga')

->where('id', $id)

->update([

'naziv' => $this->name

]);

}

public function selectOne($id)

{ return

\DB::table('uloga')

->where('id', $id)

->get()

->first();

}

}

### 5.18 admin.components.common.footer.blade.php

<footer class="sticky-footer">

<div class="container">

<div class="text-center">

<small>Copyright © Your Website 2018</small>

</div>

</div>

</footer>

### 5.19 admin.components.common.head.blade.php

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<meta name="description" content="">

<meta name="author" content="">

<title>SB Admin - Start Bootstrap Template</title>

<!-- Bootstrap core CSS-->

<link href="{{asset('admin/vendor/bootstrap/css/bootstrap.min.css')}}" rel="stylesheet">

<!-- Custom fonts for this template-->

<link href="{{asset('admin/vendor/font-awesome/css/font-awesome.min.css')}}" rel="stylesheet" type="text/css">

<!-- Page level plugin CSS-->

<link href="{{asset('admin/vendor/datatables/dataTables.bootstrap4.css')}}" rel="stylesheet">

<!-- Custom styles for this template-->

<link href="{{asset('admin/css/sb-admin.css')}}" rel="stylesheet"> </head>

### 5.20 admin.components.common.scripts.blade.php

<script src="{{asset('admin/vendor/jquery/jquery.min.js')}}"></script>

<script src="{{asset('admin/vendor/bootstrap/js/bootstrap.bundle.min.js')}}"></script>

<!-- Core plugin JavaScript-->

<script src="{{asset('admin/vendor/jquery-easing/jquery.easing.min.js')}}"></script>

<!-- Page level plugin JavaScript-->

<script src="{{asset('admin/vendor/chart.js/Chart.min.js')}}"></script>

<script src="{{asset('admin/vendor/datatables/jquery.dataTables.js') }}"></script>

<script src="{{asset('admin/vendor/datatables/dataTables.bootstrap4.js')}}"></script>

<!-- Custom scripts for all pages-->

<script src="{{asset('admin/js/sb-admin.min.js')}}"></script>

<!-- Custom scripts for this page-->

<script src="{{asset('admin/js/sb-admin-datatables.min.js')}}"></script>

<script src="{{asset('admin/js/sb-admin-charts.min.js')}}"></script>

### 5.21 admin.components.common.sidenav.blade.php

<ul class="navbar-nav navbar-sidenav" id="exampleAccordion">

<li class="nav-item" data-toggle="tooltip" data-placement="right" title="" dataoriginaltitle="Menu Levels">

<a class="nav-link nav-link-collapse collapsed" data-toggle="collapse" href="#collapseMulti" data-parent="#exampleAccordion" aria-expanded="false">

<i class="fa fa-fw fa-sitemap"></i>

<span class="nav-link-text">Uredi</span>

</a>

<ul class="sidenav-second-level collapse" id="collapseMulti" style="">

<li>

<a href="{{route('users.create')}}">Dodaj korisnika</a>

</li>

<li>

<a href="{{route('posts.create')}}">Dodaj tutorijal</a>

</li>

<li>

<a href="{{route('gallery.create')}}">Dodaj sliku u galeriju</a>

</li>

<li>

<a href="{{route('roles.create')}}">Dodaj novu ulogu</a>

</li>

<li>

<a href="{{route('navigation.create')}}">Dodaj stavku u meni</a>

</li>

<li>

<a class="nav-link-collapse" data-toggle="collapse" href="#collapseMulti2" ariaexpanded="true">Prikazi</a>

<ul class="sidenav-third-level collapse show" id="collapseMulti2" style="">

<li>

<a href="{{route('users.index')}}">Prikazi korisnike</a>

</li>

<li>

<a href="{{route('posts.index')}}">Prikazi tutorijale</a>

</li>

<li>

<a href="{{route('gallery.index')}}">Prikazi galeriju</a>

</li>

<li>

<a href="{{route('roles.index')}}">Prikazi uloge</a>

</li>

<li>

<a href="{{route('navigation.index')}}">Prikazi stavke iz menija</a>

</li>

</ul>

</li>

</ul>

</li>

</ul>

<ul class="navbar-nav sidenav-toggler">

<li class="nav-item">

<a class="nav-link text-center" id="sidenavToggler">

<i class="fa fa-fw fa-angle-left"></i>

</a>

</li>

</ul>

## 5.22 admin.components.common.topnav.blade.php

<ul class="navbar-nav ml-auto">

<li class="nav-item">

<a href="{{route('logout')}}">

<i class="fa fa-fw fa-sign-out"></i>Logout</a>

</li>

</ul>

## 5.23 admin.components.gallery.table.blade.php

<table id="mainTable" class="table table-striped">

<thead>

<tr>

<th>Naslov</th>

<th>Opis</th>

<th>Slika</th>

<th>Izmeni</th>

<th>Obrisi</th>

</tr>

</thead>

<tbody>

@foreach($galleries as $g)

<tr>

<td>{{ $g->title }}</td>

<td>{{ $g->description }}</td>

<td><img height="100" src="{{ asset($g->putanja) }}" alt="{{ asset($g->alt) }}"></td>

<td><a href="{{ route("gallery.show", ['id' => $g->id]) }}" >Izmeni</a></td>

<td><a href="{{ route("gallery.delete", ['id' => $g->id]) }}" >Obrisi</a></td>

</tr>

@endforeach

</tbody>

</table>

## 5.24 admin.components.gallery.insert\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Add new gallery picture</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route("gallery.store") }}" method="post" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="title" type="text" class="form-control" placeholder="Title">

</div>

</div>

<div class="form-group">

<div class="form-line">

<textarea name="description" type="text" class="form-control" placeholder="Description"></textarea>

</div>

</div>

<div class="form-group">

<label for="picture">Slika:</label>

<input name="picture" type="file" id="picture">

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Add">

<a href="{{ route("gallery.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.25 admin.components.gallery.edit\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Update existing gallery post</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route("gallery.update", ['id' => $gallery->id]) }}" method="post" enctype="multipart/form-data">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="title" type="text" class="form-control" value="{{ $gallery->title }}" placeholder="Title">

</div>

</div>

<div class="form-group">

<div class="form-line">

<textarea name="description" type="number" class="form-control" placeholder="Description">{{ $gallery->description }}

</textarea>

</div>

</div>

<div class="form-group">

<label for="picture">Picture:</label>

<input name="picture" type="file" id="picture">

<br>

<img class="img thumbnail" height="200px" src="{{ asset($gallery->putanja) }}" alt="{{ $gallery->alt }}">

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Edit">

<a href="{{ route("gallery.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div> </div>

## 5.26 admin.components.navigation.table.blade.php

<table class="table">

<tr>

<td>ID</td>

<td>Naziv</td>

<td>Ruta</td>

<td>Izmeni</td>

<td>Obrisi</td>

</tr>

<!-- Prikaz korisnika-->

@isset($navigations)

@foreach($navigations as $nav)

<tr>

<td> {{ $nav->id }} </td>

<td> {{ $nav->naziv }} </td>

<td> {{ $nav->link }} </td>

<td> <a href="{{ route('navigation.show', ['id' => $nav->id]) }}">Izmeni</a> </td>

<td> <a href="{{ route('navigation.delete', ['id' => $nav->id]) }}">Obrisi</a> </td>

</tr>

@endforeach

@endisset

</table>

## 5.27 admin.components.navigation.insert\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Add new menu item</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('navigation.store') }}" method="POST" enctype="multipart/form-data">

{{ csrf\_field() }}

<div class="form-group">

<input name="name" type="text" class="form-control" placeholder="Naziv linka">

</div>

<div class="form-group">

<input name="route" type="text" class="form-control" placeholder="URL">

</div>

<div class="form-group">

<input type="submit" value="Add" class="btn btn-default" />

<a href="{{ route('navigation.index') }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div> </div>

## 5.28 admin.components.navigation.edit\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Edit navigation link</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div> @endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route("navigation.update", ['id' => $navigation->id]) }}" method="post">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="name" type="text" value="{{ $navigation->naziv }}" class="form-control" placeholder="Link name">

</div>

</div>

<div class="form-group">

<div class="form-line">

<input name="route" type="text" value="{{ $navigation->link }}" class="formcontrol" placeholder="Page URL">

</div>

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Edit">

<a href="{{ route("navigation.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.29 admin.components.posts.post\_tab.blade.php

<table id="mainTable" class="table table-striped">

<thead>

<tr>

<th>ID</th>

<th>Naslov</th>

<th>Sadrzaj</th>

<th>Video link</th>

<th>Slika</th>

<th>Izmeni</th>

<th>Obrisi</th>

</tr>

</thead>

<tbody>

@foreach($posts as $post)

<tr>

<td>{{ $post->postId }}</td>

<td>{{ $post->naslov }}</td>

<td>{{ $post->sadrzaj }}</td>

<td>{{ $post->video }}</td>

<td><img height="100" src="{{ asset($post->putanja) }}" alt="{{ asset($post->alt) }}"></td>

<td><a href="{{ route("posts.show", ['id' => $post->postId]) }}" >Izmeni</a></td>

<td><a href="{{ route("posts.delete", ['id' => $post->postId]) }}" >Obrisi</a></td>

</tr>

@endforeach

</tbody>

</table>

## 5.30 admin.components.posts.insert\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Add new post</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('posts.store') }}" method="post" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="title" type="text" class="form-control" placeholder="Naslov">

</div>

</div>

<div class="form-group">

<div class="form-line">

<textarea name="description" type="text" class="form-control" placeholder="Sadrzaj"></textarea>

</div>

</div>

<div class="form-group">

<div class="form-line">

<input name="video" type="text" class="form-control" placeholder="Video link">

</div>

</div>

<div class="form-group">

<label for="photo">Slika:</label>

<input name="photo" type="file" id="photo">

</div>

<div class="form-group">

<div class="form-line">

<input name="alt" type="text" class="form-control" placeholder="alt">

</div>

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Add">

<a href="{{ route("posts.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.31 admin.components.posts.edit\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Update existing post</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route("posts.update", ['id' => $post->postId]) }}" method="post" enctype="multipart/form-data">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="title" type="text" class="form-control" value="{{ $post->naslov }}" placeholder="Title">

</div>

</div>

<div class="form-group">

<div class="form-line">

<textarea name="description" type="number" class="form-control" placeholder="Description">{{ $post->sadrzaj }}

</textarea>

</div>

</div>

<div class="form-group">

<div class="form-line">

<input name="video" type="text" class="form-control" value="{{ $post->video }}" placeholder="Title">

</div>

</div>

<div class="form-group">

<label for="picture">Picture:</label>

<input name="picture" type="file" id="picture">

<br>

<img class="img thumbnail" height="200px" src="{{ asset($post->putanja) }}" alt="{{ $post->alt }}">

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Edit">

<a href="{{ route("posts.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.32 admin.components.roles.table.blade.php

<table id="mainTable" class="table table-striped">

<thead>

<tr>

<th>Name</th>

<th>Izmeni</th>

<th>Obrisi</th>

</tr>

</thead>

<tbody>

@foreach($roles as $role)

<tr>

<td>{{ $role->naziv }}</td>

<td> <a href="{{ route('roles.show', ['id' => $role->id]) }}">Izmeni</a> </td>

<td> <a href="{{ route('roles.delete', ['id' => $role->id]) }}">Obrisi</a> </td>

</tr>

@endforeach

</tbody>

</table>

## 5.33 admin.components.roles.insert\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Add new role</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('roles.store') }}" method="POST" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<input name="name" type="text" class="form-control" placeholder="Naziv uloge">

</div>

<div class="form-group">

<input type="submit" value="Add" class="btn btn-default" />

<a href="{{ route('roles.index') }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div> </div>

## 5.34 admin.components.roles.edit\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Edit navigation link</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route("roles.update", ['id' => $role->id]) }}" method="post">

{{ csrf\_field() }}

<div class="form-group">

<div class="form-line">

<input name="name" type="text" value="{{ $role->naziv }}" class="formcontrol" placeholder="Link name">

</div>

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary waves-amber" value="Edit">

<a href="{{ route("roles.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.35 admin.components.users.table.blade.php

<table class="table">

<tr>

<td>ID</td>

<td>Korisnicko ime</td>

<td>Email</td>

<td>Uloga</td>

<td>Izmeni</td>

<td>Obrisi</td>

</tr>

<!-- Prikaz korisnika-->

@isset($korisnici)

@foreach($korisnici as $korisnik)

<tr>

<td> {{ $korisnik->korisnikId }} </td>

<td> {{ $korisnik->korisnicko\_ime }} </td>

<td> {{ $korisnik->email }} </td>

<td> {{ $korisnik->naziv }} </td>

<td> <a href="{{ route('users.show', ['id' => $korisnik->korisnikId]) }}">Izmeni</a> </td>

<td> <a href="{{ route('users.delete', ['id' => $korisnik->korisnikId]) }}">Obrisi</a> </td>

</tr>

@endforeach

@endisset

</table>

## 5.36 admin.components.users.insert\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Add new user</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div> @endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('users.store') }}" method="POST" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<label>Korisnicko ime:</label>

<input type="text" name="korisnickoIme" class="form-control" value="{{ old('korisnickoIme') }}"/>

</div>

<div class="form-group">

<label>Email:</label>

<input type="text" name="email" class="form-control" value="{{ old('email') }}"/>

</div>

<div class="form-group">

<p><i>Uloga:</i></p>

@foreach($roles as $role)

<input id="role{{$role->id}}" name="role\_id" class="chk-col-deep-purple" value="{{ $role->id }}" type="radio">

<label for="role{{$role->id}}"> {{ $role->naziv }} </label>

@endforeach

<div class="form-group">

<label>Lozinka:</label>

<input type="password" name="lozinka" class="form-control" value="{{ (isset($korisnik))? $korisnik->lozinka : old('lozinka') }}"/>

</div>

<div class="form-group">

<input type="submit" value="Add" class="btn btn-default" />

<a href="{{ route("users.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div>

</div>

## 5.37 admin.components.users.edit\_form.blade.php

<div class="row">

<div class="col-md-4 col-md-offset-4">

<p class="lead">Update user</p>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('users.update', ['id' => $korisnik->id]) }}" method="post">

{{ csrf\_field() }}

<div class="form-group">

<label>Korisnicko ime:</label>

<input type="text" name="korisnickoIme" class="form-control" value="{{ $korisnik->korisnicko\_ime }}"/>

</div>

<div class="form-group">

<label>Email:</label>

<input type="text" name="email" class="form-control" value="{{ $korisnik->email }}"/>

</div>

<div class="form-group">

<p><i>Uloga:</i></p>

@foreach($roles as $role)

<input id="role{{$role->id}}" name="role\_id" class="chk-col-deep-purple" value="{{ $role->id }}" {{ $role->id == $korisnik->uloga\_id ? "checked" : "" }} type="radio">

<label for="role{{$role->id}}"> {{ $role->naziv }} </label>

@endforeach

<div class="form-group">

<label>Lozinka:</label>

<input type="password" name="lozinka" class="form-control" value="{{ $korisnik->lozinka }}"/>

</div>

<div class="form-group">

<input type="submit" value="Add" class="btn btn-default" />

<a href="{{ route("users.index") }}" class="btn btn-warning waveseffect">Cancel</a>

</div>

</form>

</div> </div>

## 5.38 admin.components.pages.gallery.blade.php

@extends("layouts.admin")

@section("content")

@empty($form)

@include("admin.components.gallery.table")

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div>

@endempty

@endempty

@isset($form)

@switch($form)

@case('edit')

@include('admin.components.gallery.edit\_form')

@break

@case('insert')

@include('admin.components.gallery.insert\_form')

@break

@endswitch

@endisset

@endsection

## 5.39 admin.components.pages.navigation.blade.php

@extends("layouts.admin")

@section("content")

@empty($form)

@include("admin.components.navigation.table")

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div>

@endempty

@endempty

@isset($form)

@switch($form)

@case('edit')

@include('admin.components.navigation.edit\_form')

@break

@case('insert')

@include('admin.components.navigation.insert\_form')

@break

@endswitch

@endisset

@endsection

## 5.40 admin.components.pages.posts.blade.php

@extends("layouts.admin")

@section("content")

@empty($form)

@include("admin.components.posts.post\_tab")

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div>

@endempty

@endempty

@isset($form)

@switch($form)

@case('edit')

@include('admin.components.posts.edit\_form')

@break

@case('insert')

@include('admin.components.posts.insert\_form')

@break

@endswitch

@endisset

@endsection

## 5.41 admin.components.pages.roles.blade.php

@extends("layouts.admin")

@section("content")

@empty($form)

@include("admin.components.roles.table")

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div> @endempty

@endempty

@isset($form)

@switch($form)

@case('edit')

@include('admin.components.roles.edit\_form')

@break

@case('insert')

@include('admin.components.roles.insert\_form')

@break

@endswitch

@endisset

@endsection

## 5.42 admin.components.pages.users.blade.php

@extends("layouts.admin")

@section("content")

@empty($form)

@include("admin.components.users.table")

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div>

@endempty

@endempty

@isset($form)

@switch($form)

@case('edit')

@include('admin.components.users.edit\_form')

@break

@case('insert')

@include('admin.components.users.insert\_form')

@break

@endswitch

@endisset

@endsection

## 5.43 components.footer.blade.php

<div class="footer">

<div class="container">

</div>

</div>

<div class="footer-bottom">

<div class="container">

<p>Template by<a href="http://w3layouts.com/"> w3layouts</a></p>

</div>

</div>

## 5.44 components.header.blade.php

<div class="header">

<div class="container">

<div class="header-info">

<div class="logo">

<a href="index.html"><img src="{{asset('images/logo.png')}}" alt=" " /></a>

</div>

<div class="logo-right">

<span class="menu"><img src="{{asset('images/menu.png')}}" alt=" "/></span>

<ul class="nav1">

@isset($menus)

@foreach($menus as $menu)

<li class="nav-item {{ (Request::url().'/' == asset($menu-

>link))? 'active' : '' }}">

<a class="nav-link" href="{{ asset($menu->link) }}">

{{ $menu->naziv }}

</a>

</li>

@endforeach

@endisset

@if(session()->has('user') && session()->get('user')[0]->naziv ==

'korisnik')

<li><a href="{{ route('createPost') }}" >Dodaj tutorijal</a></li>

@endif

@if(session()->has('user') && session()->get('user')[0]-

>naziv == 'admin')

<li><a href="{{ route('adminpanel') }}">Admin panel</a></li>

@endif

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

<li>

<a class="nav-link" href="{{ route('logout') }}">Logout</a></li>

@endif

<li><a href="{{ asset('dokumentacija.pdf') }}">Dokumentacija</a></li>

</ul>

</div>

<div class="clearfix"> </div>

<!-- script for menu -->

<script>

$( "span.menu" ).click(function() {

$( "ul.nav1" ).slideToggle( 300, function() {

// Animation complete.

});

});

</script>

<!-- //script for menu -->

</div>

</div>

</div>

## 5.45 layouts.admin.blade.php

<!DOCTYPE html>

<html lang="en">

@include('admin.components.common.head')

<body class="fixed-nav sticky-footer bg-dark" id="page-top">

<!-- Navigation-->

<nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top" id="mainNav">

<a class="navbar-brand" href="index.html">Start Bootstrap</a>

<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarResponsive">

@include('admin.components.common.sidenav')

@include('admin.components.common.topnav')

</div>

</nav>

<div class="content-wrapper">

@yield('content')

<!-- /.container-fluid-->

<!-- /.content-wrapper-->

@include('admin.components.common.footer')

<!-- Scroll to Top Button-->

<a class="scroll-to-top rounded" href="#page-top">

<i class="fa fa-angle-up"></i>

</a>

<!-- Logout Modal-->

<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" arialabelledby="exampleModalLabel" aria-hidden="true">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title" id="exampleModalLabel">Ready to Leave?</h5>

<button class="close" type="button" data-dismiss="modal" aria-label="Close">

<span aria-hidden="true">×</span>

</button>

</div>

<div class="modal-body">Select "Logout" below if you are ready to end your current session.</div>

<div class="modal-footer">

<button class="btn btn-secondary" type="button" datadismiss="modal">Cancel</button>

<a class="btn btn-primary" href="login.html">Logout</a>

</div>

</div>

</div>

</div>

<!-- Bootstrap core JavaScript-->

@include('admin.components.common.scripts')

</div>

</body>

</html>

## 5.46 layouts.front.blade.php

<!--

Author: W3layouts

Author URL: http://w3layouts.com

License: Creative Commons Attribution 3.0 Unported

License URL: http://creativecommons.org/licenses/by/3.0/

-->

<!DOCTYPE html>

<html>

<head>

<title>@yield('title')</title>

@section('appendCss')

<link href="{{asset('css/style.css')}}" rel="stylesheet" type="text/css" media="all" />

<link href="{{asset('css/bootstrap.css')}}" rel="stylesheet" type="text/css" media="all" />

@show

<link href='http://fonts.googleapis.com/css?family=Open+Sans:400,300,600,700,800' rel='stylesheet' type='text/css'>

<link href='http://fonts.googleapis.com/css?family=Lato:100,300,400,700,900,100italic,300italic,40 0italic,700italic,900italic' rel='stylesheet' type='text/css'>

<link href='http://fonts.googleapis.com/css?family=Playfair+Display:400,700,900,400italic,700itali

c,900italic' rel='stylesheet' type='text/css'>

<!-- js -->

@section('appendJs')

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

<!-- //js -->

<!-- for-mobile-apps -->

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<meta name="keywords" content="Floral Responsive web template, Bootstrap Web

Templates, Flat Web Templates, Andriod Compatible web template,

Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG, SonyErricsson, Motorola web design" />

<script type="application/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false);

function hideURLbar(){ window.scrollTo(0,1); } </script>

<!-- //for-mobile-apps -->

<!-- start-smoth-scrolling -->

<script type="text/javascript" src="{{asset('js/move-top.js')}}"></script>

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

<script type="text/javascript">

jQuery(document).ready(function($) {

$(".scroll").click(function(event){

event.preventDefault();

$('html,body').animate({scrollTop:$(this.hash).offset().top},1000);

});

});

</script>

@show

<!-- start-smoth-scrolling -->

</head>

<body>

<!-- header -->

@include('components.header')

<!-- header -->

<!-- content -->

<!-- content -->

<div class="content">

<div class="container">

<div class="content-text">

@yield('content')

</div>

</div>

</div>

<!-- //content -->

<!-- content -->

<!-- footer -->

@include('components.footer')

<!-- //footer -->

<!-- here stars scrolling icon -->

<script type="text/javascript"> $(document).ready(function() {

/\*

var defaults = {

containerID:

'toTop', // fading element id

containerHoverID: 'toTopHover', // fading element hover id

scrollSpeed:

1200,

easingType:

'linear'

};

\*/

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!-- //here ends scrolling icon -->

</body>

</html>

## 5.47 pages.autor.blade.php

@extends('layouts.front')

@section('content')

<div class="col-lg-4 col-sm-6 text-center mb-4">

<img class="rounded-circle img-fluid d-block mx-auto" src="{{asset('images/profilna.jpg')}}" alt="">

<h3>Marko Lazic</h3>

<p>My name is Marko Lazic. I was born in Serbia, Smederevo on October 21st. 1995.

After graduating high school in my home town I enrolled in ICT College of Vocational Studies. My interest in web programming is growing with each subject we learn. I hope that one day I will become a successful programmer.</p>

</div>

@endsection

## 5.48 pages.createPost.blade.php

@extends('layouts.front')

@section('title')

Registracija

@endsection

@section('content')

<div class="col-md-4">

<h3>Dodaj tutorijal</h3>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div>

@endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ asset('/post/store') }}" method="POST" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<label>Title:</label>

<input type="text" name="title" class="form-control" value="{{ old('title') }}"/>

</div>

<div class="form-group">

<label>Embed link:</label>

<input type="text" name="embedLink" class="form-control" value="{{ old('embedLink') }}"/>

</div>

<div class="form-group">

<label>Body:</label>

<textarea name="body" class="form-control" rows="7">{{ old('body') }}</textarea>

</div>

<div class="form-group">

<label>Photo:</label>

<input type="file" name="photo" class="form-control" />

</div>

<div class="form-group">

<label>Alt:</label>

<input type="text" name="alt" class="form-control" value="{{ old('alt') }}"/>

</div>

<div class="form-group">

<input type="submit" name="addPost" value="Add post" class="btn btn-default" />

</div>

</form>

</div>

@endsection

## 5.49 pages.gallery.blade.php

@extends('layouts.front')

@section('title')

Galerija

@endsection

@section('appendCss')

@parent

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

@endsection

@section('content')

<div class="row text-center text-lg-left">

@isset($galleries)

@foreach($galleries as $gallery)

<div class="col-lg-3 col-md-4 col-xs-6" >

<a href="{{ asset($gallery->putanja) }}" data-lightbox="image-1" class="d-block mb-4 h-100">

<img style="width: 210px;height: 160px" class="img-fluid img-thumbnail" src="{{ asset($gallery->putanja) }}" alt="{{$gallery->alt}}">

</a>

</div>

@endforeach

@endisset

</div>

@endsection

@section('appendJs')

@parent

<script src=" {{ asset('js/lightbox-plus-jquery.js') }}"></script>

@endsection

## 5.50 pages.home.blade.php

@extends('layouts.front')

@section('title')

Pocetna

@endsection

@section('content')

<div class="title">

<!-- POST -->

@isset($posts)

@foreach($posts as $post)

<div class="some-title">

<h3><a href="{{ asset('/posts/'.$post->postId) }}">{{

$post->naslov }}</a></h3>

</div>

<div class="john">

<p><a href="#">{{ $post->korisnicko\_ime

}}</a><span>{{ date("d.m.Y H:i:s", $post->created\_at) }}</span></p>

</div>

<div class="clearfix"> </div>

<div class="tilte-grid">

<a href="{{ asset('/posts/'.$post->postId) }}"><img style="height: 340px" src="{{ asset('/'.$post->putanja) }}" alt=" " /></a>

<span><p></p></span>

</div>

<div class="">

<a href="{{ asset('/posts/'.$post->postId)

}}">ULOGUJTE SE DA BISTE VIDELI TUTORIJAL</a>

</div>

<div class="border">

<p>a</p>

</div>

@endforeach

@endisset

|  |  |
| --- | --- |
|  |  |
| <!-- POST --> |  |
|  | <!-- paginacija --> |
| {{ $posts->links() }} |  |
|  | <!-- paginacija --> |
|  | <div class="clearfix"> </div> |
|  | </div> |

</div>

<div class="categories">

<div class="categ">

<div class="cat">

<!-- LOGIN FORM -->

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

<h3>Login</h3>

@endif

<div class="card my-4">

<div class="card-body">

<div class="row">

<div class="col-lg-12">

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

<form action="{{ route('login') }}" method="POST">

{{ csrf\_field() }}

<div class="form-group">

<label>Korisnicko ime:</label>

<input type="text" name="tbKorisnickoIme" class="form-control">

</div>

<div class="form-group">

<label>Lozinka:</label>

<input type="password" name="tbLozinka" class="form-control">

</div>

<input type="submit" name="btnLogin" value="Login" class="btn btn-success">

</form>

<br/>

<a href="{{ route('register') }}" class="btn btn-danger">Registracija</a>

<br/>

<br>

@empty(!session('error'))

<div class="alert alert-danger">{{ session('error') }}</div>

@endempty

@empty(!session('success'))

<div class="alert alert-success">{{ session('success') }} </div>

@endempty

@endif

</div>

</div>

</div>

</div>

<!--// LOGIN FORM -->

</div>

<!-- ANKETA -->

<div class="recent-com">

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger">{{ $error }}</div>

@endforeach

@endif

@endisset

</div>

@if(session()->has('user') && session()->get('user')[0]->naziv == 'korisnik')

<a href="{{ asset('/posts/create') }}" class="btn btn-warning">Create post</a>

<div class="recent-com">

<h3>Anketa</h3>

</div>

<br/>

@endif

<!-- ANKETA -->

</div>

</div>

<div class="clearfix"> </div>

@endsection

## 5.51 pages.register.blade.php

@extends('layouts.front')

@section('title')

Registracija

@endsection

@section('content')

<div class="col-md-4">

<h3>Register</h3>

@empty(!session('message'))

<div class="alert alert-success">{{ session('message') }}</div> @endempty

@isset($errors)

@if($errors->any())

@foreach($errors->all() as $error)

<div class="alert alert-danger"> {{ $error }} </div>

@endforeach

@endif

@endisset

<form action="{{ route('registerStore') }}" method="POST" enctype="multipart/formdata">

{{ csrf\_field() }}

<div class="form-group">

<label>Korisnicko ime:</label>

<input type="text" name="korisnickoIme" class="form-control" value="{{ old('korisnickoIme') }}"/>

</div>

<div class="form-group">

<label>Email:</label>

<input type="text" name="email" class="form-control" value="{{ old('email') }}"/>

</div>

<div class="form-group">

<label>Lozinka:</label>

<input type="password" name="lozinka" class="form-control" value="{{ (isset($korisnik))? $korisnik->lozinka : old('lozinka') }}"/>

</div>

<div class="form-group">

<input type="submit" name="registerKorisnik" value="{{ (isset($korisnik))? 'Change korisnik' : 'Add korisnik' }} " class="btn btn-default" />

</div>

</form>

</div>

@endsection

## 5.52 pages.single.blade.php

@extends('layouts.front')

@section('title')

Jedan

@endsection('title')

@section('content')

@isset($post)

<h1 class="my-4">{{$post->naslov}}</h1>

<br>

<!-- Portfolio Item Row -->

<div class="row">

<div class="col-md-8">

<iframe width="750" height="500" src="{{ $post->video }}">

</iframe>

</div>

<div class="col-md-4">

<h3 class="my-3">TO DO: </h3>

<p>{{$post->sadrzaj}}</p>

</div>

</div>

<!-- /.row -->

@endisset

@endsection

## 5.53 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!

|

\*/

// Pocetna

Route::get('/','FrontendController@index')->name('home');

Route::group(['middleware' => 'korisnik'], function(){

Route::get('/posts/{id}', 'FrontendController@single')->name("single"); });

Route::get('/post/create','FrontendController@createPost')->name('createPost');

Route::post('/post/store','FrontendController@storePost')->name('storePost');

Route::get("/gallery", "FrontendController@gallery")->name('gallery');

Route::get("/autor", "FrontendController@autor")->name('autor');

// Logovanje i registracija

Route::post('/login', 'LoginController@login')->name('login');

Route::get('/logout', 'LoginController@logout')->name('logout'); Route::get('/register', 'FrontendController@showRegistrationForm')->name('register');

Route::post('/registerStore','LoginController@registerStore')->name('registerStore');

// admin rute

Route::group(['middleware' => 'admin'], function(){ Route::get('/adminpanel', function () { return view('layouts.admin');

})->name('adminpanel');

// admin rute za korisnike

Route::get("/admin/users", "Admin\UsersController@index")->name("users.index");

Route::get("/admin/users/create", "Admin\UsersController@create")->name("users.create");

Route::get("/admin/users/{id}", "Admin\UsersController@show")->name("users.show");

Route::post("/admin/users", "Admin\UsersController@store")->name("users.store");

Route::post("/admin/users/{id}/update",

"Admin\UsersController@update")>name("users.update");

Route::get("/admin/users/{id}/delete",

"Admin\UsersController@destroy")>name("users.delete");

// admin rute za meni

Route::get("/admin/navigation",

"Admin\NavigationController@index")>name("navigation.index");

Route::get("/admin/navigation/create",

"Admin\NavigationController@create")>name("navigation.create");

Route::get("/admin/navigation/{id}", "Admin\NavigationController@show")-

>name("navigation.show");

Route::post("/admin/navigation",

"Admin\NavigationController@store")>name("navigation.store");

Route::post("/admin/navigation/{id}/update",

"Admin\NavigationController@update")>name("navigation.update");

Route::get("/admin/navigation/{id}/delete",

"Admin\NavigationController@destroy")>name("navigation.delete");

//admin rute za galeriju

Route::get("/admin/gallery", "Admin\GalleryController@index")->name("gallery.index");

Route::get("/admin/gallery/create",

"Admin\GalleryController@create")>name("gallery.create");

Route::get("/admin/gallery/{id}", "Admin\GalleryController@show")- >name("gallery.show");

Route::post("/admin/gallery", "Admin\GalleryController@store")->name("gallery.store");

Route::post("/admin/gallery/{id}/update",

"Admin\GalleryController@update")>name("gallery.update");

Route::get("/admin/gallery/{id}/delete",

"Admin\GalleryController@destroy")>name("gallery.delete");

//admin rute za uloge

Route::get("/admin/roles", "Admin\RoleController@index")->name("roles.index");

Route::get("/admin/roles/create", "Admin\RoleController@create")->name("roles.create");

Route::get("/admin/roles/{id}", "Admin\RoleController@show")->name("roles.show");

Route::post("/admin/roles", "Admin\RoleController@store")->name("roles.store");

Route::post("/admin/roles/{id}/update",

"Admin\RoleController@update")>name("roles.update");

Route::get("/admin/roles/{id}/delete",

"Admin\RoleController@destroy")>name("roles.delete");

// admin rute za postove

Route::get("/admin/posts", "Admin\PostController@index")->name("posts.index");

Route::get("/admin/posts/create", "Admin\PostController@create")->name("posts.create"); Route::get("/admin/posts/{id}", "Admin\PostController@show")->name("posts.show");

Route::post("/admin/posts", "Admin\PostController@store")->name("posts.store");

Route::post("/admin/posts/{id}/update",

"Admin\PostController@update")>name("posts.update");

Route::get("/admin/posts/{id}/delete",

"Admin\PostController@destroy")>name("posts.delete");

});